6-6-2014

Distributed smart camera network for safety and security

Nathan Fox
Santa Clara University

Matthew Kelley
Santa Clara University

Christopher Rapa
Santa Clara University

Christopher Yarp
Santa Clara University

SANTA CLARA UNIVERSITY

DEPARTMENT OF COMPUTER ENGINEERING
DEPARTMENT OF ELECTRICAL ENGINEERING

Date: June 4, 2014

I HEREBY RECOMMEND THAT THE THESIS PREPARED UNDER MY SUPERVISION BY

Christopher Yarp
Nathan Fox
Matthew Kelley
Christopher Rapa

ENTITLED

Distributed Smart Camera Network for Safety and Security

BE ACCEPTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREES OF

BACHELOR OF SCIENCE IN COMPUTER SCIENCE & ENGINEERING
BACHELOR OF SCIENCE IN ELECTRICAL ENGINEERING

Thesis Advisor

Thesis Advisor

Department Chair

Department Chair

SANTA CLARA UNIVERSITY

DEPARTMENT OF COMPUTER ENGINEERING
DEPARTMENT OF ELECTRICAL ENGINEERING

Date: June 6, 2014

I HEREBY RECOMMEND THAT THE THESIS PREPARED UNDER MY SUPERVISION BY

Christopher Yarp
Nathan Fox
Matthew Kelley
Christopher Rapa

ENTITLED

Distributed Smart Camera Network for Safety and Security

BE ACCEPTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREES OF

BACHELOR OF SCIENCE IN COMPUTER SCIENCE & ENGINEERING
BACHELOR OF SCIENCE IN ELECTRICAL ENGINEERING

_____
Thesis Advisor

_____
Thesis Advisor

_____
Department Chair

_____
Department Chair

Distributed Smart Camera Network for Safety and Security

by

Christopher Yarp
Nathan Fox
Matthew Kelley
Christopher Rapa

Submitted in partial fulfillment of the requirements
for the degrees of
Bachelor of Science in Computer Science & Engineering
Bachelor of Science in Electrical Engineering
School of Engineering
Santa Clara University

Santa Clara, California
June 6, 2014

Distributed Smart Camera Network for Safety and Security

Christopher Yarp
Nathan Fox
Matthew Kelley
Christopher Rapa

Department of Computer Engineering
Department of Electrical Engineering
Santa Clara University
June 6, 2014

ABSTRACT

Current CCTV surveillance solutions are generally retrospective tools. Because real time use of CCTV requires human monitors to view a potentially exorbitant number of video feeds, CCTV is usually only useful after an incident has occurred. However, new technologies are making it possible for machines to perform some tasks that previously required a human monitor. The proposed project seeks to augment existing CCTV systems with behavioral analytics. The system uses a series of cameras, FPGAs, and computers to track object movement throughout a facility. This information is used to build a model of normal movement. Object movements are compared against this model and any ones that diverge from the model are flagged for review by security personnel.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background and Motivation

Many companies and organizations employ Closed Circuit Television (CCTV) systems for security and safety reasons. The current CCTV technology is largely passive and is mostly used for reviewing a course of events after an incident has occurred. Organizations that wish to react in real time to developing situations are limited by the passive nature of CCTV. Time sensitive events such as medical emergencies and security breaches rely on surveillance staff to manually view the video feeds and look for potential problems. In many installations of these systems, the number of cameras can easily outnumber attentive monitoring personnel. This often lead to situations where is incidents go unnoticed by staff, further cementing CCTV systems as primarily retrospective tools. However, recent advancements in high density re-configurable hardware (FPGAs), parallel processing systems (GPUs), software, and other current fields of research have made it possible to develop automated and distributed systems that can operate without the need for constant human monitoring. This presents an opportunity to develop new systems that leverage the power of these new technologies to address the shortcomings of existing CCTV systems.

1.2 Project Overview

The proposed project is based around fixing the shortcomings of the currently used security systems by augmenting them with behavioral model processing. This "smart" surveillance system builds on the existing CCTV systems by attaching small processing modules to each camera. These processing modules monitor the video feeds, identifying objects and movement. Information about the movement of objects within the frame of view for a camera and information about the movement between cameras is collected and used to create a behavioral model. This information is accumulated over time and is formed into a definition of normal behavior. After this definition is created the real time object movements are compared against the normal behavioral and checked for abnormal behavior. If the system detects an anomaly, a notification is sent

to security informing them of the situation. The system is not meant to replace security staff but to provide them with a unique tool that can help direct their attention and increase safety and security in real time.

1.3 Motivation

The project was originaly intended to provide real time behavioral analysis to enhance crime prevention and security. However two incidents that took place during the implementation of this project have helped to inspire and motivate the creation of the system by realizing its potential applications. The first of these events occurred in late September / early October 2013 where a patient at a San Francisco Hospital tragically passed away in an exterior stairwell [7]. While a surveillance system was installed at the hospital [7], the patient was not seen entering the stairwell by monitoring staff. During the investigation of this missing patient officials were hampered by hardware problem while attempting to review the CCTV footage [7]. The other inspiring incident occurred in April 2014 where a 15 year old boy was able to breach perimeter security at Mineta San Jose International Airport and crawl into the wheel well of a jet destined for Hawaii [8]. The boy miraculously survived the flight [8] but the fact that he was able to breach airport security unnoticed is a disturbing realization. The airport, similar to the hospital, had a surveillance system in place before this event occurred. The boy was spotted by a camera while walking to the plane [8], however no surveillance staff were able to catch this event until he climbed out of the plane in Hawaii [8].

Both of these incidents highlight the flaws of the current CCTV systems and need for smarter surveillance systems. The hospital incident could have been prevented by a smart camera system that analyzed foot traffic by identity. This system would have realized that the patient left their room and had wandered into a section of the hospital where foot traffic was uncommon. Additionally, a behavioral analysis algorithm using time could have identified that the patient entered the stairwell but did not come out within a reasonable time frame. If these warnings had been communicated to monitoring staff or nurses, perhaps the patient would have survived. Similarly, if a smart camera system was installed at Mineta San Jose International Airport it could have identified that an individual was walking through the secured area without passing through one of the standard entry points. In addition, if the boy was running in the secured area, the system should have also identified abnormal movement rates. While the risk was relatively low in this case, it shows a security weakness in airport security which could be exploited in the future for more malicious reasons. The system being proposed here aims to address the deficiencies with current CCTV systems in both scenarios.

1.4 Significance

A CCTV system with semi-automated behavioral monitoring would have several distinct benefits over existing passive CCTV systems:

- Increased Personal Safety: In hospitals, assisted living facilities, schools and playgrounds, the system could monitor patents and children at times and in locations where staff may be unavailable or outnumbered. For example, the system could alert staff if a confused patient wanders away from their room during the night.

- Increased Security: In security applications, the system could identify criminals who had bypassed perimeter security, something that is currently difficult with human monitors. The system could also enhance security by identifying individual abnormal behavior and analyzing movement rates.

- Increased Fairness: Since the system uses behavior analysis, it is not susceptible to the racial, gender, religious, and social prejudices of human monitors.

Chapter 2

System Overview

2.1 High Level Architecture

In order to accomplish the ambitious project, the system has been split into several different modules, each with a specialized role. The project begins with a raw video feed from the IP Cameras of an existing CCTV system. An image processing module located directly behind the camera known as the FPGA or camera Back-end takes the video feed, prepares it for analysis, and performs object detection. Once an object is detected the image is cropped around it and is sent to be identified. The next module, the Identity Engine, takes each cropped image from the FGPA and ascribes an identity to the pictured object (ex. with facial or object recognition). The information created in the system about the object up to this point is bundled and saved into a database. Next, another module called the Behavior Analysis Engine processes the information in the database and attempts to understand the behavior of the object. If a behavioral discrepancy is detected, it creates an anomaly entry in the database. The User Interface, the primary interaction between the system and user, then pulls this information and displays it to the monitoring staff. It should be noted that this project is a proof of concept and may make some assumptions about the existence of certain technologies. The effectiveness of this system is dependent on progress made in advancing fields of research including facial recognition and super resolution. The architecture of the system described above is outlined in Figure 2.1.

2.1.1 Dataflow Example

The operation of this system is quite complex due to the modular design and innovative ideas, however understanding the architecture is best done by explaining the flow of data within the system. An example commonly used as reference in implementation and during conversational descriptions of this project is a hospital deployment. It is assumed in this case study that many aspects of setup and initialization are already completed. The setup involves the ideal placement of IP Cameras and attached FPGA Back-ends (hallways, nurse stations, entrances, and exits) and an onsite server room. The initialization includes the existence of

enough already collected data concerning the movement and speed of individuals throughout the hospital.

Now we can begin analyzing the example a patient wandering out of their room at night and collapsing in the hall. The cameras, being positioned in the hallways, capture images of the patient exiting their room and entering the hallway. The FPGA Back-end detects the face of the patient and passes this information to the Identity Engine. The Identity Engine recognizes the patient and collects information on their movement creating an entry in the database. While this is happening, the Behavioral Analysis Engine is checking all new entries into the database to determine if anomalous behavior has occurred. The FPGA Back-end is constantly sending the Identity Engine small metadata updates on the patient's position. When the patient falls, they remain in the frame longer than would be expected in the behavioral model. Since the patient has collapsed, they have not exited the frame of view of the camera and no exit event has been recorded. The Behavioral Analysis Engine would detect that the patient had not left the cameras frame in the normal amount of time and generate an anomaly. The anomaly is entered into the database and picked up by the User Interface. The User Interface notifies the monitoring staff of this anomaly and directs them to the appropriate video feed. Action can then be taken to help the patient.

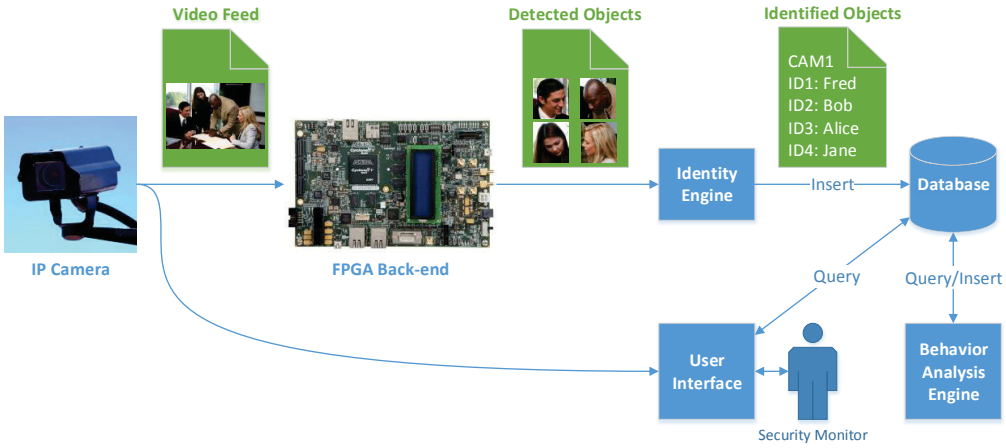


Figure 2.1: High Level System Architecture and Data Flow Diagram

2.2 System Component Descriptions

As the system is being designed modularly, each function of the system is assigned to a particular module. Detailed descriptions of each of these modules follow.

2.2.1 IP Camera

The purpose of the IP cameras is to capture video footage that will be analyzed by the rest of the system. IP cameras are one of the most common types of surveillance cameras used in modern CCTV installations. As one of the project goals is to be able to integrate with existing CCTV systems, IP cameras were chosen over other, less common, technologies. The project is only focusing on incorporating cameras that are ONVIF Profile S compliant. ONVIF is a security industry standard and Profile S is a sub-standard for IP cameras [9]. ONVIF guarantees a standard SOAP (Simple Object Access Protocol) interface to configure the cameras and request streams [10]. The cameras provide video streams in the form of MJPEG or H.264 compressed video over RTP/RTSP [9]. Focusing only on the cameras within the scope of this protocol allows us to ignore many of the compatibility issues that occur when developing a system for multiple camera vendors.

2.2.2 FPGA Back-end

The purpose of the FPGA Back-end is to receive a video stream from the existing IP Cameras and to ultimately detect objects within the video. When an object is detected, a message is sent to the Identity Engine with metadata and a cropped image of the object. The metadata includes the camera ID, a timestamp, the position in the image the object was detected, and the size of the image. In order to perform this object detection, the FPGA Back-end undergoes many steps each with a particular function. The operations conducted are listed in order of operation below:

1. Receives video from IP cameras
2. Processes video in preparation for analysis (ex. brightness / contrast)
3. Performs object detection
4. Crops image around objects
5. Performs additional processing on cropped image
6. Sends cropped images to Identity Engine
7. Continues to track objects

The FPGA Back-end is based on the Cyclone V SX SoC FPGA from Altera (5CSXFC6D6F31C8NES) [11]. This FPGA has the distinction of having a dual-core ARM Cortex A9 integrated as hard IP alongside the FPGA [12]. The ARM processor is utilized to run a version of embedded Linux. Linux software is used for managing the network functions of the FPGA Back-end as well as some of the video stream processing

(buffering and decompressing). The image processing is implemented on the FPGA and acts like a hardware peripheral from the perspective of the processor [2].

2.2.3 Identity Engine

The Identity Engine takes input from the FPGA Back-end in the form of cropped frames and assigns identities to them. First, cropped images of objects are received and compared to a set of known objects. In the case of the current implementation, this comparison is based on the arrangement of colors in the artificial target. Once the identity is determined, the Identity Engine queries the database to see if the particular object has already been seen in previous detections. At this point the database is queried for results on the existence of an object, if it has never been seen an identity is created. Then all the information created in the system thus far along with the queried information is grouped together to form an event. This event is then saved in the database for future analysis.

2.2.4 Database

The database is an essential back-end component for the system because it provides not only a place to store the vast quantities of data that the system will be generating, but it also provides a connection and method of communication between components of the project. The most important connection is to the Identity Engine. The Identity Engine sends new data to the database about the identities of objects and their events. The input of new data from the Identity Engine triggers the other database connections to begin analyzing. First, the Behavioral Analysis Engine pulls the information about object detection events and statistics model variables to begin analyzing. This data is used to detect anomalies by comparing new events against the saved models. The Behavioral Analysis Engine is constantly updating the saved behavioral statistical models based on new detection events. Simultaneously, the User Interface is pulling information from the database primarily to display recent anomalies and detection events. The User Interface also pulls information on initialization to setup the IP Camera connections. The database is based on SQL because it works well with the other modules of the system. This is primarily due the prebuilt libraries for SQL in the development languages of the modules within system.

2.2.5 Behavioral Analysis Engine

The Behavioral Analysis Engine is responsible for querying the database to compare recent movements against a model of normal movement. During the course of the project, the methods of determining abnormal behavior have change dramatically. Directed Weighted Graph Analysis was originally intended to be implemented as our method of behavioral detection. However, a simpler statistical model was implemented

given the scale and experimental nature of the project. This behavioral analysis was termed Frame Duration Analysis, based on how it was implanted. This type of analysis determines anomalous behavior based on comparing the speed of a subject against a statistical model. This comparison can be made because statisticians have determined that human walking speed corresponds to a normal distribution [13]. Although the full extent of the behavioral analysis that was not implemented, the ideas associated with the original project are novel.

2.2.6 User Interface

The User Interface is the only module in the project which interacts with the monitoring staff. The User Interface attempts to display the most relevant information within the system to the users, primarily objects, events, anomalies, and video feeds. Although this is only meant to be a proof on concept project, the partial success of this project depends upon the User Interface being intuitive and efficient. The User Interface still had to meet the requirements set forth by the design of the system to display the potential of the system. It will be written using Visual Studio Ultimate 2013 for code management, Team Foundation Services for application life cycle management, and Git for revision control. It was written in the C# language and used many libraries, such as VLC ActiveX Plugin, Microsoft LINQ, and many other standard .Net Development libraries.

2.3 Scope and Current Progress

The number and complexity of components in this system make the development of a near-commercial solution unrealistic within the time-frame of the project. Instead, the system was designed as a proof of concept to prove the overall methodology and architecture rather than serve as a commercial product. In order to expedite the development of an operational system, certain components that would be expected in a commercial system were replaced with simplified components. The modular architecture of the system allows components to easily be replaced with more advanced or commercial versions in the future.

One of the changes involves replacing facial recognition with a simpler stand-in. Although facial detection and recognition are used in commercial applications their cost and complexity made them impractical for this proof of concept. An artificial target was designed instead to work with simpler object algorithms that could be implemented within the timeframe of this project. This target, shown in Figure 5.1 is an example of one of these targets.

Many different algorithms were proposed for the Behavioral Analysis Engine. One of the proposed algorithms involved the use of Directed Weighted Graphs to detect individuals taking uncommon paths through a facility. While it was agreed that this algorithm would likely yield useful information, the budget only al-

lowed for a three camera system which would not allow a proper evaluation of the algorithm's effectiveness. Instead, Frame Duration Analysis was implemented as the method of behavioral analysis for the purposes of this project.

Additional scoping information is provided in the detailed chapter for each component.

2.3.1 System Progress

The system is currently in the final stages of implementation. With the overall system architecture finalized, each module underwent its own design and implementation process. Due to time constraints with such an ambitious project along with unanticipated challenges, not all of the modules were completed by the end of the development period. While MATLAB based simulations were completed for the FPGA Back-end and for the Identity Engine, the FPGA and CUDA implementation are still in progress. The MATLAB simulations are used as stand-ins for the actual components in the initial end-to-end demonstration.

2.3.2 FPGA Progress

The FPGA Back-end was designed and various image processing algorithms were evaluated. MATLAB was used to simulate the functionality of the FPGA and to verify the effectiveness of the chosen image processing algorithm. The hardware development environment for the FPGA was brought up on a Windows machine and the software development environment was brought up on an Linux VM (virtual machine). A custom Linux distribution with the software libraries required by the FPGA Back-end was developed using the OE (Open Embedded) build environment. The hardware glue logic linking all the system components including the ARM processor, DDR3 memory, and custom IP was created using the Qsys system level integration tool. Example IP that conformed to the Avalon bus standard used by Qsys was created to test system functionality. Corresponding software was also written to exercise the custom IP. More details on the FPGA demo IP can be seen in section 5.12.

2.3.3 Identity Engine Progress

The Canny Edge Detection Algorithm was researched and a MATLAB simulation was created. Libraries for the decoding of images from the FPGA Back-end, for interaction with the database, and for communicating with the FPGA Back-end were also researched. A CUDA based implementation of the Identity Engine has been prototyped.

2.3.4 Database and Behavioral Analysis Engine Progress

For the extent of this project these sections are completed and working properly. That being said with a continuation of this project these modules defiantly should be revisited and improved. For more information

on future improvements these sections see their respective implementation chapters.

2.3.5 User Interface Progress

The User interface is working at a base level it can handle responses and accurately display the information being created by the system. During the implementation many assumptions about future progress were made that should be corrected. Also a few the functional requirements that were not implemented should be built upon.

2.3.6 Expected Results

It is expected that this system will be able to detect targets within the field of view of cameras, identity them, and place entries in the database. The system will be able to generate a behavioral model based on real-time movement data. After an initial initialization period, abnormal movements such as an object stopping for an unprecedented amount of time or moving exceedingly fast will be flagged as anomalous and reported via the User Interface.

As with any system that attempts to do pattern matching and automated statistical model generation, it is unreasonable to assume that the system will be anywhere near 100% accurate. False positives (normal behavior interpreted as anomalous) and false negatives (failure to detect anomalous behavior) will likely occur during normal operation of the system. However, all efforts will be made to make the system as accurate as possible.

Chapter 3

Real World Constraints

Since this project is a proof of concept, many of the real world constraints were circumvented during the implementation. In order to successfully deploy this system as a commercial product, these constraints will need to be addressed. These constraints can be as simple as budgetary issues or as complex as the effectiveness of a component of this project.

3.1 Power

All of the components used in the system require electricity. This includes components explicitly listed in this document as well as other general but necessary components like network switches. In order to ensure the best possible reliability and resilience, the server should be located in a server room with redundant power and an Uninterrupted Power Supply (UPS). The cameras and FPGA have their own power requirements but their placement largely determines how their needs must be considered. Some network cameras come with Power Over Ethernet (PoE) support which allows them to be powered over the network cable by a special PoE switch. A PoE system may need to be employed in certain scenarios to make the project viable. As a point of reference the PoE camera (D-Link DCS-2310L) used in the project has a maximum power consumption of 5.3 Watts [14]. The FPGA Back-ends also require power but, since they are also peripherals on the network in their current configuration, they could potentially be placed farther away from the cameras and closer to a power source. Based on the reference document for the development kit, a maximum of 62.172 Watts would be consumed by the components used in the project [4]. This value depends on the actual design flashed onto the FPGA as well as the workload conducted on the ARM. The FPGA development kit used in the project does not support PoE and further research would need to be conducted to determine if it is a viable option.

3.2 Network

The IP Cameras employed in the prototype system use 10/100 BASE-TX Ethernet [14]. This form of Ethernet has a maximum theoretical throughput of 100 Mbits/sec [15]. A simple test was conducted using the Live555 library's QoS measurement utility to determine the bitrate received from the cameras running in TCP mode. For a 33 second test, the minimum bitrate was 1082 Kbit/sec, the average was 5533 Kbit/sec, and the maximum was 10434 Kbit/sec. This means that the theoretical maximum amount of video data that could be transfered simultaneously over one line would be $\frac{1024*100 \text{ kBit/sec Ethernet}}{5533 \text{ kBit/sec/stream}} \approx 18.5$ streams. Better evaluations of capacity for multimedia do exist [16]. These metrics take into account the "bursty" network traffic [16]. However, additional characterization of the network traffic would be required to use the more complex models. For the purposes of this document, we will under-spec the theoretical capacity by half in an attempt account for network congestion and fluctuation in the media stream datarates. In that case, 9 MJPEG stream could share one line. 9 cameras is actually quite small for many applications. However, the FPGA crops images that are sent to the identity engine, drastically reducing load on the connections to the identity engine. The size of the cropped image depends on the size of the object in frame but a typical size would be 34.3 kB (PNG) based on a test performed with an object approximately three feet from the camera. The metadata passed from the FPGA Back-end would be exceedingly small as it only needs to contain the NTP timestamp (64 bit), the camera ID (64 bit), the FPGA ID (64 bit), the object ID (64 bit), the position of the object in the frame (2x 32 bit), and the width and height of the object (2x 32 bit). The metadata would only be approximately 48 bytes per message.

In order to utilize the load reduction of the FPGA Back-end to alleviate network congestion, the network must be set up in the following way:

- The FPGA and the camera it processes are all connected locally via a switch
- The FPGA and cameras are on the same subnet

The load reduction relies on the fact that devices on the same subnet can communicate directly without a router thanks to Layer 2. Since each camera can connect to the FPGA directly via a common switch, the stream data is not sent over the entire network, consuming bandwidth. The FPGA, after processing the video sends only a very small amount of information to the central Identity Engine which must traverse to the center of the network.

Note that this capacity analysis does not consider the case when a copy of the stream from each camera must be recorded or displayed at all times. There are may factors that can affect how this impact the capacity requirements of the network including the quality and compression of the video. If this implementation is

required, the maximum number of cameras per link will decrease. Scaling to Gigabit Ethernet may help to alleviate capacity concerns if a large quantity of video must be recorded. An alternative solution would be to distribute recording equipment so that they are connected to the same switch as the FPGAs and cameras.

3.3 FPGA Capacity and Speed

Since facial detection was not implemented on the FPGAs in this proof of concept, a figure on how many faces a given FPGA can be detected at once is unavailable. However, it is safe to assume that such a limitation does exist. There are a selection of FPGAs on the market. The FPGA used in the project is a high-end model of the Cyclone V line of FPGAs from Altera. The name "Cyclone" is used to represent Altera's Low-Cost line of FPGAs with the midrange line referred to as "Arria" and the top-of-the-line being "Stratix". In general, as one goes up in device class, the number of logic elements and potentially the speed grade of the device increases. For examples, the Cyclone V line ranges from FPGA with no transceivers and 25,000 LEs (Logic Elements) to FPGAs with an embedded ARM processor, 9 5-Gbps transceivers and 110 LEs [17]. The Aria V line goes up to 660 LEs with an ARM processor in its SX variant [17]. Going up in lines also introduces new features. It is possible that a larger and more costly FPGA than was used in the prototype would be required to detect all the faces in a typical scene. Additional characterization would be required to determine the optimal FPGA for the application.

3.4 Database Connections and Storage

One of the primary real world constraints of a database is the volume of data being stored and the rate at which data is being generated by the system. The initial observations of the prototype connections indicate a relatively small size per entry associated with the Detection Events, Anomaly and Individual Tables; however issues with quantity may arise when an operational system is implemented. The primary concern for data growth is expected to be the KnownIMG Table because the size of an image, even a cropped image, is exponentially larger than the other database entries. During commercial implementation of this project the object identification should be replaced with facial recognition. This may change the usage pattern of the database and the database performance should be reevaluated at that time. Fortunately, Microsoft SQL Server does support very large databases up to 524.272 petabytes in size [18], which is far more than would be expected in even the largest installations of this system. SQL Server also has features built in for dealing with large file storage and access [19]. Clustering of SQL Servers [20] could also be used to address scalability concerns.

Another issue that may arise during the commercialization of this project is the number of SQL queries

being executed by the system at any given time. Scalability issues may occur if multiple Identity Engines, User Interface, and Behavioral Analysis Engine instances are present in a system, all polling the database for information simultaneously. This database stress should be analyzed carefully for each deployment of the system. Fortunately, Microsoft SQL Server is specified to be able to support a maximum of 32,767 user connections [18], far more than would be expected in any installation. SQL Server is also a fast Relational Database Management System which can handle anywhere from hundreds to hundreds of thousands of transactions per second depending on the configuration [21].

3.5 Behavioral Analysis Engine Processing

The real world constraints associated with the Behavioral Analysis Engine mainly relate to the accuracy of its analysis. Users may expect the system to perform at or near 100% accuracy. Like any form of behavior modeling, the system does not provide 100% accuracy and the customer must be made aware of this fact.

The only technical issue that may arise is the amount of processing power required to continuously run the more sophisticated behavioral algorithms. This might pose a problem for implementation but, theoretically, should be divisible into unique threads, mitigating the issue.

3.6 User Interface Scalability

At the end of this project the User Interface will not be easily scalable because of dependencies created during implementation. During the commercialization of this project these dependencies will have to be removed to allow for multiple working environments. Some of these dependencies include the lack of a working installer, assumed database configuration and connection, file directory dependencies, and libraries not included in the files associated with the project.

3.7 Target Identification

There are four major factors that constrain real world target identification: lighting, distance, orientation, and number of targets. Lighting, distance and orientation affect the appearance of a target. Inconsistent appearance from frame to frame can cause erroneous target identification. Targets may either be misidentified, or not identified at all. It requires fairly sophisticated algorithms to combat these factors. For the purposes of testing, we held the targets at close range and parallel with respect to the camera lens. Additionally, the tests were conducted indoors under uniform lighting.

Total number of targets is a concern of the system capacity and throughput. For each target in the frame the FPGAs and Identity Engine GPUs must use a certain number of clock cycles to process it. Too many

targets may require too many clock cycles to complete the entire frame. This will propagate through the entire system as a delay. All delays affect system performance and may result in data degradation. This is especially true when dealing with time sensitive statistics.

Chapter 4

Development Bill of Materials (BOM) & Budget

The cost of the system is realtively high because not only is this is an area of ongoing research but it also requires expensive new hardware and software platforms. However, the hardware costs of this system are not unreasonable given the cost of high end CCTV systems that may include high definition IP Cameras that can cost over $1,000 a piece. The proposed development budget for the project is shown in Table 4.1. The budget provides an itemized breakdown of all of the system materials required for the initial proof of concept implementation. The selected IP cameras were chosen because they are used in many current CCTV systems and support the ONVIF standard. Altera Corporation has generously donated two Cyclone V SoC Development Kits and SoC Embedded Design Suites as well as three copies of Quartus II Subscription Edition. Several Microsoft products are being licensed through Microsoft's standard academic initiative, DreamSpark.

The development system only included three cameras but could likely support many more. The largest capital expenditures for the system are the servers. Thanks to the system architecture, much of the image processing is distributed to the FPGA Back-ends, reducing the load on the Identity Engine. While the Identity Engine has not undergone extensive load testing, it is expected to be able to support many cameras in a normal use case. Depending on the traffic density, potentially hundreds of cameras could be handled by one Identity Engine. See chapter 6 for more information on how the Identity Engine performance depends on traffic density. The Database Management System used in this project, Microsoft SQL Server, has a long running track record in industry. It is a scalable, high performance solution [20] that can support a large number of transactions per second [21]. A single SQL Server instance would be expected to support hundreds of cameras. The User Interfaces would each require a computer with a windows operating system, these are expected to exist in currently operational CCTV systems. The Behavioral Analysis Engine could either be distributed and run in the background on the user interface machines or piggy-backed onto the server.

4.1 Real World Cost Estimates and BOM

The costs shown in Table 4.1 represent the theoretical costs of developing the proof of concept system. It does not represent the actual cost of the end system because it includes several components which, while necessary for development, are not needed in a deployed system. These components include the development tools and, to some extent, the development kit which contains many features that were left unused.

Table 4.2 contains an itemized list of the major components that would need to be included in a production version of the FPGA back-end. It is largely based on the Cyclone V SoC Development Kit part list [4] with the unused components of the development kit removed. Overall, each FPGA back-end board is expected to cost approximately $566.57 which is reasonable given that the price of some "smart cameras" on the market have prices that reach into the thousands of dollars.

Table 4.3 contains the specifications to a server comparable to the one used in the proof of concept system. This server would able to run both the SQL Server instance as well as the Identity Engine. In most installations one server of this type should be sufficient. In high load environments, additional RAM may be required.

Description	Mfr.	Mfr. Part Number	Qty.	Cost / Unit	Cost	Amount Requested from SCU	Donation Expected	Provided by
Quartus II Subscription Edition	Altera	SW-QUARTUS-SE-FIX	3	$2,995.00	$8,985.00	0	$8,985.00	Altera
Cyclone V SoC Development Kit and SoC Embedded Design Suite	Altera	DK-DEV-5CSXC6NES	2	$1,595.00	$3,190.00	0	$3,190.00	Altera
ONVIF Compliant Fixed IP Camera	D-Link	DCS-2310L	2	$297.99	$595.98	$595.98	0	
ONVIF Compliant 720P PTZ IP Camera	Panasonic	WV-ST165	1	$488.99	$488.99	$488.99	0	
Seagate Backup Plus 2 TB USB 3.0 Desktop External Hard Drive	Seagate	STCA2000100	1	$89.99	$89.99	$89.99	0	
Seagate Barracuda 1 TB HDD SATA Internal Bare Drive	Seagate	ST1000DM003	3	$67.19	$201.57	$201.57	0	
Misc. Cables / Connectors			1	$80.00	$80.00	$80.00	0	
Microsoft Windows Server 2012 Standard Edition	Microsoft		3	$882.00	$2,646.00	0	$2,646.00	Dreamspark Academic (Microsoft)
Microsoft SQL Server 2012 Standard	Microsoft	1 Core License (Min 4 Per Server)	4	$1,793.00	$7,172.00	0	$7,172.00	Dreamspark Academic (Microsoft)
Microsoft Visual Studio Premium (w/ MSDN)	Microsoft		4	$6,119.00	$24,476.00	0	$24,476.00	Dreamspark Academic (Microsoft)
Team Foundation Service (TFS)	Microsoft	1 License	1	0	0	0	0	Microsoft (Free Service)
				Total	$47,925.53	$1,456.53	$46,469	
Key:								
System Materials								
Development Tools								

Table 4.1: Development BOM - For Development Setup + Development Tools

Description	Manufacturer	Manufacturer Part Number	Vendor	Vendor Part Number	Qty	Cost/Unit	Cost
Cyclone V SoC FPGA	Altera	5CSXFC6 D6F31C7N	Digikey	5CSXFC6D 6F31C7N-ND	1	$283.61	$283.61
MAX V CPLD System Controller	Altera	5M2210Z F256I5N	Digikey	544-2973-ND	1	$25.90	$25.90
Programmable LVDS clock	Silicon Labs	570FAB0 00973DG	Silicon Labs	Custom Part	1	$40.00	$40.00
50 MHz crystal oscillator	Silicon Labs	510GBA50 M0000BAG	Silicon Labs	Custom Part	1	$5.00	$5.00
MagJack 1000BaseT	Bel Fuse	L829-1J1T-43	Digikey	380-1110-ND	1	$5.75	$5.75
Real Time Clock	Maxim	DS1339C	Digikey	DS1339C-33#-ND	1	$6.78	$6.78
32M 16 8, 1024-MB DDR3 SDRAM	Micron	MT41K256 M16HA-125:E	Digikey	MT41K256M16 HA-125:E-ND	1	$7.18	$7.18
256-Mb NOR flash	Altera	EPCQ256SI16N	Digikey	544-2779-ND	1	$50.00	$50.00
32-Kb EEPROM	Microchip	24LC32A	Digikey	24LC32A-I/SN-ND	1	$0.41	$0.41
Micro SD Card Slot	Wurth	693 071 010 811	Digikey	732-3819-2-ND	1	$2.10	$2.10
Power Supply Controller	Linear	LTC2978	Digikey	LTC2978AI UP#PBF-ND	1	$21.51	$21.51
Misc Components					1	$100.00	$100
PCB (6“ x 6” Standard 2 Layer) Low Quantity	Advance Circuits				1	$18.33	$18.33
						Total	$566.57

Table 4.2: FPGA Estimated BOM - Based Largely on Cyclone V SoC Dev Kit Part List [4]

Description	Cost
Colfax ProEdge SXT8600: -Intel Xeon E5-2637 (x2) -16GB DDR3 RAM (4x2048MB each) -120GB SSD -6TB Hard Drive -Nvidia Tesla K20C -Microsoft Windows Server Standard 2008 R2 SP1	$8,715.49

Table 4.3: Server Estimated BOM - Based on Online Quote from Colfax International [5]

Chapter 5

FPGA Back-end Design and Implementation

5.1 Purpose

Before a description of FPGA Back-end Design and Implementation can be presented, it is important to state the tasks that the FPGA is responsible for performing. The FPGA Back-end is tasked with:

1. Receiving video from IP cameras
2. Processing video in preparation for analysis (ex. Brightness / contrast)
3. Performing object detection
4. Cropping images around objects
5. Performing additional processing on cropped images
6. Sending cropped images to the Identity Engine
7. Continuing to track objects

One of the goals of the FPGA Back-end is to reduce the load on the central Identity Engine. It accomplishes this by distributing image processing. Since each camera is attached to an an FPGA Back-end, multiple FPGA Back-ends can be processing multiple video streams simultaneously. The FPGA Back-end is able to crop images around objects in the frame that must be identified by the identity engine. By performing this step, the bandwidth requirements between the FPGA Back-end and the Identity Engine is reduced along with the load on the Identity Engine since it knows images passed to it have an object in the center. This frees it from having to processes multiple 720P video streams simultaneously to find objects to identify. The decision to separate object detection and object identification was informed by the fact that facial detection and facial recognition are often separated with facial detection being much easier to implement and easier to

perform than facial recognition. By separating object detection and recognition, it allows these components to be fitted with updated versions of their respective algorithms at a later date.

5.2 Scoping

In a production system, the FPGA would use a facial detection algorithm, potentially coupled with additional object tracking. In scoping this project, the overall system architecture was considered to be the primary goal. As such, an artificially designed target was used in place of faces for this proof of concept. The system was designed so that the object detection method used for the target could later be replaced with facial detection IP. The artificial target (shown in Figure 5.1) makes use of a black circle at the center and a black ring around the outside for object detection. The inner two colored rings are used for identification with each permutation of colors representing a different target.

Figure 5.1: Example Target

5.2.1 FPGA Paradigm Shifts

The inclusion of the ARM Processor with the FPGA constitutes a major shift for the FPGA design methodology. While soft core processors have existed for some time, they entirely relied on the FPGA fabric to run and thus were subject to the performance constraints of the FPGA. While FPGAs are fast, they are not as fast as an ASIC. Because the included ARM processor is hard IP with its own clock, it can run at a much higher clock rate than the FPGA. This shifts the focus of the FPGA from being the center of the design to being a resource to be used by the ARM processor. Customs hardware accelerators are one possible application of the FPGA fabric. While this is an exciting new direction for FPGAs, it has required reworking the typical FPGA development process. While the development tools have been adapted to this new process, there is a substantial ease of use barrier and steep learning curve.

5.2.2 Current Progress

The FPGA development kit and the development process that came with it were brand new at the time that the project was being developed. As with any new technology, there were some issues getting this new platform up and running. How to utilize the FPGA from the ARM processor as well as how to streamline the development processes is still under investigation. As such, a complete end to end demo of the FPGA could not be completed by the project deadline. The underlying design of the FPGA Back-end, however, is still relevant and given more development time would likely be completed. The information given in section 5.3 represents the design of the FPGA Back-end while the information in section 5.11 section 5.12 discuss a simulation of the FPGA Back-end in Matlab and the work accomplished on the FPGA respectively.

5.3 Development Platform Overview

5.3.1 Hardware Overview

In order to reduce development time and complications, the FPGA Back-end was built using an existing development kit produced and provided by Altera Corporation. The development kit being used is the *Cyclone V SoC Development Kit and EDS*. It provides all the hardware required for the FPGA Back-end including one Gigabit Ethernet adapter, two 10/100 Ethernet adapters, DDR3 memory, power supply circuity, and a LCD line display [1]. An image of the development is shown in Figure 5.2. The FPGA featured in the development kit is the 5CSXFC6D6F31C8NES model of the Cyclone V SoC [1]. Specifications for this chip are reproduced in section 16.1 for the reader's convenience. The key differentiator for this FPGA is that it includes a dual-core ARM Cortex-A9 as hard IP alongside the FPGA fabric [1]. Rather than having to instantiate a soft core processor, which takes up valuable LEs (Logic Elements) in the FPGA fabric, the higher performance ARM Cortex-A9 provides all of the functions of a standard embedded processor along with having the ability to manage the configuration of the FPGA fabric and to interface directly with some hard IP such as the Gigabit Ethernet controller. Bridges exist to allow the ARM processor to interact with soft IP configured in the FPGA fabric and vice versa [12].

5.3.2 Software Overview

Development Software

Several different software packages were used in the development of the FPGA Back-end. The main software packages that were used were:

- Quartus II Subscription Edition (Altera Corporation)
- OpenEmbedded Build System - Bitbake (Openembedded.org)

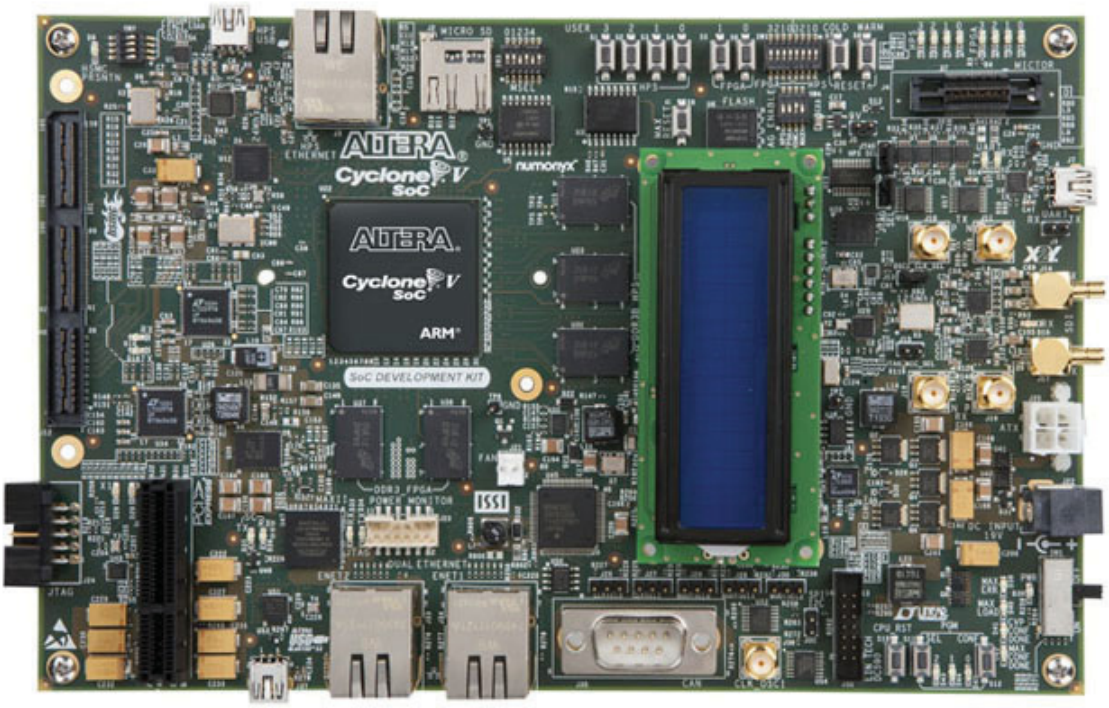


Figure 5.2: Cyclone V SoC Development Kit - Image Courtesy Altera Corporation [1]

Quartus II is part of the Altera suite of development tools for FPGAs. Implementation of the FPGA hardware accelerators used in the project was done in Quartus II and system level integration was done in the companion Qsys tool.

Bitbake, OpenEmbedded's build tool, was used to compile and package a custom Linux distribution based on the Yocto distribution which included the device drivers and libraries necessary to interface with the FPGA as well as our application software.

5.3.3 Embedded Software

Several software packages are used in the FPGA Back-end while it is running:

- Yocto Linux (Linux Foundation)
- LIVE555 (Live Networks, Inc.)
- libjpeg (Independent JPEG Group)

Based on reference designs from Altera [22] it was decided that Linux development would be done instead of bare-metal development. Several factors contributed to this decision including the large amount of software already available for Linux, the ability of Linux to handle low level functions such as the network

stack, and the relative difficulty and complexity of bare-metal development. Also, because of the common use of ARM processors in embedded systems such as smart phones, embedded Linux distributions have already been created which cross compile to the ARM architecture. Yocto Linux is the base distribution used in Altera example systems and is what the FPGA Back-end uses [22]. LIVE555 is a RTP/RTSP streaming library used to open and buffer the RTP/RTSP video streams [23] from the IP cameras. libjpeg is a library [24] used to decompress the MJPEG video received from the cameras. Other open source software packages are used in the custom distribution and are detailed in the *Licenses* directory of the distribution.

While the end-to-end application for the FPGA Back-end was not written due to the complexities discussed in subsection 5.2.2, live555 was tested with a sample application. Live555 was able to successfully open a connection to the camera and start a MJPEG stream. libjpeg was cross compiled based on an existing OE recipe and included with the custom Linux distribution loaded on the FPGA. All indications are that the application software would not take long to develop as most of the work is being conducted by libraries or the FPGA accelerators. It is likely that the end-to-end application could have been developed in a matter of days once the hardware was ready.

5.4 FPGA Back-end Design - Data Flow Perspective

Looking at how data flows through the FPGA Back-end provides a lot of insight into how the FPGA Back-end was designed and implemented. A data flow diagram for the FPGA Back-end is shown in Figure 5.3. This diagram shows how the FPGA Back-end interacts with other components in the system. It uses the SOAP protocol to configure the cameras and the video stream. This service is guaranteed by the ONVIF Profile S standard [9]. The video is streamed to the FPGA Back-end over the RTP/RTSP protocols and is decoded using MJPEG. The FPGA Back-end processes the video and uses sockets to send the cropped images, along with some metadata, to the Identity Engine.

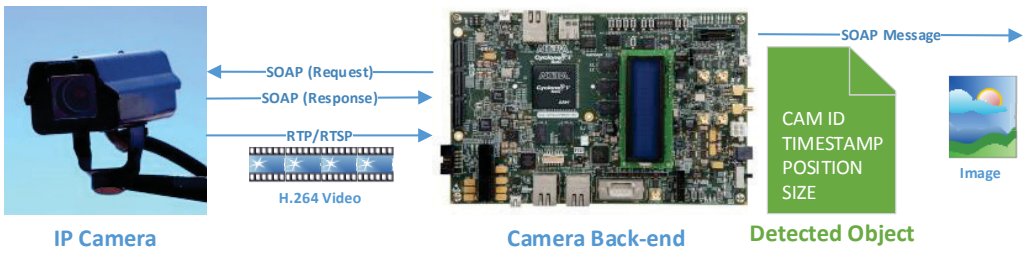


Figure 5.3: FPGA Back-end Data Flow Diagram

5.5 FPGA Back-end Design - Software Perspective

Now that the Data Flow perspective has been presented, the design of the FPGA Back-end software can be detailed. A data flow diagram from the software perspective is presented in Figure 5.4. Embedded Linux handles the network stack, sends and receives SOAP messages, manages/buffers the RTP/RTSP video stream, and decodes the MJPEG video. LIVE555 is used to manage the RTP/RTSP video stream while libjpeg is used to decode the MJPEG video. The image processing takes place in a hardware accelerator implemented in the FPGA. The software provides the decoded images to this hardware accelerator (which appears as a processor peripheral) and reads the results back when they are available. The results are packed up and sent to the Identity Engine via socket connections.

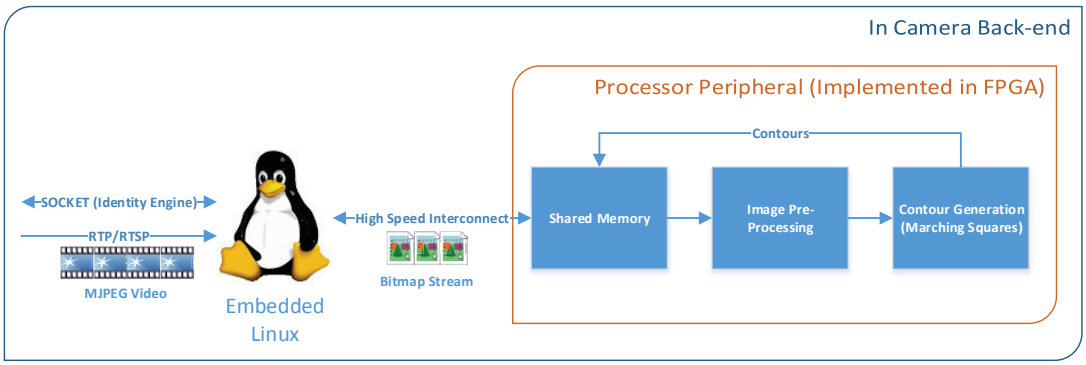


Figure 5.4: FPGA Back-end Software Data Flow Diagram

5.6 FPGA Back-end Design - Hardware Perspective

Now that the software architecture has been presented, the hardware architecture can be detailed. An architectural diagram is shown in Figure 5.5.

The components in the blue region of the diagram are part of the Hard Processor System (HPS). These components have minimal configurability and act just like the hardware peripherals of a standard embedded system. The green region represents the FPGA fabric, which can be reconfigured completely. Three bridges allow the HPS and components in the FPGA to communicate. The HPS2FPGA is a high speed, high bandwidth bus and is used for large data transfers [12]. The LWHPS2FPGA is a lightweight bus used mostly for controlling and checking the status of IP components in the FPGA [12]. The FPGA2HPS bridge provides a way for FPGA IP to access the HPS [12]. The interconnect between IP components in the FPGA fabric uses the proprietary open Avalon standard, created by the Altera Corporation [25]. Specifically, the Avalon-MM (memory mapped) standard is being used. This standard employs a master - slave methodology where mas-

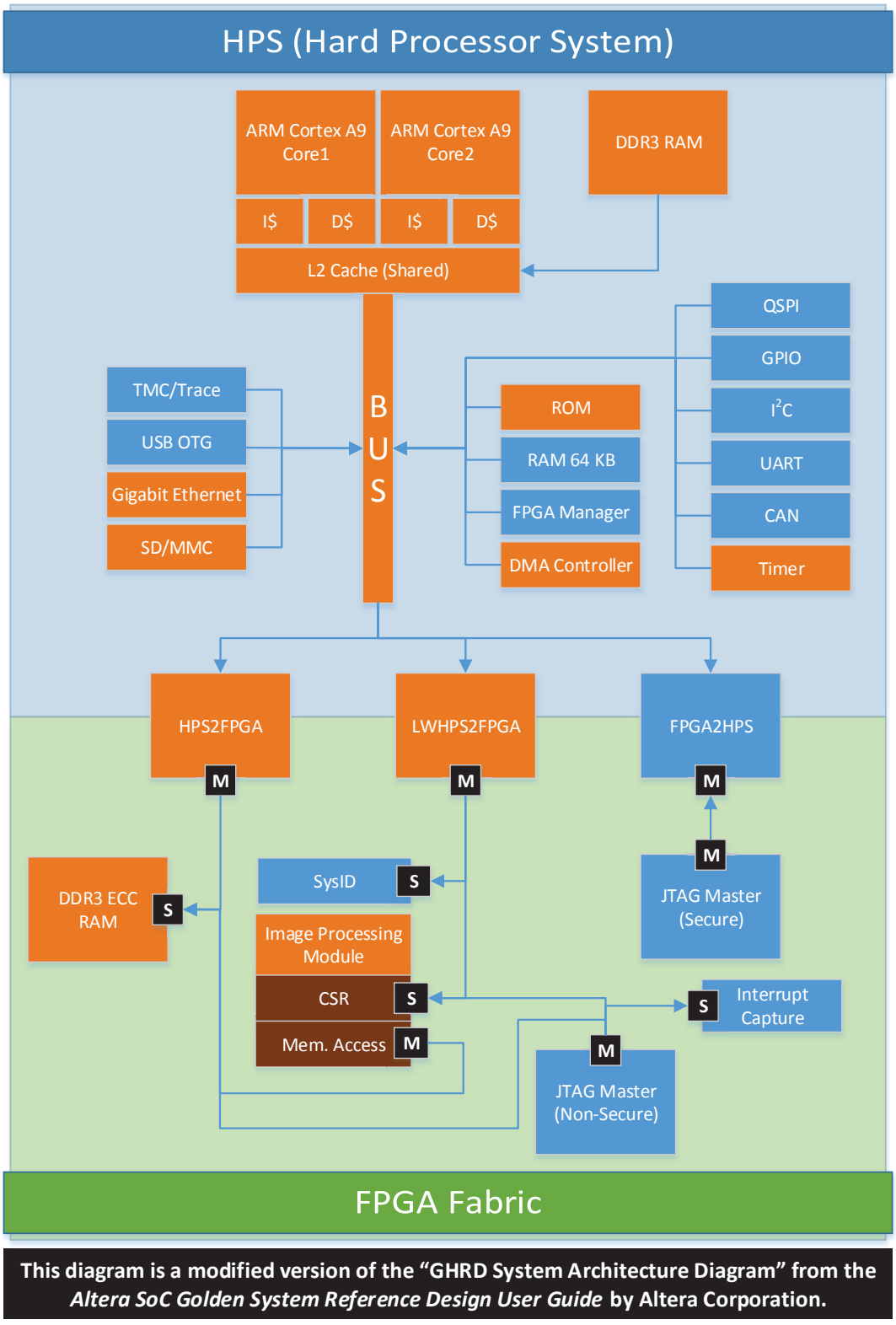
ters can exchange messages with slaves [25]. Each slave is mapped to the address space of the master [25]. Avalon allows many-to-many relationships between masters and slaves [25]. Master and slave interfaces are denoted in Figure 5.5 as black squares with a “M” for “Master” and “S” for “Slave”.

The FPGA Back-end architecture is based on the *GHRD (Golden Hardware System Reference Design)* from Altera [2] [22] [26]. Only certain components are used by the FPGA Back-end and are represented as the orange blocks in Figure 5.5. The ARM processor (along with its memory subsystem) is used to run Linux. The Gigabit Ethernet adapter is used to communicate with the IP cameras and the Identity Engine. The SD/MMC peripheral is used to store the Linux OS image. The ROM is used to store the FPGA configuration. The DMA controller is used to manage memory operations.

The Image Processing Module is the custom hardware accelerator created for the FPGA Back-end. It is a Verilog module and conforms to the Avalon-MM standard. It provides two interfaces: a Control Status Register (CSR) and a Memory Access interface. A DDR3 ECC Memory Controller is instantiated in the FPGA fabric. This is connected to the high bandwidth bridge as well as the Memory Access interface of the Image Processing Module. This memory is used as shared memory between the processor and the Image Processing Module. It is used to store video frames to be processed as well as the cropped output images and metadata from the Image Processing Module. Because the CSR does not have large bandwidth requirements, it is connected to the lightweight bridge.

In order to fully describe the motivation and utilization behind the hardware architecture, the operations the FPGA Back-end performs are presented below:

1. Video data is received and decoded by the ARM processor
2. The decoded raw RGB data is copied into the shared DDR3 memory (via the FPGA memory controller)
3. The address of the image data in the shared memory is set in the CSR of the Image Processing Module to tell it where to find the new image
4. The Image Processing Module processes the image and generates cropped images and metadata
5. The cropped images and metadata are stored in the shared memory
6. The CSR is updated with the address of the output data in shared memory
7. The ARM processor polls the CSR and, when the address of the output data changes, it copies the output data to the HPS memory
8. The ARM processor packages the output data and sends it to the Identity Engine



This diagram is a modified version of the "GHRD System Architecture Diagram" from the *Altera SoC Golden System Reference Design User Guide* by Altera Corporation.

Figure 5.5: FPGA Back-end Hardware Architecture, Adapted from Altera Reference Design Diagram [2]

5.7 Video Format Selection

The MJPEG format was selected as the format for the video being consumed by the FPGA Back-end for several reasons. First and foremost, it is a format that is guaranteed by the ONVIF Profile S standard [9]. Secondly, unlike H.264 or MPEG, each frame of MJPEG video is a complete frame. In H.264 or MPEG, only I-frames are complete frames. Other frames only contain differences from previous frames. The problem is that if a frame is lost or dropped, the video will be corrupted until the next I-frame is received. This can take quite some time depending on how the codec is configured. Since the ARM processor is managing the stream and doing the de-multiplexing and decoding of the video, it may be unable to keep up with the number of frames coming in and be forced to drop some. In this case, MJPEG's complete transmission of each frame is preferable. The difference in format are summarized in Table 5.1.

MJPEG	H.264
Each frame is a complete frame	On I-frames are complete. The rest are diffs from previous frames
Higher bandwidth required	Less bandwidth required
Lossy Compression	

Table 5.1: Video Formats

5.8 Object Detection - Contour Forming

5.8.1 General Method

The targets being detected (see Figure 5.1) all have a black circle in the center and a black ring around the outside. Each of these features has a white boarder. The boundaries between the black features and the white boundaries create clearly defined closed edges. The ratios of the edge diameters are set by the design of the target. Targets can therefore be identified by looking at edges and comparing their center points and diameters. For each target, the edges of the rings will have very similar center-points since the edges are all concentric. Finally, since the edges should be circular, the aspect ratio of the edges (ratio of width to height) should be 1:1. Since concentric edges, especially those with the ratios defied by the target, are uncommon it is not actually necessary to ensure the edges are circular. Checking aspect ratio has worked well in our tests as a way to slim down the number of edges to process without introducing the added complexity of determining if an edge is completely circular. In addition, the diameter of the circular edge is the same as the height or width of the edge's 1:1 bounding box, greatly simplifying calculation of the diameter. This does mean, however, that any series of concentric edges where the ratios of the bounding boxes are equivalent to those of the target will be recognized as a target. This is rare however and could be remedied by the inclusion additional processing step including more sophisticated ways of identifying edges as circles. This would

likely include checking the perimeter of the edge and the area enclosed are consistent with a circle.

The parameters defining the tolerance on edge center-point spacial locality, edge aspect ratio, minimum points in edge, and minimum edge dimensions can all be fine tuned to provide an optimal solutions. Because the focus of this project was to demonstrate the overall architecture, these values were set and tested until a target could be detected reliably. The values selected may not be the optimal values and further testing would likely improve object detection accuracy.

The steps to target identification can be broadly described below:

1. Extract closed edges from image
2. Discard edges that are too small to provide meaningful dimensions (noise in image or very small objects) - used to reduce data set
3. Discard edges with aspect ratios not close to 1:1
4. Group edges with close center-points and put them in "bins"
5. Discard any bin with less than 3 edges (targets have at least 3 edges)
6. In each bin, find the edge for the center ring (should be smallest edge)
7. Compare ratios of each edge in bin against the edge of the center ring and check for defined ratios.
8. If both the inner and outer ring edges are identified, then the collection of edges constitutes a target.
9. Crop image around bounding box of outer ring (will include fill target).

This algorithm could be modified and optimized further. One assumption that was made in this algorithm is that the smallest edge in a bin is the edge around the center circle of the target. Due to noise in the image or some other phenomena, this may not be the case. The algorithm could be made more robust by checking the ratio of edges to each successively larger edge until there are no more edges to compare against or the target is correctly identified. While this would increase the robustness of the algorithm to real world conditions, it would also add complexity and ensure a larger load on the system. Since the artificial targets were created for the proof of concept, and because this algorithms would be replaced by facial detection in a production system, it was decided that the less robust but faster algorithm would be suitable for the initial implementation.

5.8.2 Contour Lines vs. Edges

The algorithms discussed in subsection 5.8.1 assume that there is a way to extract closed edges from an image but does not describe how that operation is performed. There are several algorithms that can be used

for edge detection including the Canny Edge Detection algorithm which is discussed in detail in section 6.2. While Canny Edge Detection is a well researched algorithm, it does not guarantee that the edges it detects are closed. An example of this can be seen in the seminal paper for Canny Edge Detection [27] where the results of running example images through the edge detection algorithm yield some unclosed edges.

Project advisors Professor Sally Wood suggested that, since the targets had clearly defined, closed edges along the border of the black circle and the black ring (see Figure 5.1) that contour lines be considered instead of edge detection.

Contour lines and edges of an image appear very similar. However, contour lines denote the boundaries of the the portions of the image that are below a certain intensity threshold and the segments of the image that are above that threshold. A good way to think about this is to visualize the image like a mountain range with different intensity values being different elevations. The contour lines are the different elevation lines on the mountain. In fact one of the most common applications of contour lines is in topographic maps as elevation lines.

5.9 Marching Squares

There are several method that can be employed to extract the contour lines of an image. One of the best know algorithms is commonly known as the "Marching Square" algorithm. The Marching Square Algorithm was never officially published but is the 2D case of the Marching Cubes Algorithm [28]. The seminal paper on the Marching Cubes algorithm, published in 1987 by William E. Lorensen and Harvey E. Cline [29], is referenced by most papers discussing the Marching Squares Algorithm. The 2D Marching Dquares case is considered a natural extension of the Marching Cubes Algorithm and is considered by many image processing insiders as common knowledge. As such, it is often not explained in depth in papers. However, it is explained in the paper *Marching Cube Algorithm: Review and Trilinear Interpolation Adaptation for Image-Based Dosimetric Models* by D.A. Rajon and W.E. Bolch [30].

The marching square algorithm consists of the following steps [30]:

1. Select a threshold value and compare each pixel in the image against the threshold.
2. Compare 2x2 pixel blocks (with no overlap) against a look up table to determine the contour segments in that block
3. Perform linear interpolation of the contour segments along the boarder of the 2x2 pixel blocks to obtain a more accurate contour

An illustrations of the of the first two steps of the algorithm are shown below. While the linear interpola-

tion is important to form very accurate contours, it is not required, strictly speaking. In order to simplify the implementation of the algorithm and the explanation here, the linear interpolation step will be omitted.

Figure 5.6 shows the look up table used by the marching square algorithm. The grey box represents a pixel that is above the threshold and a white box represents a pixel that is below the threshold. The red line represents a contour segment. Note that there are two cases that have both red and blue lines. These are ambiguous cases as either the red lines or the blue lines could be the contour segments for those cases [30].

Figure 5.7 shows the original image while Figure 5.8 shows the image after it is compared against the threshold value. Figure 5.9 shows the result of the image after each 2x2 block is compared against the look up table. The red lines represent the contours. Note that in order to know the direction of the contour at the edge, a 1 pixel overlap with an adjacent segment of the image is required. This is represented by the green line.

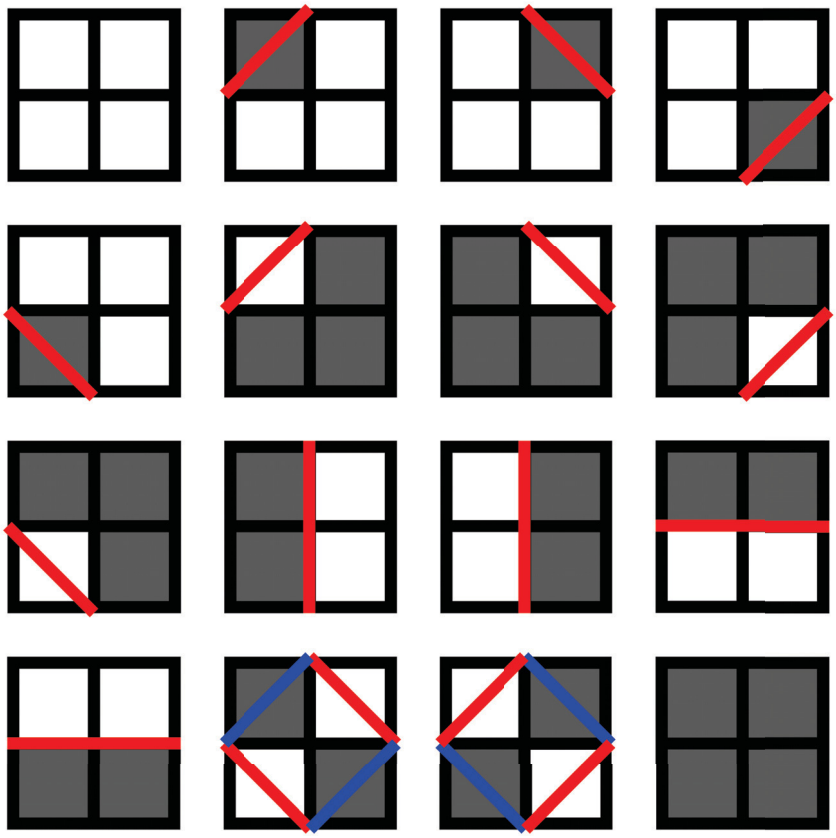


Figure 5.6: Marching Square Look Up Table

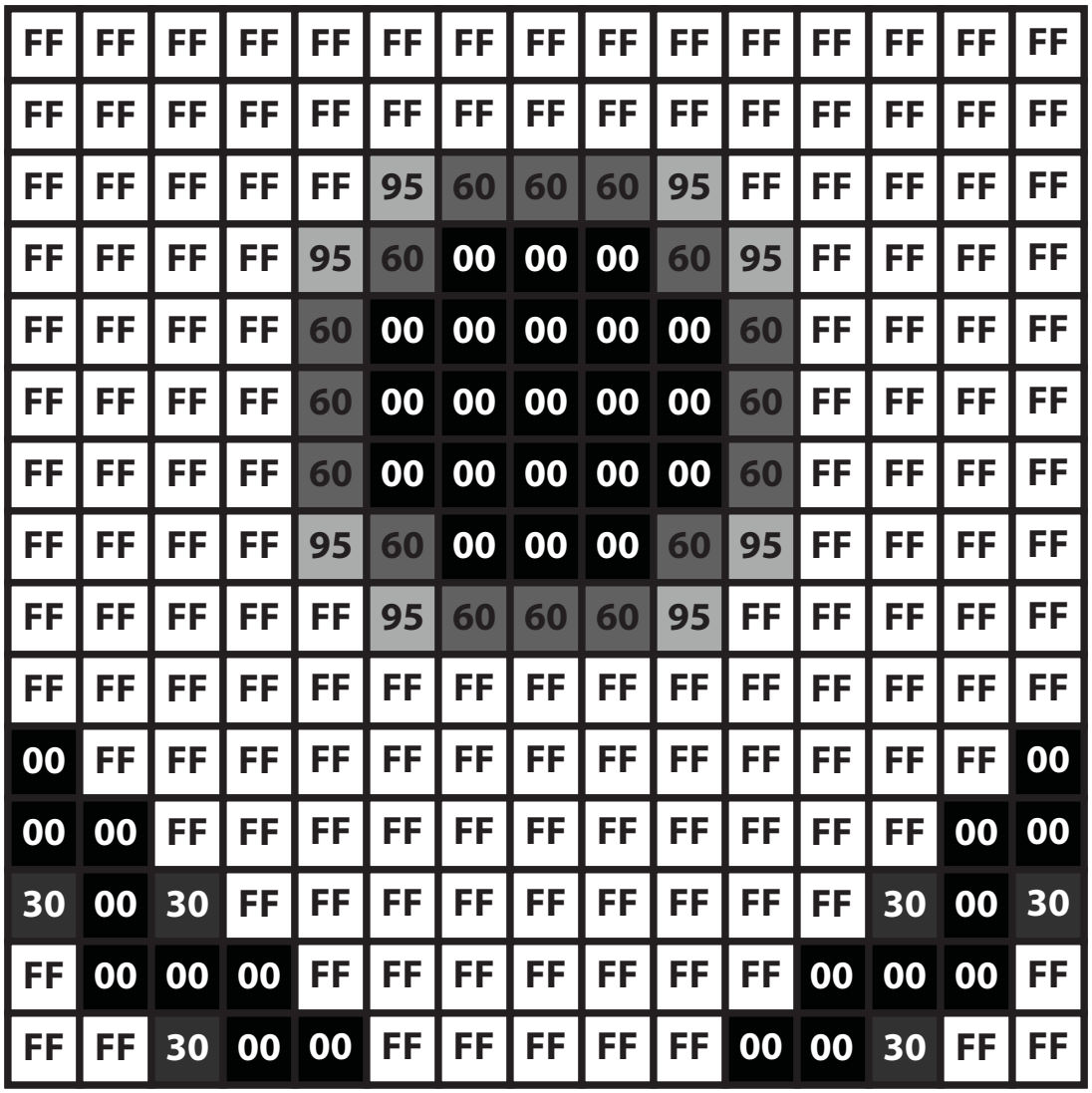


Figure 5.7: Original Image

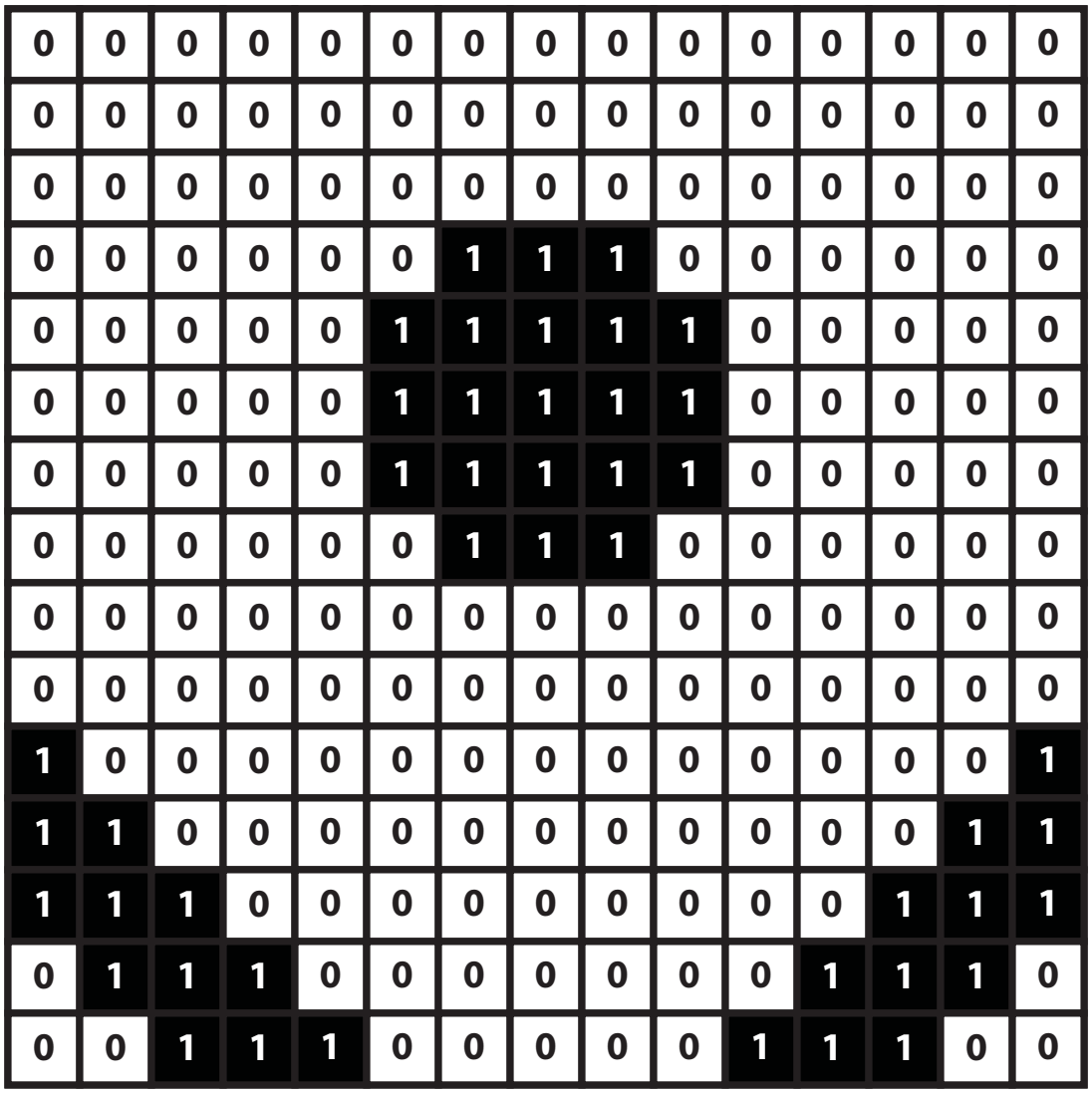


Figure 5.8: Image After Comparison to Threshold 0x45

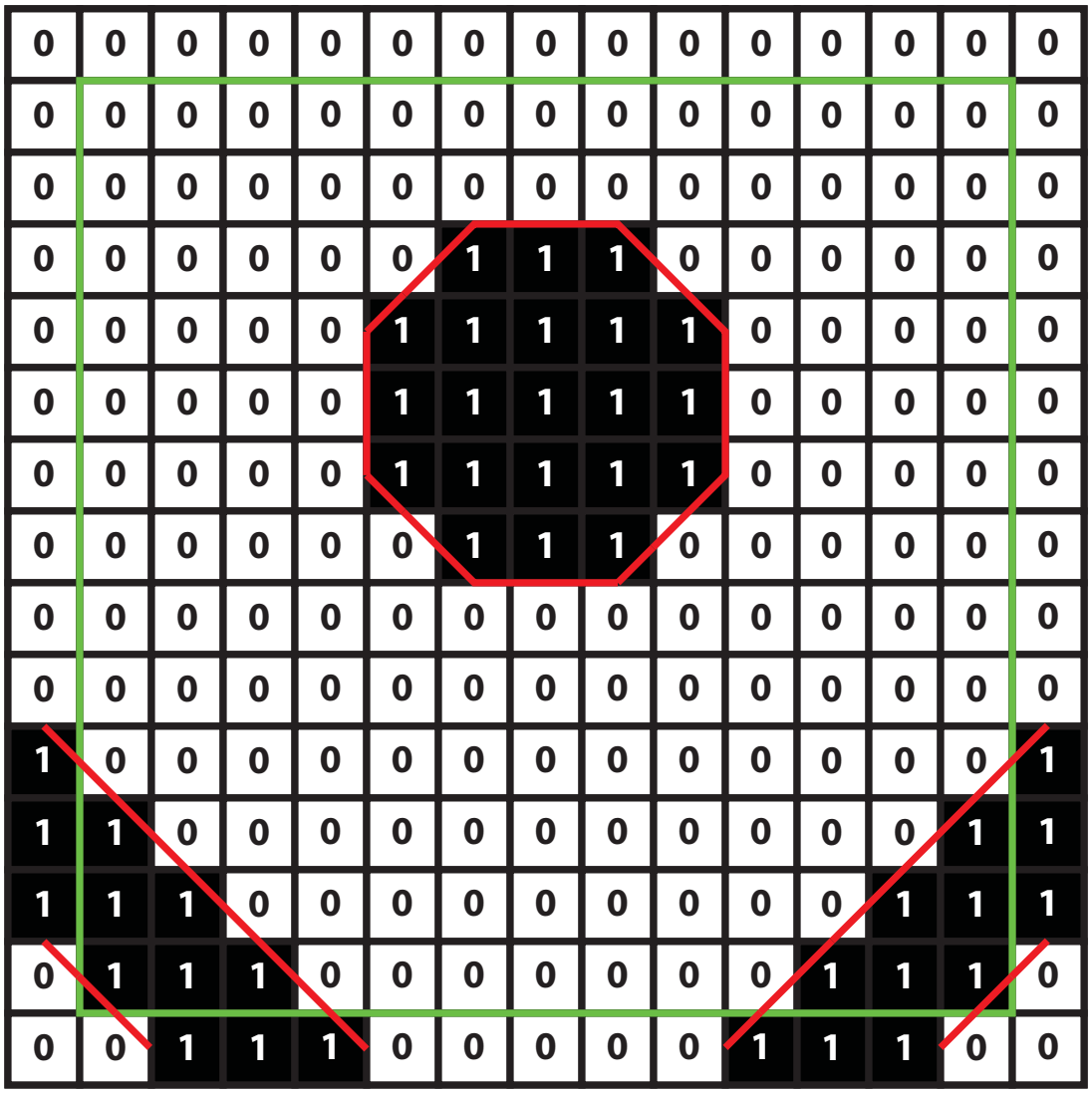


Figure 5.9: Image After Evaluating 2x2 Pixel Blocks for Contour Segments

5.10 Marching Squares Design

The marching square algorithm as described in section 5.9 is clearly parameterizable. However, it is not clear what the best parallel implementation of the algorithm is. Further more, the above description of the algorithm could be easily implemented by a processor based solution but the mapping of the solution to a FPGA platform is more nuanced. FPGAs do not behave in the same way as processors; rather than executing a set of instructions, FPGAs are essentially configurable logic networks. Multiple datapaths can exist and be processed completely in parallel. In an attempt to exploit the parallelism of the FPGA platform an alternate implementation of the marching squares algorithm was drawn up.

It should be noted that FPGAs have a limited number of configurable logic elements. Developing a design to process a full uncompressed HD image from the cameras would require more resources than the FPGA has. In order to avoid this issue, the image is segmented into blocks to be processed. Note that a 1 pixel overlap on each side is required to determine the shape of the contour at the edge of the block

The initial step of the marching square algorithm remains largely the same: individual pixel intensity values are compared against a threshold. The FPGA implementation does this operation for each pixel in parallel.

The next stage of the process is carried out by a network of processing units. Figure 5.10 shows the processing modules in blue and the connections in green. The white and black squares in between the green connections represent the pixels. Each processing unit has access to a 2x2 pixel segment of the image. Each processing unit is directly connected to the unit above, below, to the left, and to the right. When the threshold compared image block is ready, each processing unit looks at the values of the four pixels it is connected to. Each processing unit is an implementation of the look up table described in Figure 5.6. The processing units configure their external connections based on the look up table. This state is shown in Figure 5.11 with the red lines representing the mapping between external interfaces. Each processing unit also has two memory units (since there can be a maximum of two contours per computing unit). This memory unit is used to hold the min x, max x, min y, and max y values for each contour passing through the processing unit. *Note that, since contours cannot overlap or cross, the four tuple should be unique to each contour in the block.* Each processing unit knows its position within the network and is therefore able to set the initial maximum and minimum values for each contour segment. The state of the processing units after this initial setup is shown in Figure 5.11. The orange compute units represent units which have had a maximum or minimum change since the last clock cycle.

At this point, each compute unit believes that maximum and minimum coordinate of the contours that pass through it are the coordinates it initially assigned; this is of course incorrect. The next stages of the algorithm

involve propagating the maximum and minimum values throughout each contour. Propagation of maximum and minimum values occurs on each clock cycle by each compute unit sending its current maximum and minimum values to its connected neighbors that are part of the given contour (determined by the look up table in the setup phase). The neighbor compares their minimum and maximum values to the values received along the connection and update their maximum and minimum values accordingly. Figure 5.12 shows this process for one compute unit during the first round of the algorithm. A flag is set for each compute unit signifying if its min/max registers changed in the last cycle. The processes is completed when a clock cycle has passed and no min/max register has changed. At this point, the system is said to have "settled". Figure 5.13 shows the system at this final stage. It should be noted that the propagation occurs in parallel for each contour but the total time depends on the contour with the longest settling time. The best case scenario is for a contour which is closed in the block. In this case, it takes 1/2 the length of the contour cycles for the min/max values to fully propagate. In the worst case scenario, when the contour segment crosses over the boundary of the image block being processed, it takes the length of the contour segment cycles for the min/max values to settle.

At this point the complete contours must be put into a set. This process is more difficult to conduct in parallel and the implementation proposed here involves visiting each compute unit to identity whether or not it contains a contour and if that contour has already been included in the set. There is likely a more efficient and elegant way of addressing this last stage of the algorithm but additional investigation would be required to find it.

There are two proposed methods for dealing with contour segments which enter and leave the image block being processed. One method involves creating a list of contour segments that are then "stitched" back together by the processor. An alternative method is to use memory units to store the coordinates of the contour segments as they enter or leave the bottom or right edge of the image block. The saved values are then used as inputs for later runs of the algorithm on the neighboring image blocks. This requires that the image blocks be evaluated in order since it introduces data dependency. However, it does provide the benefit of ignoring contour segments until they form a closed loop. Figure 5.10, Figure 5.11, and Figure 5.13 all show the memory hardware required for this method and how they are connected to the compute units.

5.11 Object Detection Simulation

The suitability of contour lines to perform the object detection detailed in subsection 5.8.1 was tested using a simulation in MATLAB. The `contour()` function was used in place of the FPGA marching square implementation to get the contours in the image. Video recorded from one of the IP cameras was converted to

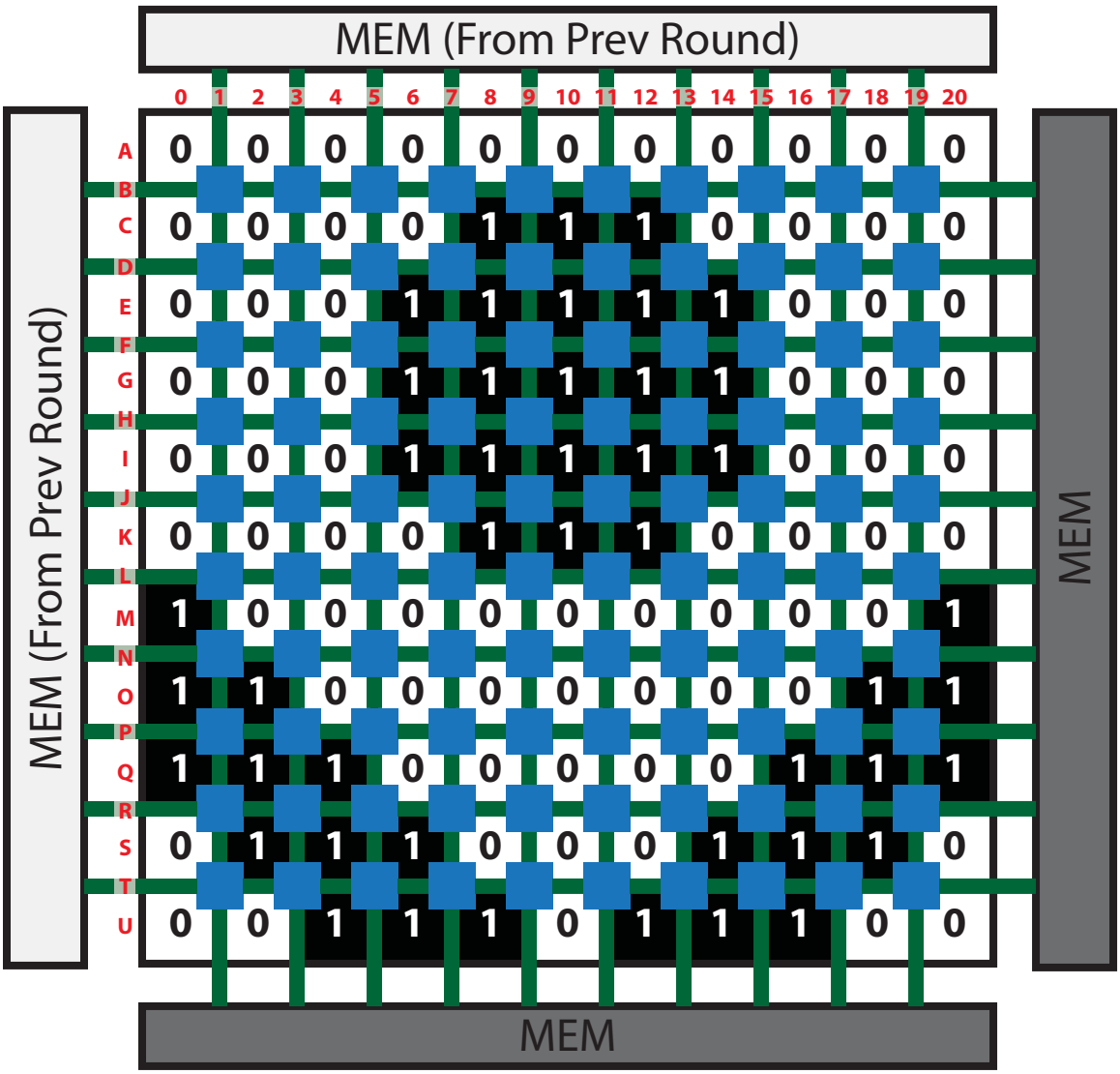


Figure 5.10: Layout Processing Units and Connections Between Them

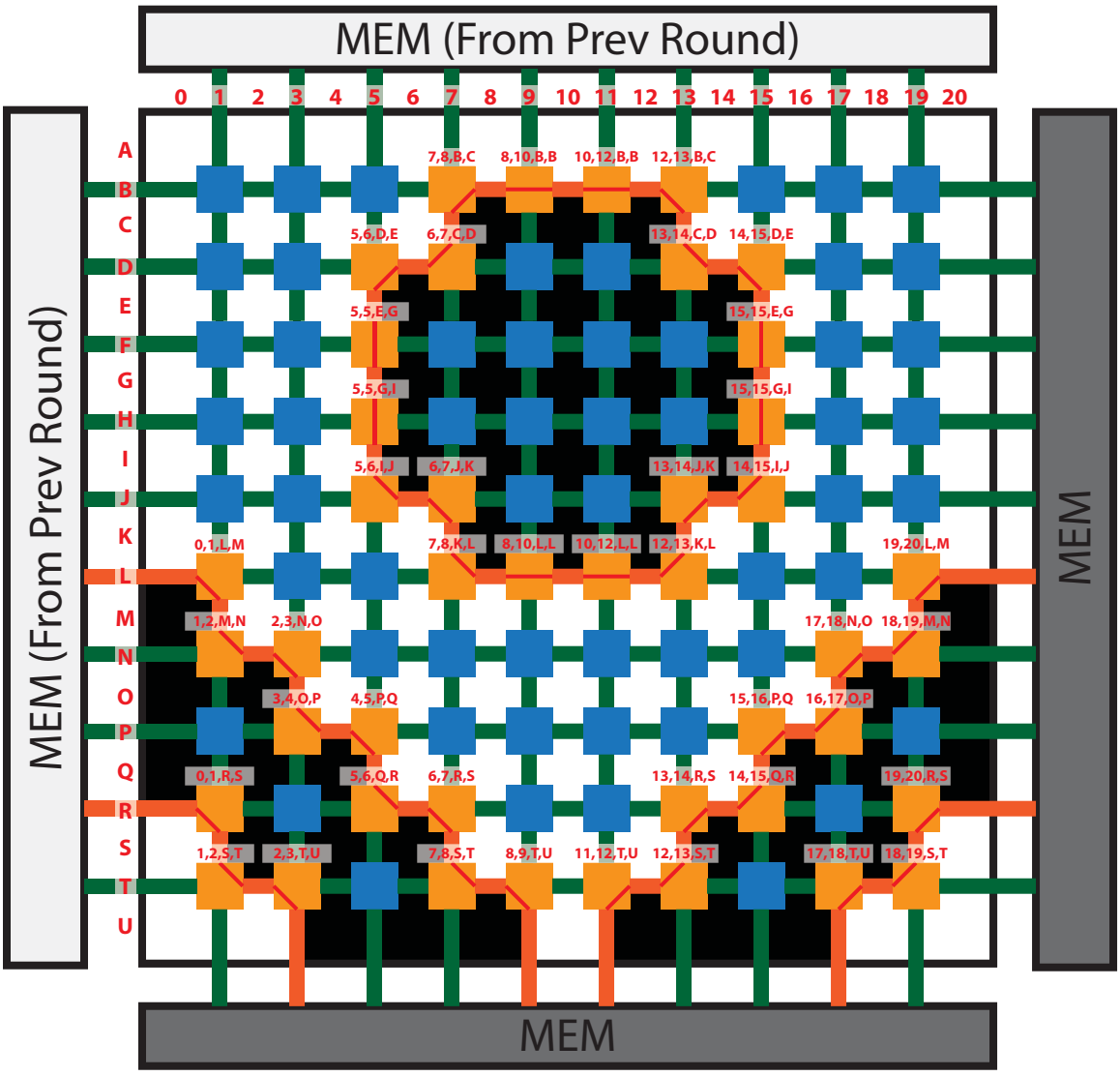


Figure 5.11: Processing Units After Setup

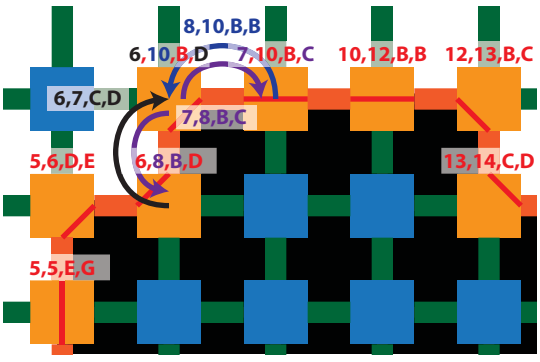


Figure 5.12: Processing Units During Step 1

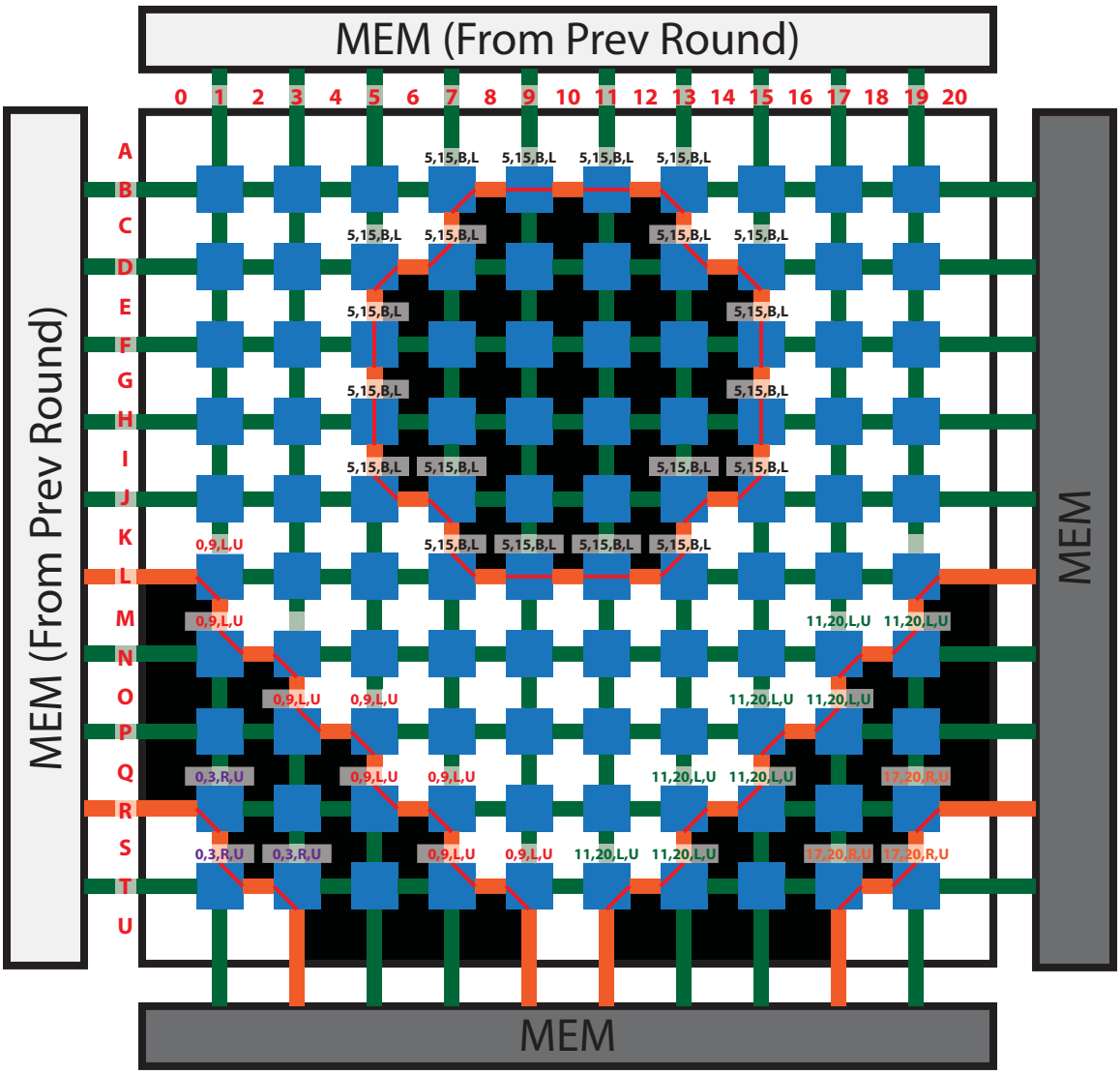


Figure 5.13: Processing Units After Final Step

images and read into MATLAB. The MATLAB code for this simulation can be seen in section 16.2. The result of the simulation is shown in Figure 5.14. The result of the simulation for the distant target is shown in more detail in Figure 5.15. The images shown were part of a preliminary test to determine the distance the targets could be from the camera while still being reliably detected. In the initial test, it was determined that the target detection became unreliable as the target got to the end of the image processing lab. Increasing the tolerance of the diameter ratios to 1.25 allowed the algorithm to be more reliable when the targets were farther away. Additional fine tuning of parameters would likely improve reliability further. The issues with distant targets can begin to be seen in Figure 5.15. As the objects get father away, the number of pixels which can represent the rings decreases. This problem is compounded by the fact that the camera use lossy compression schemes that result in compression artifacts. These artifices can be seen in Figure 5.15 with some irrelevant contours appearing inside the black ring.

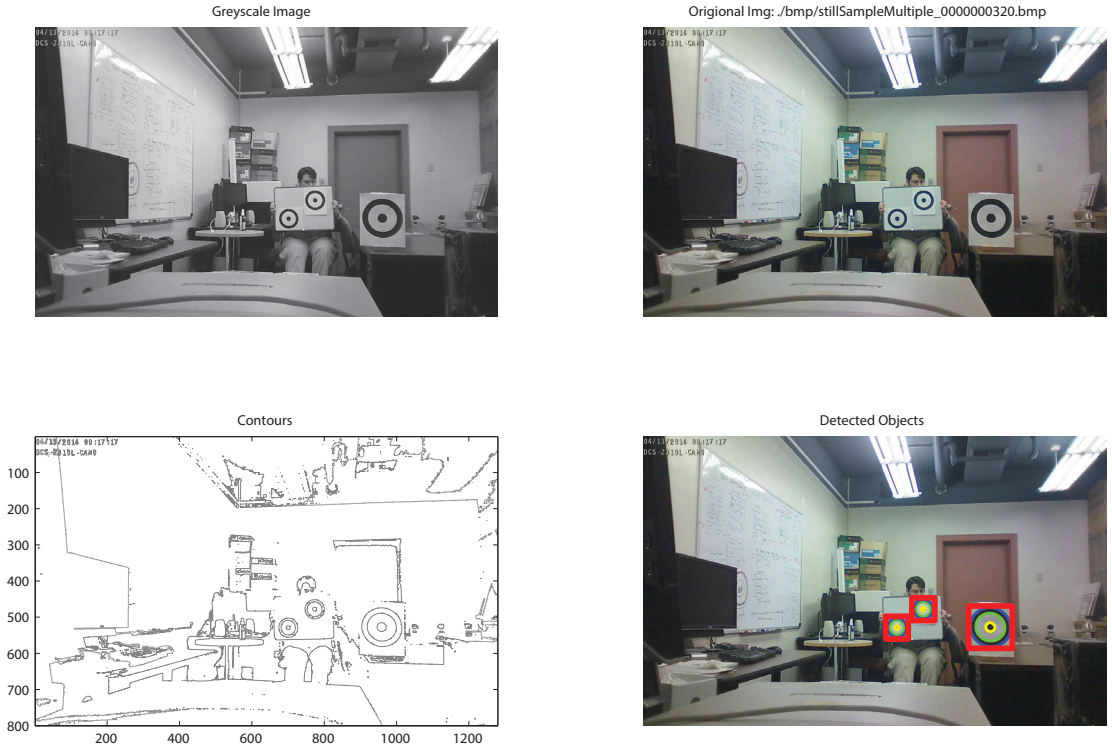


Figure 5.14: Simulation of Contour Based Object Detection

5.12 FPGA Back-end Dataflow Proof of Concept

Due to time limitations, the FPGA implementation of the marching squares algorithm detailed in section 5.10 could not be implemented by the time this paper was written. Instead, a demonstration of the accelerator work flow was developed to show that the general architecture was feasible. The system has the same structure

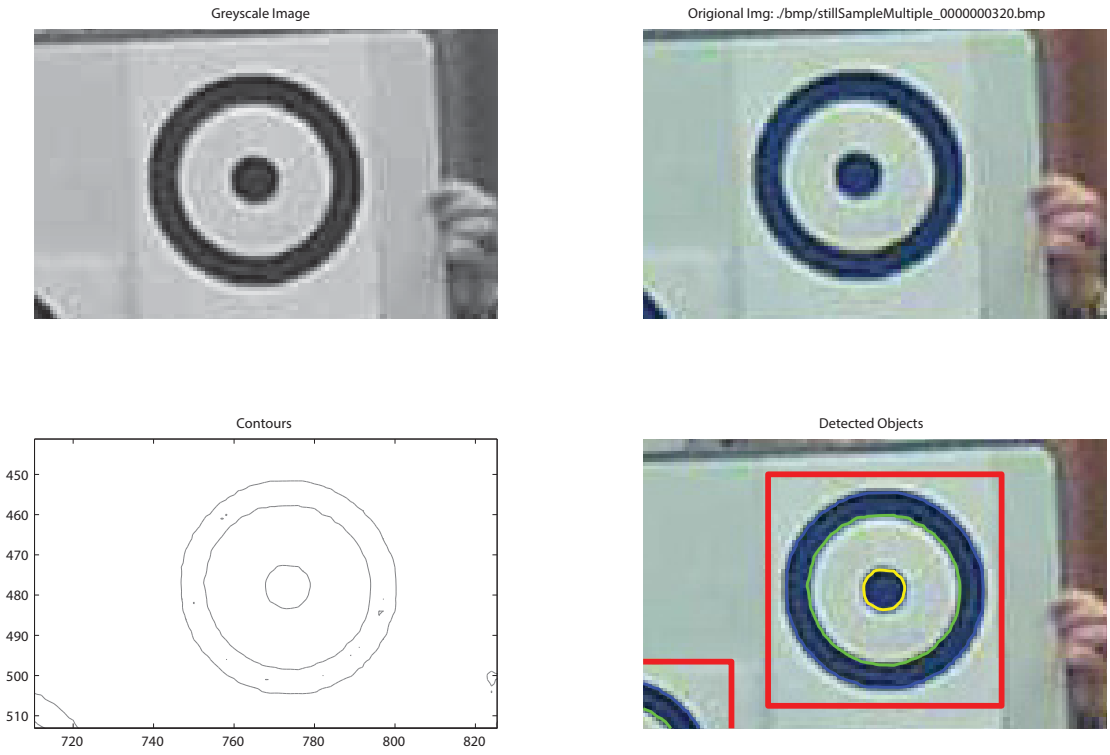


Figure 5.15: Simulation of Contour Based Object Detection - Zoomed

described in Figure 5.3 and Figure 5.5. This demonstration IP took the place of the "Image Processing IP" and demonstrated the ability of the ARM processor and the FPGA IP to communicate via the Control Status Register (CSR) and the shared DDR3 memory. The tasks performed by this demonstration IP are:

1. ARM writes data into shared memory
2. ARM commands the IP (via the CSR) to read the values stored in shared memory
3. IP returns data via the CSR
4. ARM checks data is the same
5. ARM commands the IP (via the CSR) to write data into shared memory
6. IP writes data passed in through CSR into shared memory
7. ARM reads data in shared memory
8. ARM checks data is the same

The hardware implementation was based on many Altera reference designs including the *Golden Hardware Reference Design (GHRD)* [2] [26], the Golden Top top level file [31], the Board Test System (BTS)

[32], the Avalon-MM Master Templates [33], and the Avalon-MM Slave Template avalonMMMasterTemp. The software implementation was based on the Golden System Reference Design [22] which is based on the Yocto Linux distribution and the OpenEmbedded build system.

The GHRD formed the basis for the Qsys system design while the BTS formed the bases for the DDR3 memory controller implementation. The demonstration IP relied on both the master and slave Avalon-MM templates.

Due to some technical difficulties, the demonstration is not yet running completely. A problem may exist in the IP, the driver software, or the DDR3 memory controller. Work is currently underway to debug this issue and bring the demonstration online.

The code and design for the demonstration are given in section 16.3 and section 16.4. This includes both code written by the team, Altera reference code, and modified versions of Altera reference code. The development environment was setup according to the instructions at the RocketBoards.org wiki [34]. The RocketBoards wiki was referenced by Altera as the source for documentation on running Linux on the development kit [35]. Several modifications to the distribution had to be performed to get it running correctly. Since an in depth look at the environment setup is not the focus of this paper, the step by step instructions are not given here. Additionally, the development processed used is more nuanced than can be effectively described in this document. Instructions were collected in the FPGA lab notebook as the project progressed. This lab notebook is appended to this document in chapter 17.

Chapter 6

Identity Engine Design and Implementation

The primary objective of the Identity Engine is to identify detected objects so that they can be cataloged in the database. The time of first detection and the camera that detected it are stored in the database along with object attributes. Once an object has been logged in the database, each subsequent identification will add an additional detection time and detecting camera. These data points are used by the Behavior Analysis Engine to generate normal behavior patterns and recognize anomalies.

6.1 Software Architecture

The Identity Engine uses a three stage software architecture. In the first stage, the identity engine accepts incoming data through a socket connection. Since the server is Windows based, we used a built-in function called Winsock. Winsock provides the ability to communicate via network protocols, such as TCP/IP or IPX/SPX. Once the Identity receives image data from the FPGA via the socket, image processing can begin.

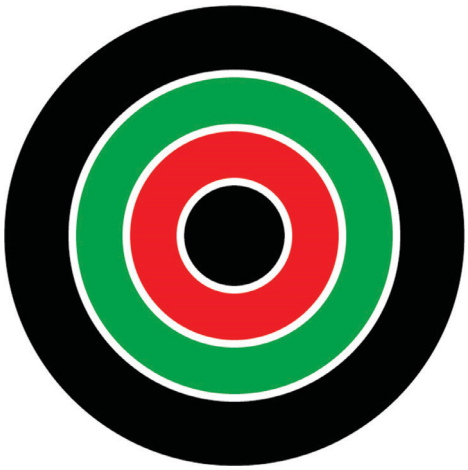


Figure 6.1: Sample Image as Received From FPGA

Image processing starts with determining the size, in pixels, of the cropped image received. After the height and width is determined, the Identity Engine can properly traverse the image array. This is important because Canny Edge detection works on a pixel by pixel basis. Once Canny Edge detection has been performed, the target boundaries are determined.

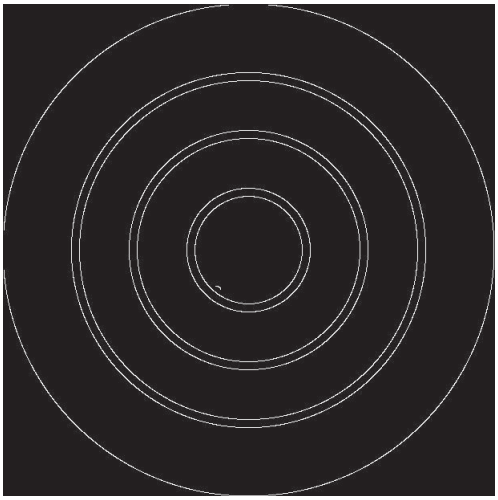


Figure 6.2: Image After Canny Edge Detection

As Figure 6.2 shows, boundaries are marked by binary ones. For the sake of demonstration, the ones are displayed as white pixels. The rest of the image is filled with binary zeroes, which are shown in the example as black pixels. This makes determining the bounding region of the object very simple. In the case of the circular demonstration targets, the algorithm finds the upper, lower, left, and right maxima of the circle. The difference between opposite sides is used to determine speculated radii. Both the speculated radii are averaged to find the radius of the outer bounding circle. Since we are using targets with a known pattern, inner rings are calculated using a ratio to the outer radius.

After establishing the boundaries of the targets inner rings, the Identity Engine works to attempt to identify it. Our target identification criteria are the color of the two ID Bands. In Figure 6.1, the ID Bands are green and red, respectively. To determine the color of each band, the Identity Engine normalizes the red, green, and blue value for each pixel within the established boundaries of each band. Once the normalized colors have been ascertained, they are used to query the database to check for prior identification of this individual. If an individual matches the description, the location and time of this detection are appended to the existing entry. Otherwise a new entry will be created for later reference. To communicate with the database, the Identity Engine uses a tool called Language-Integrated Query, abbreviated LINQ. LINQ is a Microsoft product which allows Windows applications to query SQL databases, ADO.NET Datasets and XML documents. This tool was selected since the Identity Engine is a Windows application and our database is based on Microsoft SQL.

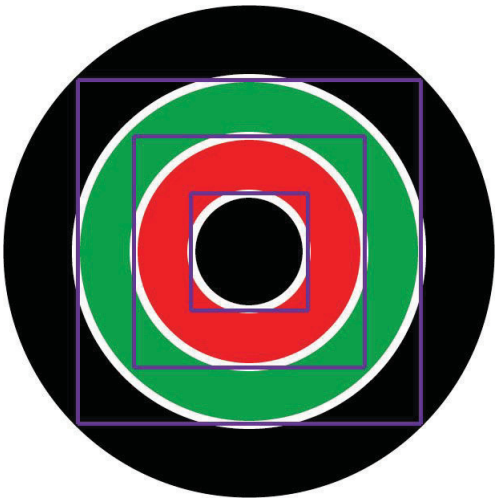


Figure 6.3: Illustration of Outer Maxima and Ratio of Radii (Highlighted in Purple)

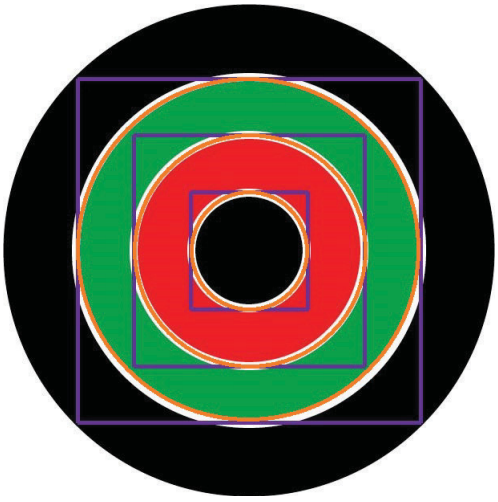


Figure 6.4: Bounding Circles Highlighted in Orange

6.2 Canny Edge Detection

The basis of Canny Edge Detection is three principals: accuracy in finding edges, the detected edge should be as close as possible to the true edge, and the detected edge must only be one pixel wide[36]. It achieves this by using the first derivative of Gaussian functions. The particular implementation used for this project, and the MATLAB implementation, use the thresholds found with these derivatives to generate a binary edge map. A binary edge map is an array of pixels where edge pixels are stored as ones or zeroes, and the rest of the image is the inverse. Which convention is used is a matter of the programmers personal preference. Our implementation used edges marked by ones with the remaining pixels as zeroes to maintain consistency between the final version and MATLAB prototype.

We selected Canny Edge Detection for multiple reasons. It is more likely to produce closed contours than the Sobel method. Perhaps more important, due to our heavy reliance on GPUs, is that Canny is highly parallelizable. Gaussian filters are applied on small groups of pixels. CUDA allows us the ability to apply these operators to multiple groups at the same time. This reduces the overall time required to identify an object.

6.3 Facial Recognition

For the purposes of this proof of concept we decided to forgo facial recognition. Facial recognition is a rich area of research. Simple target recognition provides enough data to verify the function of the Behavioral Analysis Engine. It should be noted that the term recognition encapsulates what we separately refer to here as detection and identification. We refer to the components individually due to the distributed nature of our system.

There are many different algorithms and implementations of facial detection. For the purposes of brevity, I will only discuss one implementation. DeepFace is an algorithm developed and used by Facebook. I have selected to discuss this algorithm for two reasons. First and foremost, it is arguably the most used implementation, since there are millions of users on Facebook at any one point in time. Second, it is purported to be the most accurate algorithm available. In their Conference on Computer Vision and Pattern Recognition paper, Facebook sighted 97.35% accuracy in matching a face between two images. This exceeded the previously most accurate algorithm by 27% [37].

DeepFace relies on a computer science theory called Deep Learning. Deep Learning is a highly complex and rich field of research, and there have already been many publications discussing it in depth. For the purposes of this brief overview, only how DeepFace leverages Deep Learning will be discussed. To identify unknown faces, DeepFace first aligns the facial features to the cameras. This is done by first registering

facial features in two dimensions. The registered features are then mapped onto a generic three dimensional facial model. Once the 3D representation of the unknown face has been generated, it is rotated so that it is aligned with the camera. The resulting image then goes through a series of filters and is normalized. After the post-processing, facial features are used to query the Deep Network of known faces. If the features match a known face, an identity is ascribed to it.

One of the goals of this project is to be modular and allow for integration of future technology. The current implementation of the Identity Engine is designed to be easily replaced with DeepFace or similar system. The only modifications that would need to be made to other sections of the pipeline would be modification of the database schema. All other aspects of the pipeline could be used without modification.

6.4 Testing

To verify the correctness of the overall algorithm, a prototype was first coded in MATLAB. This was due to the pre-integrated availability of all necessary functions in the MATLAB environment. These functions include Canny Edge Detection, sockets, SQL insert and query, and image output for debugging. Once the results were verified to be correct, the algorithm was translated to CUDA C/C++ and Windows functions. Afterward translation results were checked against MATLAB and the expected output.

Chapter 7

Database

The database is an essential back-end component for the system because it provides not only a place to store the vast quantities of data that the system will be generating, but it also provides a connection and method of communication between components of the project. The database interacts with the Identity Engine by receiving new images, identities and events of objects moving within the camera frames, as well as providing images and identities to compare against during object recognition. The behavioral analysis engine then pulls the new information on events and individual and process them, creating anomalies. Finally the user interface is consistently polling the database for new information on events and anomalies to display to the monitoring staff.

7.1 Setup

The database was set up on the server using Microsoft SQL Server Management Studio. Since this is a proof on concept project, many of the standard database components were ignored. These ignore components include audit support, database permissions, access levels, and security.

7.2 Schema

The schema was the primary area of development for the database. Since the database was used as a method of communication for many components of the system, the schema had to match the changing requirements of each component. Originally, the schema was designed before the implementation of many components of the project; this was done to create some structure in communication. As seen in Figure 7.1, the approach to the initial design overly complicated to satisfy all possible situations. However as components of the project began to implement the database connections, it was easy to realize change was needed in the design.

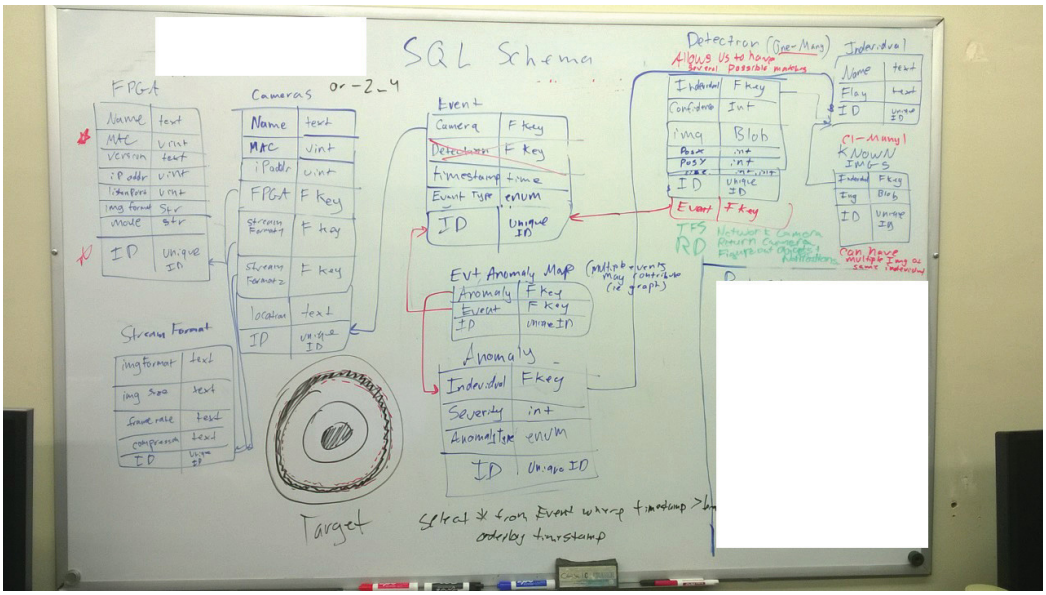


Figure 7.1: Original Draft of the Database Schema

7.2.1 Schema Modifications

The schema was modified multiple times during the implementation of the project and is likely to continue to change even after this thesis has been completed. The first major modification to the schema was the removal of the FPGA and Stream Format Tables. Since the FPGA will not be communicating with the database by the end of the project this information will not be used. In the situation that this system becomes a commercial product, or the FPGA had implemented a database connection for setup, these tables would be re-added in order to provide that information. The next major schema changes were due to the changing of behavioral analysis algorithms. The detection and EventAnomlyMap Tables were removed to simplify how the tables were being processed by the behavioral analysis engine. Instead the schema migrated to the Camera and DetectionEvent table structure shown in Figure 7.2, where only one event would be created for and individual entering and exiting the cameras frame. Finally the CameraStat table was added to easily save and load the statistical information from the database to the behavioral analysis engine.

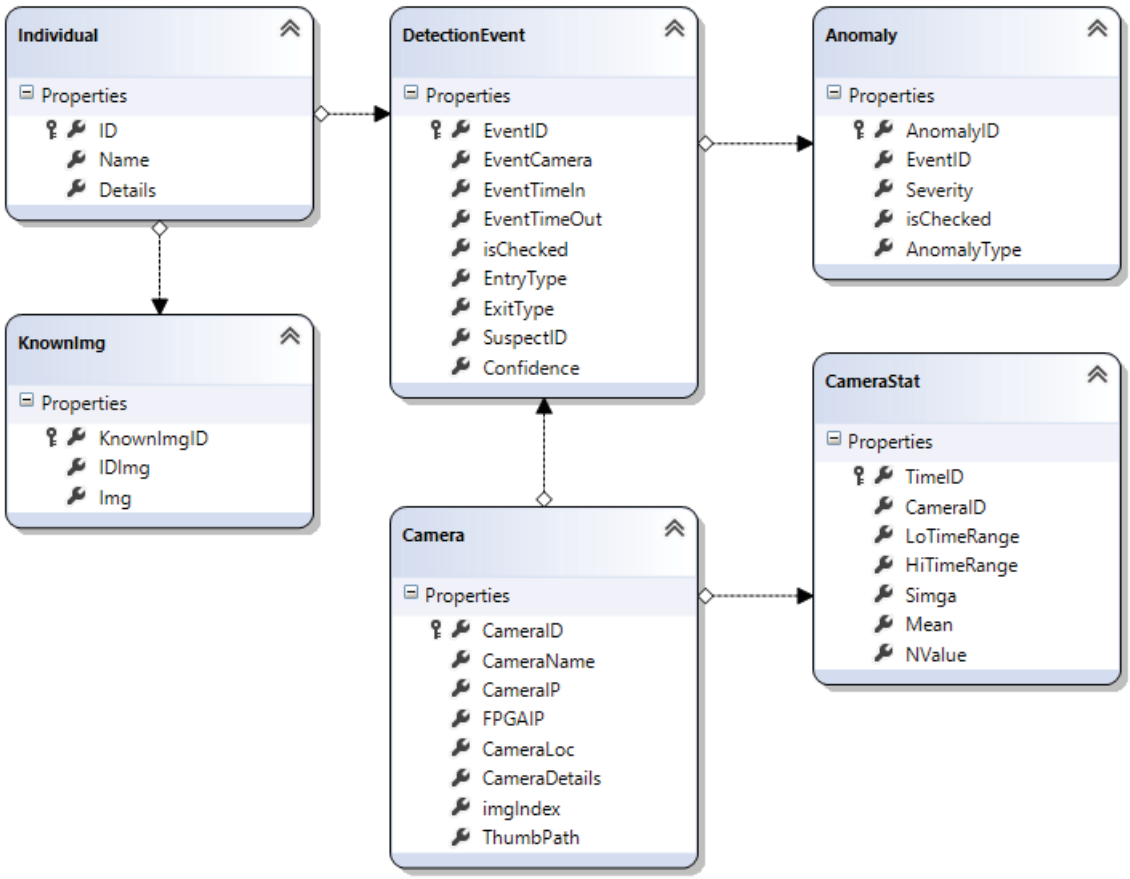


Figure 7.2: Final Schema Design

Chapter 8

Behavioral Analysis Engine Design and Implementation

The Behavioral Analysis Engine is an essential component to the system it because provides real time analysis of the data being produced by the other components of the project. During the conceptual creation of the project many ideas were produced about how to actually detect behavioral anomalies, which were distilled down to two major types of analysis. The original idea behind the behavioral detection of the project was the use of Directed Weighted Graph Analysis, but due to complications and constraints it could not be feasibly implemented. This limitation was due to many factors including time, budget and available technology. Instead Frame Duration Analysis, subset of the original idea, was implemented.

8.1 Weighted Graph Analysis

When the concept of this massive project was first presented it was primarily paired with using Directed Weighted Graph Analysis for behavioral detection. This behavioral analysis algorithm planned to take advantage of a multitude of information generated from the system, including providing analysis by time, location, pattern and identity.

Graphs are a popularly accepted mathematical idea that stems from what is known as contemporary graph theory. Graph theory involves the construction of a graph using nodes and vectors using specified rule sets. The graphs in the implementation would represent a building with IP Cameras as nodes at key locations with individuals creating vectors by moving between the cameras. The directed graph rule set states that a vector from Node A to Node B is independent and unique from a vector in the opposite direction. This rule set directly applies to the implementation because the analysis requires a person's exact location and, without a direction associated with vectors being declared, the location becomes ambiguous. This ambiguity becomes even more of a problem when including analysis based on time and identity. The weighted graph rule set states that a vector has a weight associated with it. This applies to the implementation because with specified

weights we can then assign probabilities to each path taken, and we can then modify the probability weights based on frequency, time, identity and number of connected nodes. These weights can also provide pattern based analysis especially when examining a graph pertaining only to individuals.

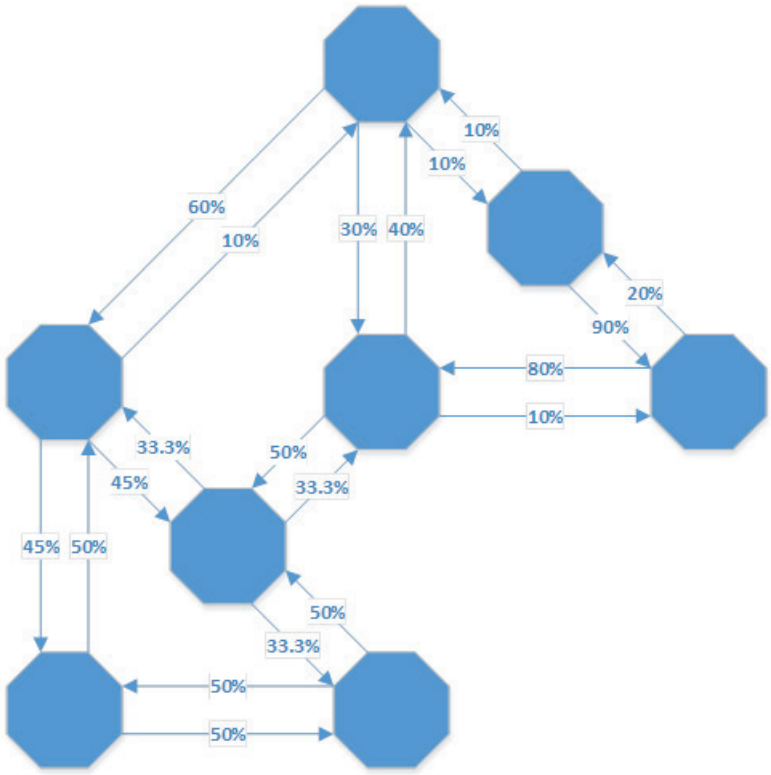


Figure 8.1: Sample of a Directed Weighted Graph

In order to show the possible capability of the directed weighted graph analysis a series of sample situations have been devised. Initially a person would walk into the view of the cameras and be detected, recording the relevant information into the database. After the database has saved enough information about the cameras it can begin to provide analysis on the actions of individuals. As a recognized person begins to move throughout the network of IP cameras, an anomaly can be generated if they move into an infrequently traveled area (taking a path with low weight). The system would save all the information on vectors including information regarding individual creating these vectors, this allows the creation of graphs for each individual or individually based analysis. These individually based graphs can then be used to further enhance the accuracy by examining personal movements with algorithms based on frequency and patterns. Another situation includes the generation of graphs based on time if an individual is entering the buildings at night where no night shift exists, an anomaly is generated. This type of analysis can also be expanded when using individual based graphs to recognize night shift employees based on their time patterns, creating even more cases for exceptions or confidence in anomaly detection. On top of all of these methods we can also add pattern based

analysis, which can also be applied system wide, individually, and to different times of the day. However pattern based analysis can also create unique forms of anomaly detection like the building entrance example, which says that if a camera is posted on every entrance and an individual appear inside a building without passing through an entrance creates an anomaly.

Directed Weighted Graph Analysis is still one of the most innovative ideas behind the project, but unfortunately due to various constraints it was not able to be implemented in the project. In order to implement this idea it required large amounts of already generated data. This lack of generated information became one of the primary constraints. During the construction of the project, parts of the system were implemented individually in enhance modularity. This proved a good strategy, but has delayed all data generation. This delay of information made this type of behavioral analysis effectively impossible to accurately implement in the time frame. Another major constraint for this analysis was the budget. This type of graph analysis requires many locations and nodes to be accurate and tested thoroughly, requiring many expensive IP Cameras. All these ideas involved with graph based analysis proved much too ambitious for the scope of the proof of concept project. In order to complete this project some type of behavioral analysis had to be generated, so we chose to only pursue Frame Duration Analysis which was originally going to be implemented into the project as a subsection of the anomaly generation.

8.2 Frame Duration Analysis

Frame Duration Analysis was originally a small component of behavioral detection software that the project was intended to run. This idea came from two potential applications for the project: security and safety. The intention of this algorithm is to identify anomalies based on the speed of an individual. For example, in a hospital if a patient collapses in an abandoned hallway or if a criminal is attempting to escape a building quickly the system would detect it. Frame duration analysis is actually quite simple. It takes the duration of an object within a camera frame and compares to the duration of previously seen objects. In both of the described situations the movement of the individual would quickly become anomalous because they are moving much slower or faster than the average human.

8.2.1 Implementation

Frame Duration Analysis uses a statistical model known as a normal distribution, shown in Figure 8.2. This can only be done because statisticians have determined that human walking speed is associated with random variables that approximately correspond to the normal distribution [13]. In order to compare the movements of individuals to each other, the system creates an Event for each object that appears in a camera. An Event contains lots of information including the individual, the camera, and timestamps of both appearance and

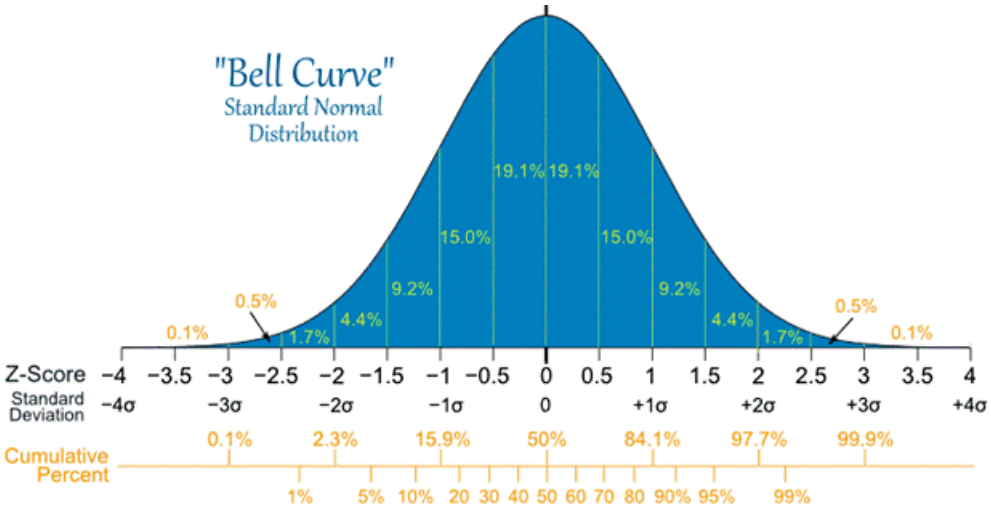


Figure 8.2: Sample of a Normal Distribution (Image Courtesy of Mathisfun.com [3])

disappearance. Using the change in time between entry and exit of an individual in a camera frame, the duration is calculated. Since the cameras do not move we can assume that the distance is constant, therefore we can use the duration as a relative speed variable. In order to generate anomalies the system saves the statistical information for each camera in the database. When an event is completed, we then take the saved camera information and calculate likelihood of this event occurring naturally. If this event has a 5% or less chance (two standard deviations from the mean) to occur it is assumed to be anomalous. The camera statistics are then updated by including the new Event information into the statistical model.

8.2.2 Implementation Complications

The simple implementation of Frame Duration Analysis as it is described above contained many assumptions and oversights. During the conceptual creation and implementation of the behavioral analysis engine many of these issues were identified and solved, based on various theoretical situations. The first situation we encountered was if a person collapses for extremely long period of time (a collapsed patient). Including their statistics in the model would skew the data and provide more inaccurate results. We have since included checks where if the individual falls within the most unlikely 0.3% chance (three standard deviations from the mean) of occurring the statistics are thrown out. Another of the biggest problems with the implementation of this behavioral analysis is the non-live factor of producing anomalies. Since the anomalies require that an Event be closed before generating the anomaly, monitoring staff would be pointed to a camera where the anomaly has previously occurred. In order to solve this problem, active events are polled against the slow end of the anomaly threshold constantly. However, it is impossible to generate an anomaly for the fast end of the anomaly threshold before the individual leaves the frame in the current implementation. Another major

problem with the system is that while reducing the amount of information needed before analysis, a normal distribution still requires at least 36 data points to be relatively accurate.

8.2.3 Constraints and Assumptions

The scope of the project, and focus on other components has forced us to make a few assumptions that do not hold in reality. The most important of which is the assumption of consistent distance within a camera frame. Simply put, we ignored the depth dimension of the camera video feeds. This can cause major problem in practical applications because an individual further from the camera will take longer to exit the frame than individuals much closer to the camera. Ideally with more time this could be implemented solving many of the biggest problems with behavioral detection by allowing real time anomaly generation and more accurate results. Since the implementation of this system was intended for indoors we have discarded the possibility of the use of vehicles, which could potentially cause many false positives in the anomaly detection.

Frame Duration Analysis is much easier to implement than Directed Weighted Graph Analysis because many of the previous constraints on budget and information no longer apply. Directed Weighted Graph Analysis has a macro based approach, viewing the building as a whole network of IP cameras, whereas the frame duration analysis only applies to the individual cameras. This allows the behavioral analysis engine to be tested without having large budgetary needs. The other major constraint associated with Directed Weighted Graph Analysis was the lack of generated information. Since Directed Weighted Graph Analysis is a macro based approach the amount of information required to generate accurate results was large. Frame Duration Analysis only requires 36 data points in a single camera frame to begin generating anomalies.

8.3 Exit and Entry Type Analysis

Exit and Entry Type Analysis was invented during implementation to solve one of the biggest problems associated with behavioral analysis, but after further consideration became an essential component of unifying the theoretical model of the project. This type of behavioral analysis examines the locations of the entry and exit of an individual in a camera frame.

A major flaw of Frame Duration Analysis is that it fails to account for the position of the object when it enters the frame. This can be a major problem because individuals that become obscured mid frame are likely to generate anomalies and negatively influence the accuracy of the system. In order to fix this problem a case was created to state that if an individual was detected to entering in the middle of the camera frame and anomaly would be created. Since the detection algorithms cannot guarantee the consistent detection of an object a buffer was created. This buffer allows for the brief lapse in detection without assuming the object has left the frame. Without this buffer any lapse in detection would generate anomalies, which proved to be a

common occurrence during tests.

The idea behind Entry and Exit Type Analysis could be expanded to incorporate benefits in both Frame Duration Analysis and Directed Weighted Graph Analysis. During the setup phase of the system the common entries and exits of individuals from a camera could be identified and used in graph analysis. As individuals travel between the cameras the exit position in a camera could greatly increase the probability the system could predict the next camera in which they will appear. For example an individual exits one of the cameras on the left hand side where a hallway exists only leading into the next camera, but does not appear, an anomaly can be generated.

Chapter 9

The User Interface

The User Interface is the only place for interaction between the user (the surveillance staff) and our system. It is built to engage the user by following modern design principals and relay the important components of our project efficiently. The User Interface design started with a mock up design seen in Figure 9.1 for conceptual confirmation, and placement of the functional requirements.

9.1 Mock-up Design

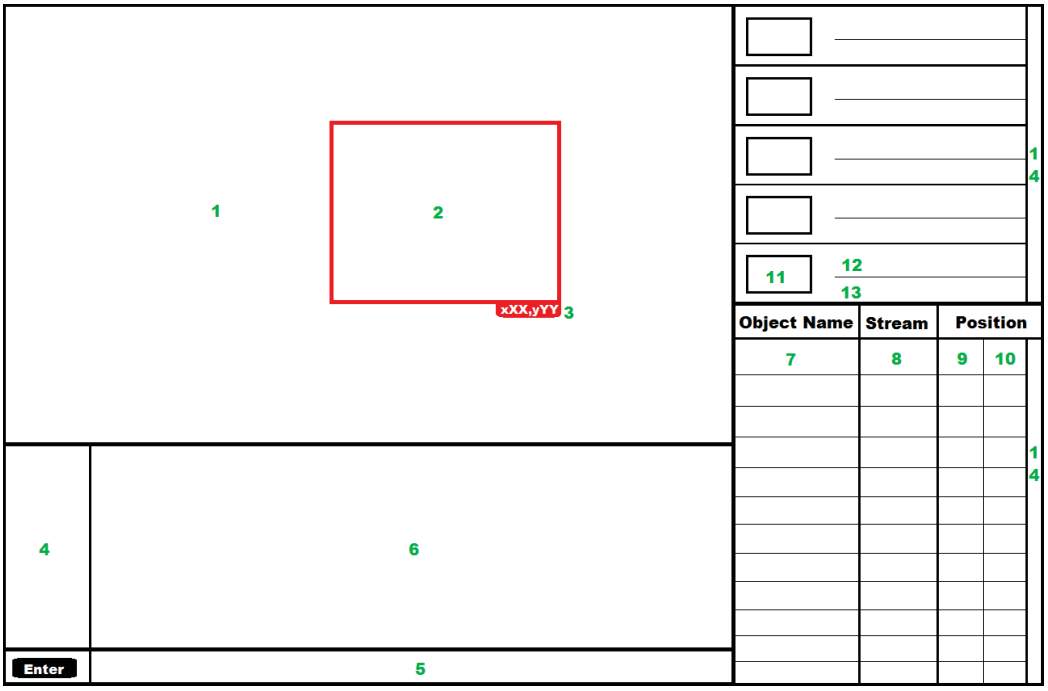


Figure 9.1: Sample Layout of the User Interface

The mock up interface was designed to incorporate modern design principals used in other popular User

Interfaces. The components of the User Interface were placed in similar locations to other popular interfaces, primarily having the text box or log at the bottom of the screen and the menu systems on the right or left hand sides for the screen. Another modern principal commonly used in photography is the rule of thirds [38], which states that the primary focus of the screen should have its center approximately a third of the way into the screen and was done for the video feed. This is a list describing the all components of the User Interface, shown in the Figure 9.1, and describes how each interacts and adds to the User Interface:

1. Video Steam - A simple video stream that displays the video feed from the IP Camera unmodified. This is an essential component to all surveillance programs.

2. Identified Object - Displays the most interesting/abnormal or selected object on screen. Clicking on the object in the object view on the right will switch the streams and highlight the individual on screen. (This was deemed an additional feature and could not be implemented in time.)

3. Position on video - Displays the position of the object in the bottom corner. (This was deemed an additional feature and could not be implemented in time.)

4. Time Stamps - These time stamps simply act as a log reference to help determine the frequency of actions.

5. Text box - This component acts as an input for the user in order to enhance the information recorded. (This could potentially be expanded to intake commands and provide additional information.)

6. Information Log - Displays all information about camera statistics, input text, setup and management notifications, and most importantly provides notifications of behavioral anomalies.

7. Object Name - This field displays object name and it should have the ability to rename objects. This should be linked with the database to save inputted information and load new information as it appears. It is part of the Object Row.

8. Stream Number - This field displays the Stream number the object is found on, and is not editable. It is part of the Object Row.

9 & 10. Position - This field displays the X and Y position of the object on the screen, and is not editable. It is part of the Object Row.

11. Thumbnail - This field displays a static image of the camera stream, and should be editable to the user. It is part of the Stream Row. The Stream row can be selected by calling a function that changes the stream selected.

12. Stream Number - This field displays the stream number, and is not editable. It is part of the Stream Row.

13. Location & Description - This field displays the description of the camera location, and is editable to the user. It is part of the Stream Row.

14. Scroll Down bar - Scrollable bars to allow for multitudes of objects improving scalability.

9.2 Requirements

This project will be expected to meet certain standards set forth by Santa Clara University; requirements have been determined in order to ensure the success of this project. However since this project is a collaboration between computer and electrical engineering the term "requirements" has been blurred. Also this is a proof of concept project only further confusing the result of this project. Unlike most traditional types of engineering, computer engineering views requirements as negotiable. It considers only the bare minimum to be truly required and the rest are extra features. In a computer engineering project features can greatly affect the outcome of the project for improving performance, ease of use, accessibility, and other widely scalable non-functional requirements.

9.2.1 Functional Requirements

A functional requirement is a component or feature of our system that either does or does not exist. As our project is highly modularized each section of system will have its requirement analyzed independently after they have been described in the design of our system. The requirements that are described in this section are for only the User Interface and are meant to help us judge whether or not the User Interface component of this project was a successful venture.

- Video Surveillance Feeds - The User Interface must be able display live security video feeds from the IP Cameras to the monitoring staff in real time. This includes the adding, swapping and loading of video feeds.
- Behavioral Analysis - The User Interface must be able to visibly notify the monitoring staff of behavioral anomalies in real time. In order to do this it must be able to establish a reasonable protocol for connecting to the database.
- Auditing Support - In order to ensure that the users are not abusing the power and information provided by our system, the User Interface must provide some way for the users actions to be saved in the database to be audited at a later date.
- Information Correction - The User Interface must be able provide some way for the users to change the information saved in the database. Our system is not guaranteed to be accurate so the users should be able to change inaccurate information. This information can also lead to corrections in our other systems to improve accuracy.

- Management & Setup - The User Interface must provide information and tools to easily setup and manage the system.

9.2.2 Non-Functional Requirements

Non-functional requirements are characteristics of the final project that cannot be classified as a feature or function, but rather are scalable and vaguely defined.

- Usability - The User Interface must be clean, intuitive, and easy to use for the monitoring staff.
- Modularity - The User Interface must be easily modifiable in case other components are replaced.
- Accuracy - The User Interface and the system must be able to provide some level of accuracy in our analysis, in order to make this a efficient product for the monitoring staff to use
- Security - The User Interface must provide easy access to important information while still remaining secure.

9.3 Implementation

After the components of the mock-up design had been established the actual implementation was able to take place. Just like any other project the idea put forth originally had to be re-evaluated and made more practical to implement. The functional requirements that were outlined during design of the mock up interface had to be prioritized forcing some features to be cut (e.g. the object information box in the middle of the video feed).

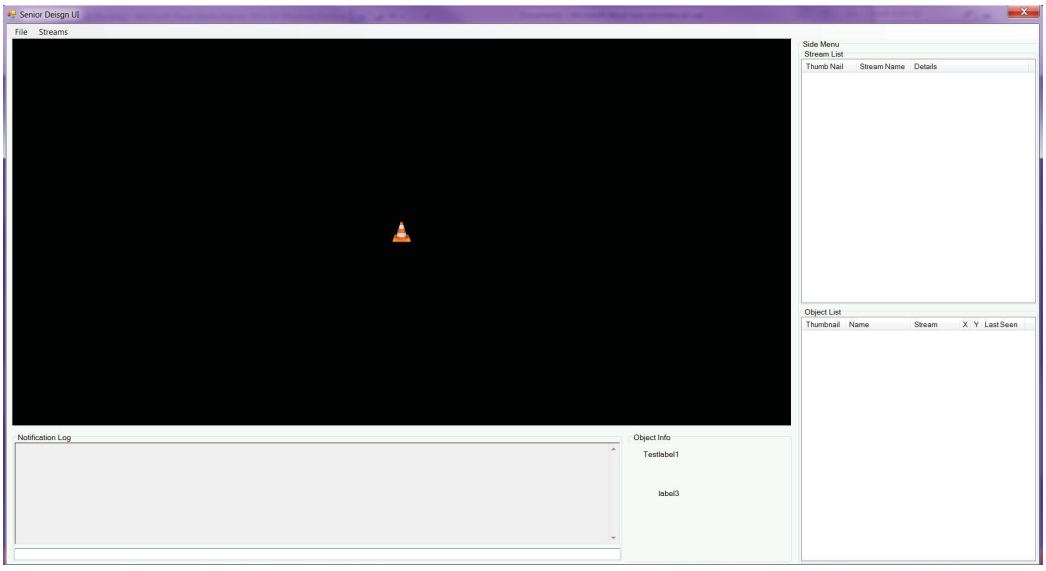


Figure 9.2: Recent Layout of the User Interface

The start of the implementation began with a massive struggle to effectively understand the tools we were generously given by Microsofts Dreamspark Program, primarily using Visual Studio Ultimate for the User Interface. After struggling for some time on the basic User Interface implementation, progress started on the primary focus for the screen - the video feed from the IP Cameras - because of its high priority. The video stream requires a connection from the User Interface to the IP Camera. In order to do this a series of calls must be made over the network creating a RSTP (Real Time Streaming Protocol) stream. While attempting to setup and understand the RSTP streams, the discovery of an API (Application Programming Interface) for RSTP streams allowed the bypass of a large part of this implementation. This discovery was a major boost to the progress of the User Interface because it saved hours of implementation and understanding of the complicated base level, as well as providing base level of implementation and accomplishment. This API was provided by VideoLAN, the developers of VLC Player, who created a VLC plugin (with RTSP support) for ActiveX in the .Net language libraries [39].

After the implementation of the video feed, the next largest priority was the implementation of the menus, lists, and other navigational components. This step of the implementation began with the placement and visual design of the User Interface by using and understanding the visual elements provided in Visual Studio Libraries. This level of implementation can be seen in the screenshot of the interface, Figure 9.2, as everything was visually in place but had no functionality. The next few steps in the development of the functionality of the User Interface became a mess of new ideas and backtracking, as the limits of actual functionality were realized. The current implementations of the next few steps to the User Interface have been outlined below.

- RSTPStream - RSTPStream is a class created to abstract away the elements required to setup a RSTP stream through VideoLANs API and to properly display the information on screen. This class has the necessary getters, setters, and constructors, but has very minimal implementation because of its use simply as a layer of abstraction.

- ListView - This is a built-in class in the .Net libraries but, in order to properly display the information in the right hand side of the User Interface, a firm understanding of its implementation is required. There was lots of testing done with custom modifications to the standard ListView class but none accurately portrayed the information and the solution was eventually found in the default class in obscured references to something called the Grid View.

- Globals - Globals was a class created in order to allow access to specified elements from all classes primarily from the main User Interface class Form1 to the Addstream class where the RSTPStream information needed to add and double check classes.

- AddStream - This is a secondary form used to popup when the user attempted to add another stream to the list. This has its own functions and methods enabling the adding of RSTPStream to the Database and the Globals class.

- Thumbnails - From the beginning thumbnails were wanted to be automatically added when a stream or object was added to their respective lists. Unfortunately these attempts were abandoned to be replaced with still images to save time.

The next important step was the construction and setup of the database; this can be seen in Chapter 7 the database implementation chapter. Since all of our software was provided by Mircosofts Dreamspark, synergy was created between programs which allowed the database schema to be directly imported to the User Interface. This importation allowed the simple creation of many new classes and increased the speed of implementation during the rest of the project. Ideally many of the previous created classes could be modified to suit this schema to increase cohesiveness and future modularity but time constraints have limited this. The database connection also allowed the User Interface to directly pull the information saved in the database to be relevant information in behavioral analysis. The behavioral analysis requires that the information generated in Behavioral Analysis Engine be updated in real time to the User Interface. In order to do this without creating major stalls in the User Interface a second thread was created to poll the database on a second by second basis. This is where progress was halted in order to meet other deadlines for our project.

9.4 Technologies Used

The technologies used in this User Interface are listed below.

- Microsoft Visual Studio Ultimate 2013

- Microsoft LINQ Libraries

- VideoLAN ActiveX Libraries

- Team Foundation Service

- Git

9.5 Risk Management

Throughout the planning of this project we also took into account the risks involved, and the prospect of not finishing our design. In order to mitigate the results of our risks we rated each one based on severity and impact and created mitigation strategies. The likelihood of each risk occurring is outlined in Figures 9.3

under the P or Probability column and the difficulty of the solution to the risk is seen in the S or Severity column. The I or Impact column is the product of the other two columns and represents how greatly the risk will affect the project.

Now that the project has been mostly implemented this table, Figure 9.3 was relatively accurate, however not completely. The major differences are that the problems from new technologies were much more significant than expected and the electrical integration had been delayed repeatedly. Also, unexpectedly most software projects have rapidly changing requirements, whereas this project has had relatively no changes to requirements because this is a proof of concept project. This lack of changing requirements might also be because the database stands between the User Interface and the rest of the project so any changes can be incorporated in the database schema instead of the User Interface implementation.

Project Risks	Consequence	P	S	I	Mitigation
Electrical Integration	Delayed end to end functionality	0.5	6	3	Weekly Meetings, Allow for extra time
Changing Requirements	Incorrect project functionality	0.4	7	2.8	Design for Change
Miscommunication	Delayed end to end functionality	0.5	4	2	Weekly Meetings
Sickness	Delayed checkpoints, Specific Modules Delayed	0.2	5	1	Weekly Meetings, Long Term Planning
New Technologies	Time delays	0.2	5	1	Plan for Extra Time, Weekly Meetings
Data Loss	Major time setback, Unable to Meet Deadlines	0.1	5	0.5	Maintain up to date Backups

Figure 9.3: User Interface Implementation Risks

Chapter 10

Test Plan

Similar to the development of any system, a test plan was put in place at the beginning of our project in order to ensure a better final product. Since this project was a proof of concept, many the quality concerns were temporarily ignored in order to further implementation and attempt to get a working pipeline. Since many of the components of our project are still in development due to the massive scope of our project, extensive testing has not been conducted.

10.1 Unit Tests

Unit testing will be done throughout the implementation of every module contained within the project. Considering the independent development of each module, the developer of a module is responsible for the unit test of their components.

10.2 Integration Tests

Once modules have progressed far enough to be near independent completion, the connections between components will be established and tested. Integration tests are much more important to the construction of this project because these tests can be used to help prove the concept behind the project.

10.3 Test Media

For the video processing systems (FPGA back-end and Identity Engine), a multi-tired test plan was developed in order to assess the suitability and reliability of those subsystems. The test plan is described as follows:

1. Record video from actual cameras used in the project. This footage will be used to provide a consistent stimulus to all test cases.

2. Develop a simulation of the image processing component in MATLAB.

3. Run the simulation with the recorded test video, noting any false positives and false negatives.
4. Fine tune the simulation and retest until pass rates are within an acceptable range.
5. Develop the FPGA or GPU implementation of the system and run the hardware implementation with the same test video, checking for false positives and false negatives.
6. Compare these results to those from the simulation to ensure consistency.
7. Debug implementation, if possible, and account for discrepancies.
8. Perform a live stress test with many moving objects in an attempt to break the implementation.

10.3.1 System Testing

Finally, after all the components have been connected, system testing begins to ensure the entire system works as a unit. This consists of a full end-to-end test of the system with both recorded and live video.

10.4 Test Results

Due to the limited development time, extensive testing has not been conducted up to this point. However, simulations of the FPGA back-end have been run against test video captured in the SCU Image Processing Lab. It was determined that object detection was reliable when the objects were close to the camera, but began to become less reliable as the object reached the end of the laboratory (approximately 30 ft away). It was determined that this is likely due to the tolerance parameters on the algorithm being too strict. It is expected that further tuning of the parameters will result in improved stability and performance.

Chapter 11

Ethics

11.1 Societal Implications

Most people feel that they have a right to their own privacy. The Bill of Rights in the United States protects the privacy of its citizens from the government with the unreasonable search and seizure clause. While people are not afforded quite the same level of protection under the law when in public, a certain amount of privacy is expected in our society. Up until recently, having some relative degree of privacy in public was a reasonable assumption. In the years preceding the widespread adoption of CCTV systems, people could move throughout the public sphere and experience some privacy. Even after institutions began installing their own CCTV systems, it was still impractical to track every action of an individual even with monitoring staff. By inserting identification technology and behavioral analysis into networked CCTV systems, the project removes this last limitation to easily tracking the interactions of an individual. Without controls, this system could be hazardous to our society; it would be easy for someone with access to the system to put a tracker on a specific individual and receive updates on their interactions. For example, the system could leave subjects vulnerable to stalking by individuals with access to the system if safeguards to prevent this type of misuse are not put in place.

In order to address the privacy implications of the system, it was designed with limits placed on the types of interactions and amount of information staff can access. In a commercial implementation of the system, the monitoring staff would be able to view video feeds just as they do today. The video streams would be augmented to show what has been detected as a face and if it has been successfully identified. However, this view would not display any identification information about the detected individuals unless they exhibit a behavioral anomaly and the severity of that anomaly is considered high enough. One such scenario would be if a patient in an assisted living facility collapses. In this case, information such as location, name, and medical conditions would be disclosed to the monitoring staff to aid in helping the victim. Any use of the system to locate or back-trace individuals movements would require an individual's supervisory approval.

Any such requests and approvals would be logged by the system and would be subject to auditing to help prevent abuse.

While there are privacy concerns with a system like this, there are some aspects of a smart CCTV system which can actually correct injustices that are present in our current society. Because of the way this system was architected, alerts are only created based on behavior that deviates from a model built off of prior observations by the system. Unlike human monitors who are susceptible to race, gender, and other biases, the computer is not. This can lead to more equal and non-discriminatory utilization of CCTV systems in security applications.

11.2 Project Development

One of the biggest concerns and ethical implications of this project is the safety and security of people being monitored. Depending on the application, there may be different expectations of the performance of the system and consequences based on those expectations. For example, in a security application, the system may act in parallel with trained monitoring staff. The system would be able to alert them to incidents that it believes are suspicious. However, the monitors should still be attentively looking for incidents the system missed. In a medical application, there may not be staff monitoring the camera feeds in real time. If a patient collapses and the system does not catch it, there is the potential for the patient to die. A reality of behavioral analysis is that it is not entirely accurate or precise. In order to maintain the potential positive impact of the project, the team must effectively communicate the limitations of the system. However, if customers do not fully understand the limitations of the system, they may discontinue activities that catch events the cameras miss. The project is meant as a companion tool rather than a replacement for monitoring staff.

Another major concern is the privacy, security, and storage of personal data. Data anonymization and retention, specifically, are common contemporary issues. Many companies store this information in order to make more informed business decisions and dynamic marketing strategies. The project maintains an isolated database of personal information entirely generated by the system to ensure data integrity. In a commercial implementation, this system will restrict access to the data by only displaying information to authorized monitoring staff when the system believes it is important, and by tracking all other requests for information. The system will make it easy for the customer to configure the data retention policy to align with the organizations documented standards.

11.3 Team and Organization

With the project focusing on such a sensitive technology as CCTV, the need for personal and team ethics take on especially high importance. All of the team members understand that, while the project can have profound social benefits, it can also be used for more nefarious ends if the ethical issues are not adequately considered. As such, each team member has the responsibility to consider the ethics of each design decision. If a team member believes that there is an ethical issue that is going unnoticed or unaddressed, they have the responsibility to bring it up with the team and advisors. Issues are discussed within the group and with advisors so that the most ethical path is taken. If a team member feels that the system no longer fulfills its goal of benefiting society and is instead doing the opposite, they would be justified in leaving the project.
Team members understand that their actions not only reflect on themselves but also on the group as a whole, the advisors, the university, and supporting corporations. As such, for each action, the team understands that it must consider the reputations of all stake holders and how a particular action will reflect on them. It is also possible that the team will be asked by one of the other stakeholders in the project to make certain design decisions. It is the responsibility of the team to consider the ethics of such requests. If a request appears to be ethically unsound, the team must discuss the request internally as well as with advisors. If an ethically appropriate course of action cannot be found to fulfill the request, the team must notify the stakeholder of their decision not to go through with the request and its reason for doing so. That way, the integrity of all parties involved is protected.

11.4 Summary

Like all CCTV technologies, this project has several points of ethical concern which must be addressed. The project runs the risk of violating individuals privacy by tracking their interactions within a monitored facility. In order to mitigate this issue, the development of strict controls were considered in thee design of the system. The system actually has the potential to provide more ethical and non-discriminatory treatment of subjects than current human monitored CCTV systems. The design and development team must act responsibly throughout the project as well as ensuring the future use of this project is ethically oriented. The customer will need to be instructed on the proper use of the system while prevented from freely accessing personal information stored within it. The customer must also be informed about the limitations of the system in order to make informed decisions on how to properly integrate it into their environment. The project, while envisioned to have a positive ethical impact, must be closely monitored to ensure the goals of the system are achieved.

Chapter 12

Environmental Impact and Sustainability

12.1 Environmental Impact

Since this project is a form of computing system, it relies heavily on integrated circuits and copper for interconnects within printed circuit boards. The life cycle assessments of some components are presented to give some sense of the environmental impact of the electronics that make up the system.

12.1.1 Semiconductor (ICs)

According to *Life-cycle Assessment of Semiconductors* by Sara B. Boyd of the University of California, Berkeley [40], the total energy consumption per die (chip) of a semiconductor produced using the 45 nm process is 1699 MJ/die. Of that total, 1593 MJ/die is use, leaving 106 MJ/die in embodied energy. The same paper states that the total Global Warming Potential (GWP) per die of a semiconductor produced using the 45 nm process is 47 $kgCO_{2eq}$/die with 43.2 $kgCO_{2eq}/die$ of that from use. This leaves 3.8 $kgCO_{2eq}/die$ from production.

12.1.2 Copper

According to the *Embodied energy and embodied carbon (ICE)* database by Dr. Craig Jones [41], the embodied energy of copper manufactured in bar or sheet form in the EU is 42.00 MJ/kg. Its embodied CO2 is 2.60 $kgCO_2/kg$ and 2.71 $kgCO_{2e}/kg$.

12.1.3 Reducing Embodied Energy and Greenhouse Gas Emissions

Since this is a proof of concept project, it mainly uses off the shelf development kits for the hardware platform. No custom PCBs are being manufactured and no ICs are being fabricated. There is very little that can be done at this stage to reduce embodied energy and greenhouse gas emissions. However, should this project progress

and custom PCB be created, there are some measures that can be taken to improve the environmental impact. Since the system uses ICs from Altera, we have no control over the production process of these devices. However, we can select the IC with the smallest die which fulfills our needs. The smaller the die, the lower the environmental impact per die. A much more practical way to reduce the environmental impact of the project is to make the PCBs as small as possible with the shortest copper interconnects possible. Not only are short interconnects good for signal integrity, but they use less copper and thus lower the embodied energy and greenhouse gas emissions of the custom hardware components.

12.2 Sustainability

Since the project is a proof of concept system, the sustainability of the prototype was not of primary concern. However, ecological protection, coupled with economic considerations, were taken into account in the design of the overall system architecture. The system does require the purchase of FPGA Back-end boards for each camera as well as a server to contain the database and run the behavioral analysis engine. However, a conscious decision was made to architect the system so that the preliminary image processing would be conducted at FPGA boards attached to each camera rather than designing entirely new network cameras to perform this function. This prevents customers from needing to replace their entire IP camera installation with new ones when installing the behavioral analysis system which is the focus of this senior design project. In addition to being able to use existing IP cameras in an installation, the project was designed so that the existing recording and archival equipment could be used without modification. This is because the system being developed in the senior design project is designed to augment existing setups with behavioral analysis. The behavioral analysis system inspects video in real time and, because of this, does not need to manage or have access to video recordings. The system does store logs of what events happen and at what time, but these logs consist primarily of metadata and do not require the large storage capacity of video archive systems. The event logs can be combined with recorded video at a later time by matching the timestamps in the metadata logs to the timestamps in the recorded video.

It is also expected that, as is the case with many new systems, the cost of the system (if productized) would decrease over time. After the initial proof of concept, additional research could take place in how to improve the efficiency of the design. It is possible, for instance, that multiple cameras could be processed by a single FPGA Back-end board. If this is the case, fewer FPGA Back-end would be required in an installation, reducing the overall cost of the system. Further improvements in the software running on the server could also possible increase the server capacity or lower the system requirements. These cost-lowering measures would make the technology more accessible to facilities which do not have the funds to install the initial

model of the behavioral analysis system.

12.3 Frugal Engineering

As with many new technologies, the initial cost of the system is expected to be high. As such, the first revision of the system was not designed for developing markets. The intention was that the early adopters of the system would be security conscious facilities with large budgets (such as secure government / military facilities or corporate R&D labs). However, as the technology matures, it is expected that the cost of the system will decrease. Even though the initial prototype is primarily using high cost, delicate development kits in place of custom hardware, several frugal engineering principles did inform the design of the system architecture. Because the system is meant to be used in safety and security applications, ruggedization was an important feature to consider. Since the Back-end FPGA boards need to be attached to each camera and, because cameras may be positioned outside, it is possible that the FPGA would need to reside outside. Since the functionality of the system depends on the FPGA board functioning correctly, it is important that the FPGA board enclosure be as weather proof as possible. The design of the enclosure would likely consist of high strength plastic coupled with rubber gaskets to protect the circuit board. All of the connectors would be rated for use outdoors. While the ruggedization has implications for the product application, it also makes it automatically suitable for deployment in developing countries where it would potentially not be housed in a heavily controlled indoor environment.

The project was designed to adapt to current IP surveillance installations. While potential customers in the developing world may not have existing IP camera installations, the system does provide the flexibility to use whatever IP cameras are available and within the price range of the customer. This could potentially improve the affordability of the system.

The frugal innovation concepts used in this project including ruggedization, adaptability, and affordability do provide some support for the professional issues and constraints detailed in the senior design requirements. The adaptability and affordability addresses the economics of the system. While it is likely that the system will initially be costly to create, the flexibility given to the consumer to use either existing IP cameras or to buy ones that are within their budget will likely help early adoption of the system. As the technology is refined and efficiency is improved, it is likely that the cost of installations will decrease, bringing the system within the budget of more organizations. The rugiddization and adaptability of the system also addresses the potential environmental impact of the system. Since the system works with existing IP cameras, facilities do not have to discard existing IP cameras after the installation. This reuse lowers the environmental impact of the project. The ruggedization of the boards, coupled with the fact that the FPGA designs can be reconfigured

at a later date with new designs means that the system should remain useful for a long period of time. By not forcing customers to upgrade all parts of the system constantly, the environmental impact of the system is reduced.

Chapter 13

Aesthetics

13.1 Aesthetic Considerations for Productization of Project

First impressions are some of the most influential factors when someone is making a purchasing decision. The aesthetics of a product are often the first impression a perspective customer has. If the aesthetics do not leave the customer with a good impression, the customer may take their business elsewhere even if the product is functionally superior to the competition. Since the project has commercial potential, it is especially important to take visuals into consideration.

While all users within a facility are being monitored by the system, they are not considered the primary end users. The end users of the system are the monitoring staff who are required to watch and analyze the CCTV video feeds. The bulk of user interaction with the system occurs through the Graphical User Interface or GUI. The GUI is the component of the software that shows information as graphical elements and supports interacting with the system via a mouse, touch screen, or other pointing device. GUIs are vastly superior to text-based user interfaces for conveying multiple pieces of information to the user at once. In this case, it also simplifies the user's experience by hiding the implementation details of the system from the end user.

In addition to providing a good first impression of the system, it is also important to provide an experience that abstracts away the complexity of the underlying implementation. Most people are not interested in the inner workings of their products and are only concerned that the product performs as advertised. The idea of this project is to provide a seamless experience to the client. This includes making our hardware as unobtrusive as possible, ideally keeping it out of sight. The goal of our project is to blend it in with standard CCTV equipment. To accomplish this, the hardware boxes connected to each camera need to be small, discrete, and capable of being placed where existing wiring is routed.

13.1.1 Graphical User Interface (GUI)

The user interface is designed to be used by a small, trained, monitoring team. As a result, the software design is not as aesthetically pleasing as some other more widely used consumer products. Due to the overall scope, the project is emphasizing functionality and features over aesthetics. Despite the circumstances of the project, a few software oriented aesthetic ideas were implemented including: graphical user interface layout and key aesthetically oriented functional features.

The user interface is oriented similarly to most modern user interfaces (see Figure 13.1). The main section of the interface is the surveillance video feed. This section is designed to occupy two thirds of the screen for visual appeal. The remaining part of the screen is divided up into two other sections. While complementing the modern look of the system, it grants additional functionality for information selection and control. This design feeds off a modern photography concept called the Rule of Thirds, which suggests that an image is more appealing if the main focus is off center [38]. Adding functionally to the sides of the window, not only makes the interface more usable, but also more aesthetically pleasing. This concept is commonly used in software engineering today, particularly in graphical user interfaces.

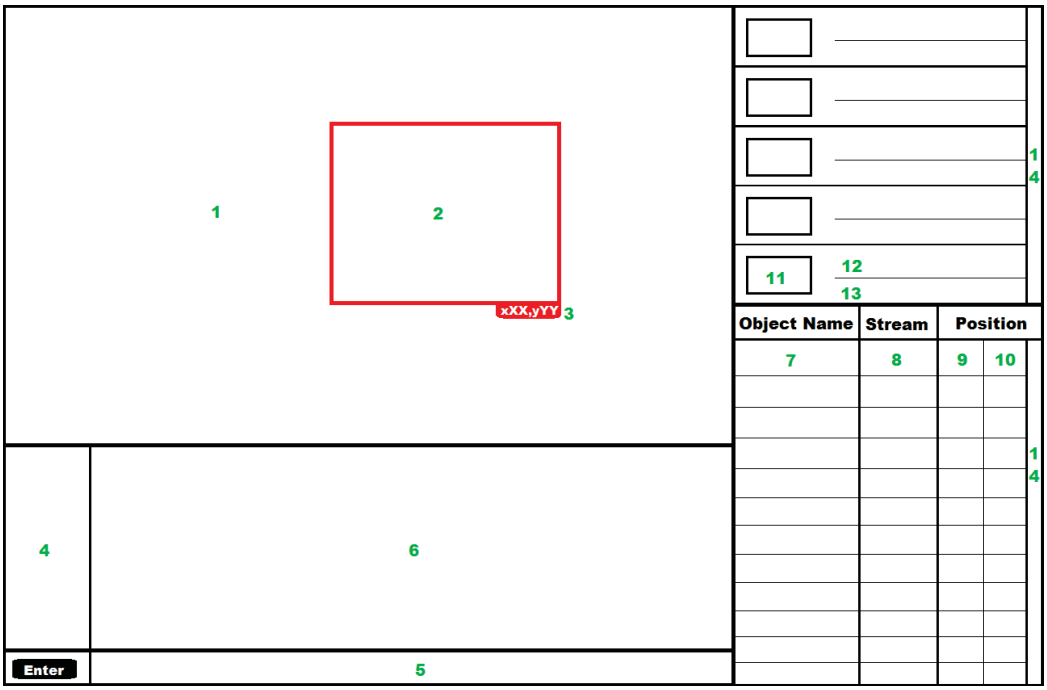


Figure 13.1: User Interface Concept

Some of the main features of the project were not only included for ease of use, but also for aesthetic appeal. The mouse can be used to imbue a more dynamic and cleaner feel to the user interface. Revealing dropdown menus after clicking on an object or video stream is one example of these features. The more time

available to develop the project, the more refined the interface can be made.

13.1.2 Hardware

There are three major hardware components that make up the system: the CCTV cameras, the servers, and the FPGA back-end boards. The aesthetics of each component is described in depth below.

CCTV Cameras

The CCTV cameras are purchased off-the-shelf for the system by the customer. The only requirement is that the cameras are ONVIF compliant, which many commercial cameras are. As such, the aesthetics of the cameras are outside of the scope of our project and are determined by the customers purchasing the system. Ideally, the customer purchases cameras which fit well with the environment in which they are being placed. This is the only part of the system that is visible to the general public with the possible exception of the FPGA back-end boards depending on the installation.

Servers

Like the CCTV cameras, the servers are provided by the customer. The system uses the server to run the Identity Engine, the Database, and the Behavioral Analysis Engine. While these components are crucial to the functionality of the system, the servers can be located anywhere in the facility. It is likely that the servers will be located in the organizations server room or datacenter where the general public does not see them. Since large organizations usually have preferred server providers and often have an existing server installation, the aesthetics of the server are outside of the scope of the project and rest with the customer.

FPGA Back-end Board

The FPGA Back-end board is the only the component of the system, besides the cameras, which may be visible to the general public. It is also the only hardware component in the project with which the team has direct control over aesthetics. Each FPGA Back-end board is linked to a camera via an Ethernet cable and switch. Because the board does not need to be physically attached to the camera body, it may be hidden discreetly. For example, the board may be hidden above ceiling tiles where the general public cannot see it. One of the guiding principles behind the project was that the behavioral analysis system would be able to integrate seamlessly into existing CCTV systems. As such, the FPGA Back-end boards should not draw attention to themselves. In order to maintain their low profile, commercial implementations of the boards would be encased in opaque plastic shells with slots for mounting brackets. The shape of the enclosures would feature rounded corners with tapered edges and would be thin so that it is less noticeable if a wall mount is required. The shells would have several status LEDs to indicate the operational state of the boards.

LCD line displays would be visible through a window in each enclosure and would display detailed diagnostic information such as the boards IP address and error codes. These features would allow a technician to diagnose problems with the board efficiently while still not drawing attention to the board from the general public.

It is worth noting that the scope of this senior design project is a proof of concept system. As such, a development kit has be used in place of a custom FPGA Back-end board. Since the boards being used are not the ones that would be used in a final product, the enclosure described above will not be produced for this project. The description above represents the aesthetics of the system after the development of custom PCBs (Printed Circuit Boards) for the FPGA Back-end.

Chapter 14

Conclusion

CCTV is one of the most commonly employed safety and security technologies in use today. However, the use of CCTV to identify and react to incidents in real time has been hampered by its need for human monitors to make sense of the video feeds. In most cases, the relatively small number of human monitors are unable to observe the potentially exorbitant number of cameras feeds simultaneously. This has led to CCTV being a primarily retrospective tool. Due to new advances in technology, it is now possible for machines to perform some tasks which previously required a human monitor. The project being proposed seeks to use this new technology to augment existing CCTV systems with behavioral analytics. The system uses a collection of FPGAs and computers to process video from the cameras and track object movement in a facility. It uses this information to build a model of normal behavior and flags abnormal movement for review by monitoring staff. This semi-automated system is not meant to replace human monitors but is meant to draw their attention to relevant camera feeds.

Since this is a Senior Design project, there was a very limited time span to develop the system. In light of this, the scope of the project was scaled back to be a proof of concept system. Because several components of the projects could be Senior Design projects in their own right (such as facial recognition), these components were replaced with simpler ones to demonstrate basic functionality. An assumption was made that these components would continue to undergo independent development and could replace the simpler components used in this project at a later time.

While a substantial amount of progress was made in implementing the system, the final implementations of the FPGA back-end and Identity Engine were not completed. However, MATLAB simulations were written for each components and were used in the first end-to-end demonstration of the system. In addition, an IP core was created to demonstrate the dataflow proposed for the FPGA based accelerator. This demo is currently non-operational but is actively being debugged and is expected to be operational shortly.

The $47,925.53 theoretical cost of the project may appear exorbitant a first glance, however, it is actually

on par with existing safety and security technologies. Many CCTV systems utilize cameras that cost over $1,000 each. The proposed budget accounts for both the materials required in an end installation as well as the development tools which are not part of the end system. Additionally, since this is a proof of concept design, pre-made development kits are being used instead of custom designed boards. These development kits contain more components than are necessary for the project. If productions boards were manufactured, they would likely cost much less than the development kits. A preliminary cost estimate suggests that each FPGA back-end would cost approximately $566.57 to produce. This is competitive to the pricing of existing "smart cameras" on the market.

Overall, this was a very ambitious project with profound real world applications. A large amount of the work set out to be accomplished was completed but there is still room for additional development and refinement. With additional development time and resources, this system could potentially be productized and brought to market.

Chapter 15

Bibliography

[1] Altera Corporation, "Cyclone v soc development kit and soc embedded design suite," Nov. 2013. [Online]. Available: http://www.altera.com/products/devkits/altera/kit-cyclone-v-soc.html

[2] *Altera SoC Golden System Reference Design User Guide*, Altera Corporation, 101 Innovation Drive, San Jose, CA 95134, May 2013, docID: UG-01138.

[3] R. Pierce, "Normal distribution," Feb. 2014. [Online]. Available: http://www.mathsisfun.com/data/standard-normal-distribution.html

[4] *Cyclone V SoC Development Board - Reference Manual*, Altera Corporation, 101 Innovation Drive, San Jose, CA 95134, May 2013, docID: MNL-01077-1.0.

[5] Colfax International, "Colfax proedge sxt8600 workstation." [Online]. Available: http://www.colfax-intl.com/nd/workstations/sxt8600.aspx

[6] *Cyclone V Device Overview*, Altera Corporation, 101 Innovation Drive, San Jose, CA 95134, Dec 2013, docID: CV-51001.

[7] H. Yan, "Body found in hospital stairwell: San francisco sheriff details what went wrong," 2013. [Online]. Available: http://www.cnn.com/2013/11/07/us/california-body-in-stairwell/

[8] R. Salonga, "Santa clara teen stowaway's survival in jet's wheelwell was literally death-defying," 2014. [Online]. Available: http://www.mercurynews.com/bay-area-news/ci_25610408/san-jose-santa-clara-teens-survival-from-flight

[9] *ONVIF Profile S Specification*, Open Network Video Interface Forum, December 2014. [Online]. Available: http://www.onvif.org/Portals/0/documents/op/20984_11763r01ONVIF_Members-Profile_S_Specification.pdf

[10] *ONVIF Core Specification*, Open Network Video Interface Forum, December 2012. [Online]. Available: http://www.onvif.org/specs/core/ONVIF-Core-Specification-v221.pdff

[11] *Cyclone V SoC Development Board - User Guide*, Altera Corporation, 101 Innovation Drive, San Jose, CA 95134, May 2013, docID: UG-01135-1.0.

[12] *Cyclone V Device Handbook*, Altera Corporation, 101 Innovation Drive, San Jose, CA 95134, Jan 2014, docID: CV-5V2.

[13] Serge P. Hoogendoorn, PhD and Winnie Daamen, PhD, "Free speed distributions for pedestrian traffic," 2006.

[14] D-Link, "Cloud camera 2300, outdoor hd day/night network cloud camera - dcs-2310l," May 2014. [Online]. Available: http://www.dlink.com/us/en/home-solutions/view/network-cameras/dcs-2310l-outdoor-hd-poe-day-night-cloud-camera

[15] J. McCool, "100base-t: an overview," in *WESCON/'95. Conference record. 'Microelectronics Communications Technology Producing Quality Products Mobile and Portable Power Emerging Technologies'*, Nov 1995, pp. 342–. [Online]. Available: http://ieeexplore.ieee.org/xpl/articleDetails.jsp?tp=&arnumber=485300

[16] F. Edwards and M. Schulz, "Performance of vbr packet video communications on an ethernet lan: A trace-driven simulation study," in *Computers and Communications, 1994., IEEE 13th Annual International Phoenix Conference on*, Apr 1994, pp. 427–. [Online]. Available: http://ieeexplore.ieee.org/xpl/articleDetails.jsp?tp=&arnumber=504149

[17] Altera Corporation, "Altera product catalog," 2013, v. 13.1. [Online]. Available: http://www.altera.com/literature/sg/product-catalog.pdf

[18] Microsoft Corporation, "Maximum capacity specifications for sql server," 2014. [Online]. Available: http://msdn.microsoft.com/en-us/library/ms143432.aspx

[19] ——, "Binary large object (blob) data (sql server)," 2014. [Online]. Available: http://msdn.microsoft.com/en-us/library/bb895234.aspx

[20] R. Mistry and S. Misner, *Introducing Microsoft SQL Server 2014: Technical Overview*. Microsoft Press, 2014. [Online]. Available: http://aka.ms/684751pdf

[21] Microsoft Corporation, "Microsoft sql server 2014: Record-setting performance," 2014. [Online]. Available: http://download.microsoft.com/download/4/F/C/4FCAE4E4-4410-4586-8F6D-E0196F96ACFE/SQL_Server_2014_Performance_Benchmarks.pdf

[22] Altera Corporation, "Golden system reference design (gsrd)," Nov. 2013. [Online]. Available: ftp://ftp.altera.com/outgoing/devkit/13.0/cycloneVSX_5csxfc6df31es_soc_v13.0.0.1.exe

[23] Live Networks, Inc., "Live555 streaming media." [Online]. Available: http://www.live555.com/liveMedia/

[24] IJG, "Independent jpeg group." [Online]. Available: http://www.ijg.org/

[25] *Avalon Interface - Specifications*, Altera Corporation, 101 Innovation Drive, San Jose, CA 95134, April 2014, docID: MNL-AVABUSREF-2.1.1.

[26] Altera Corporation, "Golden hardware reference design (ghrd)," Nov. 2013. [Online]. Available: ftp://ftp.altera.com/outgoing/devkit/13.0/cycloneVSX_5csxfc6df31es_soc_v13.0.0.1.exe

[27] J. Canny, "A computational approach to edge detection," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. PAMI-8, no. 6, pp. 679–698, Nov 1986. [Online]. Available: http://ieeexplore.ieee.org/xpl/articleDetails.jsp?tp=&arnumber=4767851

[28] D. Banks, S. Linton, and P. Stockmeyer, "Counting cases in substitope algorithms," *Visualization and Computer Graphics, IEEE Transactions on*, vol. 10, no. 4, pp. 371–384, July 2004. [Online]. Available: http://ieeexplore.ieee.org/xpl/articleDetails.jsp?tp=&arnumber=1298795

[29] W. E. Lorensen and H. E. Cline, "Marching cubes: A high resolution 3d surface construction algorithm," in *Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques*, ser. SIGGRAPH '87. New York, NY, USA: ACM, 1987, pp. 163–169. [Online]. Available: http://doi.acm.org/10.1145/37401.37422

[30] D. Rajon and W. Bolch, "Marching cube algorithm: review and trilinear interpolation adaptation for image-based dosimetric models," *Computerized Medical Imaging and Graphics*, vol. 27, no. 5, pp. 411 – 435, 2003. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0895611103000326

[31] Altera Corporation, "Golden top," Nov. 2013. [Online]. Available: ftp://ftp.altera.com/outgoing/devkit/13.0/cycloneVSX_5csxfc6df31es_soc_v13.0.0.1.exe

[32] ——, “Board test system (bts),” Nov. 2013. [Online]. Available: ftp://ftp.altera.com/outgoing/devkit/13.0/cycloneVSX_5csxfc6df31es_soc_v13.0.0.1.exe

[33] ——, “Avalon memory-mapped master templates,” Sept. 2008. [Online]. Available: http://www.altera.com/support/examples/nios2/exm-avalon-mm.html

[34] RocketBoards.org, “Documentation,” Dec. 2013. [Online]. Available: http://www.rocketboards.org/foswiki/Documentation/WebHome

[35] *Altera SoC Embedded Design Suite - User Guide*, Altera Corporation, 101 Innovation Drive, San Jose, CA 95134, 2013.

[36] X. Zhu, R. M. Rangayyan, and A. L. Ells, “Digital image processing for ophthalmology: Detection of the optic nerve head,” *Synthesis Lectures on Biomedical Engineering*, vol. 6, no. 1, pp. 1–106, 2011. [Online]. Available: http://dx.doi.org/10.2200/S00335ED1V01Y201102BME040

[37] “DeepFace: Closing the Gap to Human-Level Performance in Face Verification,” *Conference on Computer Vision and Pattern Recognition (CVPR)*.

[38] J. Smith, *Remarks on Rural Scenery; with Twenty Etchings of Cottages, from Nature; and Some Observations and Precepts Relative to the Pictoresque. By John Thomas Smith, Engraver of the Antiquities of London*. June MDCCXCVII. printed [by Joseph Downes] for, and sold by Nathaniel Smith ancient Print seller at Rembrandts-Head May’s Buildings, St. Martin’s Lane, and I.T. Smith, at No 40 Trith Street Soho, 1797. [Online]. Available: http://books.google.com/books?id=YWZHtwAACAAJ

[39] VideoLAN, “VLC ActiveX plugin for .Net libraries.” Feb. 2014. [Online]. Available: www.videolan.org

[40] S. B. Boyd, “Life-cycle assessment of semiconductors,” Ph.D. dissertation, 2009. [Online]. Available: http://escholarship.org/uc/item/8bv2s63d

[41] C. Jones, “Inventory of carbon and energy (ice),” University of Bath, Dataset, 2011. [Online]. Available: http://www.circularecology.com/ice-database.html

Chapter 16

Appendix

16.1 FPGA Specifications

FPGA specifications were duplicated in Table 16.1 and Table 16.2 from *Cyclone V Device Overview* for the reader's convenience [6].

Attribute	Value
Family	Cyclone V
Variant	SoC with 3-Gbps Transceivers
Embedded Hard IP	Maximum 2 hard PCIe controllers and 1 hard memory controller
Number Logic Elements	110K
Number Transceiver	9
Transceiver Speed Grade	3.125 Gbps
Package Type	FBGA
Package Pins	896
Operating Temperature	Commercial (0 - 85 deg C)
Additional Notes	Lead Free, Engineering Sample

Table 16.1: FPGA (5CSXFC6D6F31C8NES) Overview [6]

Resource	Quantity
Logic Element (LE)	110K
ALM	41,509
Register	166,036
Memory (M10K)	5,570 Kb
Memory (MLAB)	621 Kb
Variable-precision DSP Block	112
18 x 18 Multiplier	224
FPGA PLL	6
HPS PLL	3
3 Gbps Transceiver	9
FPGA GPIO	228
HPS I/O	181
LVDS (Transmitter)	72
LVDS (Receiver)	72
PCIe Hard IP Block	2
FPGA Hard Memory Controller	1
HPS Hard Memory Controller	1
ARM Cortex-A9 MPCore Processor	Dual-core

Table 16.2: FPGA (5CSXFC6D6F31C8NES) Resource Table [6]

16.2 Object Detection - Matlab Simulation Code

```
% Christopher Yarp
% Contour Test
% Senior Design

close all;
clear; clc;

%x = imread('eight.tif');
thresh = 80;
minPts = 10;
minLength = 2;

slackSquareBounding = 0.70;
%slackCircle = ;
slackCenter = 6;

slackRatio = 1;

ratioCenterInner = 3;
ratioCenterOuter = 4.25;

boundingBoxMargin = 5;

imgNameFormat = './bmp/stillSampleMultiple_%010d.bmp';
startImgName = 2;
endImgName = 850;
%startImgName = 319;
%endImgName = 320;
imgNameStep = 1;
fps = 30/imgNameStep;

videoFileName = 'tracking.avi';

M(floor((endImgName-startImgName)/imgNameStep)) = struct('cdata',[],'colormap',[]);

vidFrameNum = 1;
```

```
37
38 videoWriter = VideoWriter(videoFileName);
39 open(videoWriter);
40
41 for frameNum = startImgName:imgNameStep:endImgName
42     clear_all_but('thresh', 'minPts', 'minLength', 'slackSquareBounding', ...
43         'slackCenter', 'slackRatio', 'ratioCenterInner', 'ratioCenterOuter', ...
44         'boundingBoxMargin', 'imgNameFormat', 'startImgName', 'endImgName', ...
45         'imgNameStep', 'fps', 'M' , 'frameNum', 'vidFrameNum', 'videoFileName', ...
46         'videoWriter');
47
48     fileName = sprintf(imgNameFormat, frameNum);
49     %img = importdata('stillSample_0000000100.bmp');
50     %img = importdata('stillSample_0000000300.bmp');
51     %img = importdata('stillSample_0000000600.bmp');
52     img = importdata(fileName);
53
54     display(['frame: ' fileName]);
55
56     x = rgb2gray(img);
57     fig = figure(1);
58     %figure size from http://www.mathworks.com/matlabcentral/newsreader/view_thread/262957
59     set(fig, 'Position', [0, 100, 1280, 800]);
60
61     subplot(2,2,1);
62     imshow(x);
63     title('Greyscale Image');
64     subplot(2,2,2);
65     imshow(img);
66     title(['Origional Img: ' fileName ],'Interpreter','none');
67     %subplot(2,2,4)
68     %h1 = hist(double(x),256);
69     %plot(h1)
70     %h1 = imhist(x,256);
71     %plot(h1)
72
73     subplot(2,2,3);
74     %imcontour(x,[225, 225])
75     c1 = imcontour(x,[thresh, thresh]);
76     title('Contours');
77
78
79     ix = 1;
80     ic = 1;
81     npts = 0;
82     maxX = 0;
83     minX = 0;
84     maxY = 0;
85     minY = 0;
86     numCurvesRec = 1;
87
88     startInd = 1;
89     endInd = 1;
90
91     [c1Rows, c1Cols] = size(c1);
92
93     for i = 1:c1Cols
94         if(i == ix)
95             sizeX = maxX-minX;
96             sizeY = maxY-minY;
97
98             if(sizeX > sizeY)
99                 maxSide = sizeX;
100                minSide = sizeY;
101            else
102                maxSide = sizeY;
103                minSide = sizeX;
104            end
```

```
105
106             if(maxSide>0 && minSide >0)
107                 boundingBoxRatio = minSide/maxSide;
108             else
109                 boundingBoxRatio = 0;
110             end
111             %display(['minX ' num2str(minX) ', maxX ' num2str(maxX) ', minY ' num2str(
                    minY) 'maxY ' num2str(maxY)]);
112             %display(['sizeX ' num2str(sizeX) ', sizeY ' num2str(sizeY)]);
113             %display(['boundingBoxRatio ' num2str(boundingBoxRatio)]);
114
115             %filter out curves that have two few points or have nonsquare
116             %bounding boxes
117             if( i ~= 1 && npts>=minPts && sizeX > minLength && sizeY > minLength &&
                    boundingBoxRatio >= slackSquareBounding)
118                 %record data of prev curve (including maxx, maxy)
119                 s = struct('npts', npts, 'minX', minX, 'maxX', maxX, 'minY', minY, 'maxY',
                        maxY, 'startInd', startInd, 'endInd', endInd);
120                 curves(numCurvesRec) = s;
121                 %display(['contour ' num2str(ic) ', points = ' num2str(npts) ', (['
                        num2str(minX) ', ' num2str(maxX) '], [' num2str(minY) ', ' num2str(
                        maxY), '])'])
122                 numCurvesRec = numCurvesRec+1;
123             end
124             minX = c1(1, i+1);
125             maxX = c1(1, i+1);
126
127             minY = c1(2, i+1);
128             maxY = c1(2, i+1);
129
130             npts = c1(2, i);
131             startInd = ix+1;
132             ix = ix + npts+1;
133             endInd = ix-1;
134             ic = ic+1;
135         else
136             if(c1(1, i) < minX)
137                 minX = c1(1, i);
138             end
139             if(c1(1, i) > maxX)
140                 maxX = c1(1, i);
141             end
142             if(c1(2, i) < minY)
143                 minY = c1(2, i);
144             end
145             if(c1(2, i) > maxY)
146                 maxY = c1(2, i);
147             end
148         end
149     end
150
151     %record last curve
152     sizeX = maxX-minX;
153     sizeY = maxY-minY;
154
155     if(sizeX > sizeY)
156         maxSide = sizeX;
157         minSide = sizeY;
158     else
159         maxSide = sizeY;
160         minSide = sizeX;
161     end
162
163     if(maxSide>0 && minSide >0)
164         boundingBoxRatio = minSide/maxSide;
165     else
166         boundingBoxRatio = 0;
167     end
```

```
168
169     if(~isempty(c1) && npts>=minPts && sizeX > minLength && sizeY > minLength &&
            boundingBoxRatio >= slackSquareBounding)
170         %record data of prev curve (including maxx, maxy)
171         s = struct('npts', npts, 'minX', minX, 'maxX', maxX, 'minY', minY, 'maxY', maxY, '
                startInd', startInd, 'endInd', endInd);
172         curves(numCurvesRec) = s;
173         display(['contour ' num2str(ic) ', points = ' num2str(npts) ',  ([' num2str(minX)
                ', ' num2str(maxX) '], [' num2str(minY) ', ' num2str(maxY), '])'])
174     end
175
176     display(['num contours: ' num2str(ic)])
177     display(['num potential contours: ' num2str(numCurvesRec)])
178
179     %check center points (this is n^2 -> not good!!!!)
180     used = zeros(length(curves));
181
182     foundCount = 0;
183
184     for i = 1:length(curves)
185         if(used(i)==0)
186             centerX = (curves(i).maxX + curves(i).minX)/2;
187             centerY = (curves(i).maxY + curves(i).minY)/2;
188             %display(['(' num2str(centerX) ', ' num2str(centerY) ')'])
189
190             numConcentric = 1;
191             concentric(1) = curves(i);
192
193             centerCircleDim = ((curves(i).maxX - curves(i).minX)+(curves(i).maxY + curves(
                    i).minY))/2;
194
195             for j = i+1:length(curves)
196                 centerXnew = (curves(j).maxX + curves(j).minX)/2;
197                 centerYnew = (curves(j).maxY + curves(j).minY)/2;
198
199                 %check if curve already put in bin and if centers are same
200                 if(used(j) == 0 && abs(centerX - centerXnew) < slackCenter && abs(centerY
                        - centerYnew) < slackCenter)
201                     numConcentric = numConcentric + 1;
202                     concentric(numConcentric) = curves(j);
203
204                     if centerCircleDim > ((curves(j).maxX - curves(j).minX)+(curves(j).
                            maxY - curves(j).minY))/2;
205                         centerCircleDim = ((curves(j).maxX - curves(j).minX)+(curves(j).
                                maxY - curves(j).minY))/2;
206                         center = curves(j);
207                     end
208                     used(j) = 1;
209                 end
210             end
211
212             % if more than 3 possibilities, we may have the target, we now need to
213             % check ratios of bounding boxes
214             if(numConcentric >= 3)
215                 display(['possibly found one at (' num2str(centerX) ', ' num2str(centerY)
                        ')']);
216                 %candidateGroups = candidateGroups +1;
217                 %candidateGroup(candidateGroups) = concentric;
218
219                 %now look at ratios -> may be overkill for our application
220                 foundInner = 0;
221                 foundOuter = 0;
222                 for k = 1:length(concentric)
223                     newCircleDim = ((concentric(k).maxX - concentric(k).minX)+(concentric(
                            k).maxY - concentric(k).minY))/2;
224                     if abs(newCircleDim/centerCircleDim - ratioCenterInner) <= slackRatio
225                        foundInner = 1;
226                        inner = concentric(k);
```

```
                    elseif abs(newCircleDim/centerCircleDim - ratioCenterOuter) <=
                        slackRatio
                        foundOuter = 1;
                        outer = concentric(k);
                    end
                end

                if(foundInner ~=0 && foundOuter ~=0)
                    %found it
                    foundCount = foundCount+1;
                    foundStr = struct('center', center, 'inner', inner, 'outer', outer);
                    found(foundCount) = foundStr;
                    display(['found one at (' num2str(centerX) ', ' num2str(centerY) ')'])
                        ;
                end
            end
        end
    end

    %now plot rings and where we found them
    subplot(2,2,4)
    imshow(img)
    title('Detected Objects');
    hold on;
    for l = 1:length(found)
        centerBegin = found(l).center.startInd;
        centerEnd = found(l).center.endInd;
        plot(c1(1, centerBegin:centerEnd), c1(2, centerBegin:centerEnd), 'y', 'LineWidth',
            2);

        innerBegin = found(l).inner.startInd;
        innerEnd = found(l).inner.endInd;
        plot(c1(1, innerBegin:innerEnd), c1(2, innerBegin:innerEnd), 'g', 'LineWidth', 2);

        outerBegin = found(l).outer.startInd;
        outerEnd = found(l).outer.endInd;
        plot(c1(1, outerBegin:outerEnd), c1(2, outerBegin:outerEnd), 'b', 'LineWidth', 2);

        %plot bounding box
        plot([(found(l).outer.minX - boundingBoxMargin), (found(l).outer.minX -
            boundingBoxMargin), (found(l).outer.maxX + boundingBoxMargin), (found(l).outer
            .maxX + boundingBoxMargin), (found(l).outer.minX - boundingBoxMargin)], ...
            [(found(l).outer.minY - boundingBoxMargin), (found(l).outer.maxY +
                boundingBoxMargin), (found(l).outer.maxY + boundingBoxMargin), (found(l).
                outer.minY - boundingBoxMargin), (found(l).outer.minY - boundingBoxMargin
                )], 'r', 'LineWidth', 4);
    end
    hold off;

    M(vidFrameNum) = getframe(fig);
    writeVideo(videoWriter,M(vidFrameNum));

    vidFrameNum = vidFrameNum+1;
end

% Video writer: http://www.mathworks.com/help/matlab/ref/videowriter-class.html
close(videoWriter);

implay(M, fps);
```

Listing 16.1: Contour Matlab Simulation Code

16.3 FPGA Demo - Hardware Description

16.3.1 Directory Structure

```
orion/
├── orion_system.qpf
├── orion_system.qsf
├── orion_system.qsys
├── orion_system.sdc
└── ip
    └── ipTest
        ├── burst_read_master.v
        ├── burst_write_master.v
        ├── custom_master.v
        ├── IPTest_hw.tcl
        ├── latency_aware_read_master.v
        ├── SCU_Logo_Seperate_Small.gif
        ├── slave_template.v
        ├── slave_template_macros.h
        ├── test_ip.v
        ├── test_ip_top.v
        └── write_master.v
```

16.3.2 Qsys System - GUI View (orion_system.qsys)

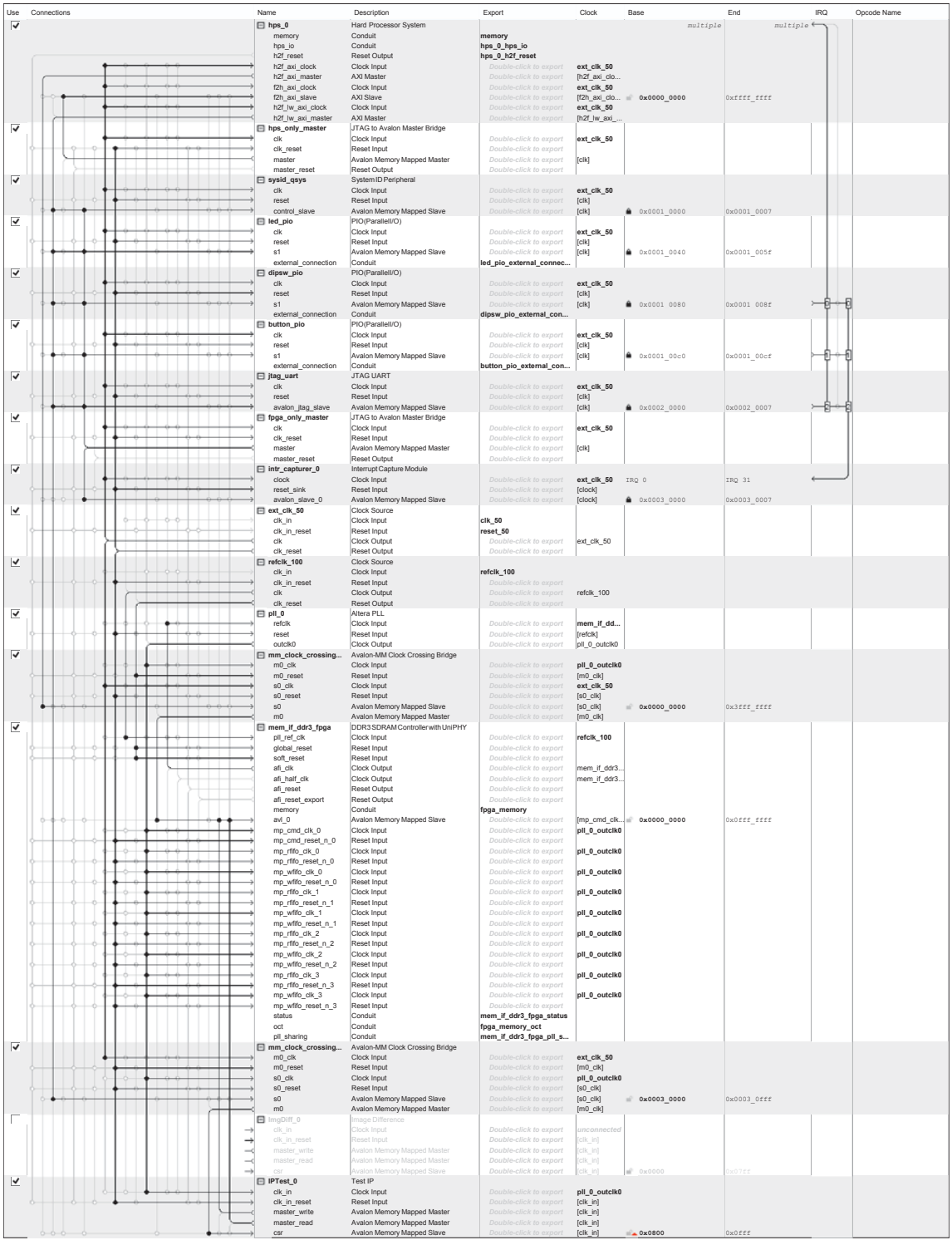


Figure 16.1: Qsys Design - GUI View (orion_system.qsys)

16.3.3 Qsys System - Code View (orion_system.qsys)

```
<?xml version="1.0" encoding="UTF-8"?>
<system name="$${FILENAME}">
 <component
   name="$${FILENAME}"
   displayName="$${FILENAME}"
   version="1.0"
   description=""
   tags=""
   categories="System" />
 <parameter name="bonusData"><![CDATA[bonusData
{
   element $${FILENAME}
   {
   }
   element IPTest_0
   {
      datum _sortIndex
      {
         value = "16";
         type = "int";
      }
   }
   element ImgDiff_0
   {
      datum _sortIndex
      {
         value = "15";
         type = "int";
      }
   }
   element jtag_uart.avalon_jtag_slave
   {
      datum _lockedAddress
      {
         value = "1";
         type = "boolean";
      }
      datum baseAddress
      {
         value = "131072";
         type = "String";
      }
   }
   element intr_capturer_0.avalon_slave_0
   {
      datum _lockedAddress
      {
         value = "1";
         type = "boolean";
      }
      datum baseAddress
      {
         value = "196608";
         type = "String";
      }
   }
   element mem_if_ddr3_fpga.avl_0
   {
      datum baseAddress
      {
         value = "0";
         type = "String";
      }
   }
   element button_pio
   {
```

```
67          datum _sortIndex
68          {
69              value = "5";
70              type = "int";
71          }
72          datum sopceditor_expanded
73          {
74              value = "1";
75              type = "boolean";
76          }
77      }
78      element sysid_qsys.control_slave
79      {
80          datum _lockedAddress
81          {
82              value = "1";
83              type = "boolean";
84          }
85          datum baseAddress
86          {
87              value = "65536";
88              type = "String";
89          }
90      }
91      element IPTest_0.csr
92      {
93          datum baseAddress
94          {
95              value = "2048";
96              type = "String";
97          }
98      }
99      element ImgDiff_0.csr
100     {
101         datum baseAddress
102         {
103             value = "0";
104             type = "String";
105         }
106     }
107     element dipsw_pio
108     {
109         datum _sortIndex
110         {
111             value = "4";
112             type = "int";
113         }
114     }
115     element ext_clk_50
116     {
117         datum _sortIndex
118         {
119             value = "9";
120             type = "int";
121         }
122     }
123     element hps_0.f2h_axi_slave
124     {
125         datum _lockedAddress
126         {
127             value = "0";
128             type = "boolean";
129         }
130         datum baseAddress
131         {
132             value = "0";
133             type = "String";
134         }
```

```
135     }
136     element fpga_only_master
137     {
138        datum _sortIndex
139        {
140           value = "7";
141           type = "int";
142        }
143     }
144     element hps_0
145     {
146        datum _sortIndex
147        {
148           value = "0";
149           type = "int";
150        }
151     }
152     element hps_only_master
153     {
154        datum _sortIndex
155        {
156           value = "1";
157           type = "int";
158        }
159     }
160     element intr_capturer_0
161     {
162        datum _sortIndex
163        {
164           value = "8";
165           type = "int";
166        }
167     }
168     element jtag_uart.irq
169     {
170        datum _tags
171        {
172           value = "";
173           type = "String";
174        }
175     }
176     element jtag_uart
177     {
178        datum _sortIndex
179        {
180           value = "6";
181           type = "int";
182        }
183     }
184     element led_pio
185     {
186        datum _sortIndex
187        {
188           value = "3";
189           type = "int";
190        }
191     }
192     element mem_if_ddr3_fpga
193     {
194        datum _sortIndex
195        {
196           value = "13";
197           type = "int";
198        }
199     }
200     element mm_clock_crossing_bridge_0
201     {
202        datum _sortIndex
```

```
203         {
204             value = "12";
205             type = "int";
206         }
207     }
208     element mm_clock_crossing_bridge_1
209     {
210         datum _sortIndex
211         {
212             value = "14";
213             type = "int";
214         }
215     }
216     element pll_0
217     {
218         datum _sortIndex
219         {
220             value = "11";
221             type = "int";
222         }
223     }
224     element refclk_100
225     {
226         datum _sortIndex
227         {
228             value = "10";
229             type = "int";
230         }
231     }
232     element jtag_uart.reset
233     {
234         datum _tags
235         {
236             value = "";
237             type = "String";
238         }
239     }
240     element mm_clock_crossing_bridge_1.s0
241     {
242         datum baseAddress
243         {
244             value = "196608";
245             type = "String";
246         }
247     }
248     element mm_clock_crossing_bridge_0.s0
249     {
250         datum baseAddress
251         {
252             value = "0";
253             type = "String";
254         }
255     }
256     element dipsw_pio.s1
257     {
258         datum _lockedAddress
259         {
260             value = "1";
261             type = "boolean";
262         }
263         datum baseAddress
264         {
265             value = "65664";
266             type = "String";
267         }
268     }
269     element led_pio.s1
270     {
```

```
271         datum _lockedAddress
272         {
273             value = "1";
274             type = "boolean";
275         }
276         datum baseAddress
277         {
278             value = "65600";
279             type = "String";
280         }
281     }
282     element button_pio.s1
283     {
284         datum _lockedAddress
285         {
286             value = "1";
287             type = "boolean";
288         }
289         datum baseAddress
290         {
291             value = "65728";
292             type = "String";
293         }
294     }
295     element sysid_qsys
296     {
297         datum _sortIndex
298         {
299             value = "2";
300             type = "int";
301         }
302     }
303 }
304 ]]></parameter>
305  <parameter name="clockCrossingAdapter" value="FIFO" />
306  <parameter name="device" value="5CSXFC6D6F31C8ES" />
307  <parameter name="deviceFamily" value="Cyclone V" />
308  <parameter name="deviceSpeedGrade" value="8_H6" />
309  <parameter name="fabricMode" value="QSYS" />
310  <parameter name="generateLegacySim" value="false" />
311  <parameter name="generationId" value="0" />
312  <parameter name="globalResetBus" value="false" />
313  <parameter name="hdlLanguage" value="VERILOG" />
314  <parameter name="maxAdditionalLatency" value="4" />
315  <parameter name="projectName" value="orion_system.qpf" />
316  <parameter name="sopcBorderPoints" value="false" />
317  <parameter name="systemHash" value="1" />
318  <parameter name="timeStamp" value="1399677421041" />
319  <parameter name="useTestBenchNamingPattern" value="false" />
320  <instanceScript></instanceScript>
321  <interface name="memory" internal="hps_0.memory" type="conduit" dir="end" />
322  <interface name="hps_0_hps_io" internal="hps_0.hps_io" type="conduit" dir="end" />
323  <interface name="clk_50" internal="ext_clk_50.clk_in" type="clock" dir="end" />
324  <interface
325    name="led_pio_external_connection"
326    internal="led_pio.external_connection"
327    type="conduit"
328    dir="end" />
329  <interface
330    name="dipsw_pio_external_connection"
331    internal="dipsw_pio.external_connection"
332    type="conduit"
333    dir="end" />
334  <interface
335    name="button_pio_external_connection"
336    internal="button_pio.external_connection"
337    type="conduit"
338    dir="end" />
```

```
<interface
  name="hps_0_h2f_reset"
  internal="hps_0.h2f_reset"
  type="reset"
  dir="start" />
<interface
  name="fpga_memory"
  internal="mem_if_ddr3_fpga.memory"
  type="conduit"
  dir="end" />
<interface
  name="fpga_memory_oct"
  internal="mem_if_ddr3_fpga.oct"
  type="conduit"
  dir="end" />
<interface name="refclk_100" internal="refclk_100.clk_in" type="clock" dir="end" />
<interface
  name="reset_50"
  internal="ext_clk_50.clk_in_reset"
  type="reset"
  dir="end" />
<interface
  name="mem_if_ddr3_fpga_status"
  internal="mem_if_ddr3_fpga.status"
  type="conduit"
  dir="end" />
<interface
  name="mem_if_ddr3_fpga_pll_sharing"
  internal="mem_if_ddr3_fpga.pll_sharing"
  type="conduit"
  dir="end" />
<module
  kind="altera_avalon_sysid_qsys"
  version="13.0"
  enabled="1"
  name="sysid_qsys">
 <parameter name="id" value="-1395322102" />
 <parameter name="timestamp" value="0" />
 <parameter name="AUTO_CLK_CLOCK_RATE" value="50000000" />
 <parameter name="AUTO_DEVICE_FAMILY" value="Cyclone V" />
</module>
<module kind="altera_hps" version="13.0.1" enabled="1" name="hps_0">
 <parameter name="MEM_VENDOR" value="JEDEC" />
 <parameter name="MEM_FORMAT" value="DISCRETE" />
 <parameter name="RDIMM_CONFIG" value="0000000000000000" />
 <parameter name="LRDIMM_EXTENDED_CONFIG">0x0000000000000000000</parameter>
 <parameter name="DISCRETE_FLY_BY" value="true" />
 <parameter name="DEVICE_DEPTH" value="1" />
 <parameter name="MEM_MIRROR_ADDRESSING" value="0" />
 <parameter name="MEM_CLK_FREQ_MAX" value="800.0" />
 <parameter name="MEM_ROW_ADDR_WIDTH" value="15" />
 <parameter name="MEM_COL_ADDR_WIDTH" value="10" />
 <parameter name="MEM_DQ_WIDTH" value="40" />
 <parameter name="MEM_DQ_PER_DQS" value="8" />
 <parameter name="MEM_BANKADDR_WIDTH" value="3" />
 <parameter name="MEM_IF_DM_PINS_EN" value="true" />
 <parameter name="MEM_IF_DQSN_EN" value="true" />
 <parameter name="MEM_NUMBER_OF_DIMMS" value="1" />
 <parameter name="MEM_NUMBER_OF_RANKS_PER_DIMM" value="1" />
 <parameter name="MEM_NUMBER_OF_RANKS_PER_DEVICE" value="1" />
 <parameter name="MEM_RANK_MULTIPLICATION_FACTOR" value="1" />
 <parameter name="MEM_CK_WIDTH" value="1" />
 <parameter name="MEM_CS_WIDTH" value="1" />
 <parameter name="MEM_CLK_EN_WIDTH" value="1" />
 <parameter name="ALTMEMPHY_COMPATIBLE_MODE" value="false" />
 <parameter name="NEXTGEN" value="true" />
 <parameter name="MEM_IF_BOARD_BASE_DELAY" value="10" />
 <parameter name="MEM_IF_SIM_VALID_WINDOW" value="0" />
```

```
<parameter name="MEM_GUARANTEED_WRITE_INIT" value="false" />
<parameter name="MEM_VERBOSE" value="true" />
<parameter name="PINGPONGPHY_EN" value="false" />
<parameter name="REFRESH_BURST_VALIDATION" value="false" />
<parameter name="MEM_BL" value="OTF" />
<parameter name="MEM_BT" value="Sequential" />
<parameter name="MEM_ASR" value="Manual" />
<parameter name="MEM_SRT" value="Normal" />
<parameter name="MEM_PD" value="DLL off" />
<parameter name="MEM_DRV_STR" value="RZQ/6" />
<parameter name="MEM_DLL_EN" value="true" />
<parameter name="MEM_RTT_NOM" value="RZQ/6" />
<parameter name="MEM_RTT_WR" value="Dynamic ODT off" />
<parameter name="MEM_WTCL" value="6" />
<parameter name="MEM_ATCL" value="Disabled" />
<parameter name="MEM_TCL" value="7" />
<parameter name="MEM_AUTO_LEVELING_MODE" value="true" />
<parameter name="MEM_USER_LEVELING_MODE" value="Leveling" />
<parameter name="MEM_INIT_EN" value="false" />
<parameter name="MEM_INIT_FILE" value="" />
<parameter name="DAT_DATA_WIDTH" value="32" />
<parameter name="TIMING_TIS" value="170" />
<parameter name="TIMING_TIH" value="120" />
<parameter name="TIMING_TDS" value="10" />
<parameter name="TIMING_TDH" value="45" />
<parameter name="TIMING_TDQSQ" value="100" />
<parameter name="TIMING_TQHS" value="300" />
<parameter name="TIMING_TQH" value="0.38" />
<parameter name="TIMING_TDQSCK" value="225" />
<parameter name="TIMING_TDQSCKDS" value="450" />
<parameter name="TIMING_TDQSCKDM" value="900" />
<parameter name="TIMING_TDQSCKDL" value="1200" />
<parameter name="TIMING_TDQSS" value="0.27" />
<parameter name="TIMING_TDQSH" value="0.35" />
<parameter name="TIMING_TQSH" value="0.4" />
<parameter name="TIMING_TDSH" value="0.18" />
<parameter name="TIMING_TDSS" value="0.18" />
<parameter name="MEM_TINIT_US" value="500" />
<parameter name="MEM_TMRD_CK" value="4" />
<parameter name="MEM_TRAS_NS" value="35.0" />
<parameter name="MEM_TRCD_NS" value="13.75" />
<parameter name="MEM_TRP_NS" value="13.75" />
<parameter name="MEM_TREFI_US" value="7.8" />
<parameter name="MEM_TRFC_NS" value="260.0" />
<parameter name="CFG_TCCD_NS" value="2.5" />
<parameter name="MEM_TWR_NS" value="15.0" />
<parameter name="MEM_TWTR" value="4" />
<parameter name="MEM_TFAW_NS" value="30.0" />
<parameter name="MEM_TRRD_NS" value="10.0" />
<parameter name="MEM_TRTP_NS" value="10.0" />
<parameter name="POWER_OF_TWO_BUS" value="false" />
<parameter name="SOPC_COMPAT_RESET" value="false" />
<parameter name="AVL_MAX_SIZE" value="4" />
<parameter name="BYTE_ENABLE" value="true" />
<parameter name="ENABLE_CTRL_AVALON_INTERFACE" value="true" />
<parameter name="CTL_DEEP_POWERDN_EN" value="false" />
<parameter name="CTL_SELF_REFRESH_EN" value="false" />
<parameter name="AUTO_POWERDN_EN" value="false" />
<parameter name="AUTO_PD_CYCLES" value="0" />
<parameter name="CTL_USR_REFRESH_EN" value="false" />
<parameter name="CTL_AUTOPCH_EN" value="false" />
<parameter name="CTL_ZQCAL_EN" value="false" />
<parameter name="ADDR_ORDER" value="0" />
<parameter name="CTL_LOOK_AHEAD_DEPTH" value="4" />
<parameter name="CONTROLLER_LATENCY" value="5" />
<parameter name="CFG_REORDER_DATA" value="true" />
<parameter name="STARVE_LIMIT" value="10" />
<parameter name="CTL_CSR_ENABLED" value="false" />
```

```
<parameter name="CTL_CSR_CONNECTION" value="INTERNAL_JTAG" />
<parameter name="CTL_ECC_ENABLED" value="false" />
<parameter name="CTL_HRB_ENABLED" value="false" />
<parameter name="CTL_ECC_AUTO_CORRECTION_ENABLED" value="false" />
<parameter name="MULTICAST_EN" value="false" />
<parameter name="CTL_DYNAMIC_BANK_ALLOCATION" value="false" />
<parameter name="CTL_DYNAMIC_BANK_NUM" value="4" />
<parameter name="DEBUG_MODE" value="false" />
<parameter name="ENABLE_BURST_MERGE" value="false" />
<parameter name="CTL_ENABLE_BURST_INTERRUPT" value="true" />
<parameter name="CTL_ENABLE_BURST_TERMINATE" value="true" />
<parameter name="LOCAL_ID_WIDTH" value="8" />
<parameter name="WRBUFFER_ADDR_WIDTH" value="6" />
<parameter name="MAX_PENDING_WR_CMD" value="8" />
<parameter name="MAX_PENDING_RD_CMD" value="16" />
<parameter name="USE_MM_ADAPTOR" value="true" />
<parameter name="USE_AXI_ADAPTOR" value="false" />
<parameter name="HCX_COMPAT_MODE" value="false" />
<parameter name="CTL_CMD_QUEUE_DEPTH" value="8" />
<parameter name="CTL_CSR_READ_ONLY" value="1" />
<parameter name="CFG_DATA_REORDERING_TYPE" value="INTER_BANK" />
<parameter name="NUM_OF_PORTS" value="1" />
<parameter name="ENABLE_BONDING" value="false" />
<parameter name="ENABLE_USER_ECC" value="false" />
<parameter name="AVL_DATA_WIDTH_PORT" value="32,32,32,32,32,32" />
<parameter name="PRIORITY_PORT" value="1,1,1,1,1,1" />
<parameter name="WEIGHT_PORT" value="0,0,0,0,0,0" />
<parameter name="CPORT_TYPE_PORT">Bidirectional, Bidirectional, Bidirectional,
    Bidirectional, Bidirectional, Bidirectional</parameter>
<parameter name="ENABLE_EMIT_BFM_MASTER" value="false" />
<parameter name="FORCE_SEQUENCER_TCL_DEBUG_MODE" value="false" />
<parameter name="ENABLE_SEQUENCER_MARGINING_ON_BY_DEFAULT" value="false" />
<parameter name="REF_CLK_FREQ" value="25.0" />
<parameter name="REF_CLK_FREQ_PARAM_VALID" value="false" />
<parameter name="REF_CLK_FREQ_MIN_PARAM" value="0.0" />
<parameter name="REF_CLK_FREQ_MAX_PARAM" value="0.0" />
<parameter name="PLL_DR_CLK_FREQ_PARAM" value="0.0" />
<parameter name="PLL_DR_CLK_FREQ_SIM_STR_PARAM" value="" />
<parameter name="PLL_DR_CLK_PHASE_PS_PARAM" value="0" />
<parameter name="PLL_DR_CLK_PHASE_PS_SIM_STR_PARAM" value="" />
<parameter name="PLL_DR_CLK_MULT_PARAM" value="0" />
<parameter name="PLL_DR_CLK_DIV_PARAM" value="0" />
<parameter name="PLL_MEM_CLK_FREQ_PARAM" value="0.0" />
<parameter name="PLL_MEM_CLK_FREQ_SIM_STR_PARAM" value="" />
<parameter name="PLL_MEM_CLK_PHASE_PS_PARAM" value="0" />
<parameter name="PLL_MEM_CLK_PHASE_PS_SIM_STR_PARAM" value="" />
<parameter name="PLL_MEM_CLK_MULT_PARAM" value="0" />
<parameter name="PLL_MEM_CLK_DIV_PARAM" value="0" />
<parameter name="PLL_AFI_CLK_FREQ_PARAM" value="0.0" />
<parameter name="PLL_AFI_CLK_FREQ_SIM_STR_PARAM" value="" />
<parameter name="PLL_AFI_CLK_PHASE_PS_PARAM" value="0" />
<parameter name="PLL_AFI_CLK_PHASE_PS_SIM_STR_PARAM" value="" />
<parameter name="PLL_AFI_CLK_MULT_PARAM" value="0" />
<parameter name="PLL_AFI_CLK_DIV_PARAM" value="0" />
<parameter name="PLL_WRITE_CLK_FREQ_PARAM" value="0.0" />
<parameter name="PLL_WRITE_CLK_FREQ_SIM_STR_PARAM" value="" />
<parameter name="PLL_WRITE_CLK_PHASE_PS_PARAM" value="0" />
<parameter name="PLL_WRITE_CLK_PHASE_PS_SIM_STR_PARAM" value="" />
<parameter name="PLL_WRITE_CLK_MULT_PARAM" value="0" />
<parameter name="PLL_WRITE_CLK_DIV_PARAM" value="0" />
<parameter name="PLL_ADDR_CMD_CLK_FREQ_PARAM" value="0.0" />
<parameter name="PLL_ADDR_CMD_CLK_FREQ_SIM_STR_PARAM" value="" />
<parameter name="PLL_ADDR_CMD_CLK_PHASE_PS_PARAM" value="0" />
<parameter name="PLL_ADDR_CMD_CLK_PHASE_PS_SIM_STR_PARAM" value="" />
<parameter name="PLL_ADDR_CMD_CLK_MULT_PARAM" value="0" />
<parameter name="PLL_ADDR_CMD_CLK_DIV_PARAM" value="0" />
<parameter name="PLL_AFI_HALF_CLK_FREQ_PARAM" value="0.0" />
<parameter name="PLL_AFI_HALF_CLK_FREQ_SIM_STR_PARAM" value="" />
```

```
<parameter name="PLL_AFI_HALF_CLK_PHASE_PS_PARAM" value="0" />
<parameter name="PLL_AFI_HALF_CLK_PHASE_PS_SIM_STR_PARAM" value="" />
<parameter name="PLL_AFI_HALF_CLK_MULT_PARAM" value="0" />
<parameter name="PLL_AFI_HALF_CLK_DIV_PARAM" value="0" />
<parameter name="PLL_NIOS_CLK_FREQ_PARAM" value="0.0" />
<parameter name="PLL_NIOS_CLK_FREQ_SIM_STR_PARAM" value="" />
<parameter name="PLL_NIOS_CLK_PHASE_PS_PARAM" value="0" />
<parameter name="PLL_NIOS_CLK_PHASE_PS_SIM_STR_PARAM" value="" />
<parameter name="PLL_NIOS_CLK_MULT_PARAM" value="0" />
<parameter name="PLL_NIOS_CLK_DIV_PARAM" value="0" />
<parameter name="PLL_CONFIG_CLK_FREQ_PARAM" value="0.0" />
<parameter name="PLL_CONFIG_CLK_FREQ_SIM_STR_PARAM" value="" />
<parameter name="PLL_CONFIG_CLK_PHASE_PS_PARAM" value="0" />
<parameter name="PLL_CONFIG_CLK_PHASE_PS_SIM_STR_PARAM" value="" />
<parameter name="PLL_CONFIG_CLK_MULT_PARAM" value="0" />
<parameter name="PLL_CONFIG_CLK_DIV_PARAM" value="0" />
<parameter name="PLL_P2C_READ_CLK_FREQ_PARAM" value="0.0" />
<parameter name="PLL_P2C_READ_CLK_FREQ_SIM_STR_PARAM" value="" />
<parameter name="PLL_P2C_READ_CLK_PHASE_PS_PARAM" value="0" />
<parameter name="PLL_P2C_READ_CLK_PHASE_PS_SIM_STR_PARAM" value="" />
<parameter name="PLL_P2C_READ_CLK_MULT_PARAM" value="0" />
<parameter name="PLL_P2C_READ_CLK_DIV_PARAM" value="0" />
<parameter name="PLL_C2P_WRITE_CLK_FREQ_PARAM" value="0.0" />
<parameter name="PLL_C2P_WRITE_CLK_FREQ_SIM_STR_PARAM" value="" />
<parameter name="PLL_C2P_WRITE_CLK_PHASE_PS_PARAM" value="0" />
<parameter name="PLL_C2P_WRITE_CLK_PHASE_PS_SIM_STR_PARAM" value="" />
<parameter name="PLL_C2P_WRITE_CLK_MULT_PARAM" value="0" />
<parameter name="PLL_C2P_WRITE_CLK_DIV_PARAM" value="0" />
<parameter name="PLL_HR_CLK_FREQ_PARAM" value="0.0" />
<parameter name="PLL_HR_CLK_FREQ_SIM_STR_PARAM" value="" />
<parameter name="PLL_HR_CLK_PHASE_PS_PARAM" value="0" />
<parameter name="PLL_HR_CLK_PHASE_PS_SIM_STR_PARAM" value="" />
<parameter name="PLL_HR_CLK_MULT_PARAM" value="0" />
<parameter name="PLL_HR_CLK_DIV_PARAM" value="0" />
<parameter name="PLL_AFI_PHY_CLK_FREQ_PARAM" value="0.0" />
<parameter name="PLL_AFI_PHY_CLK_FREQ_SIM_STR_PARAM" value="" />
<parameter name="PLL_AFI_PHY_CLK_PHASE_PS_PARAM" value="0" />
<parameter name="PLL_AFI_PHY_CLK_PHASE_PS_SIM_STR_PARAM" value="" />
<parameter name="PLL_AFI_PHY_CLK_MULT_PARAM" value="0" />
<parameter name="PLL_AFI_PHY_CLK_DIV_PARAM" value="0" />
<parameter name="PLL_CLK_PARAM_VALID" value="false" />
<parameter name="ENABLE_EXTRA_REPORTING" value="false" />
<parameter name="NUM_EXTRA_REPORT_PATH" value="10" />
<parameter name="ENABLE_ISS_PROBES" value="false" />
<parameter name="CALIB_REG_WIDTH" value="8" />
<parameter name="USE_SEQUENCER_BFM" value="false" />
<parameter name="DEFAULT_FAST_SIM_MODEL" value="true" />
<parameter name="PLL_SHARING_MODE" value="None" />
<parameter name="NUM_PLL_SHARING_INTERFACES" value="1" />
<parameter name="EXPORT_AFI_HALF_CLK" value="false" />
<parameter name="ABSTRACT_REAL_COMPARE_TEST" value="false" />
<parameter name="INCLUDE_BOARD_DELAY_MODEL" value="false" />
<parameter name="INCLUDE_MULTIRANK_BOARD_DELAY_MODEL" value="false" />
<parameter name="USE_FAKE_PHY" value="false" />
<parameter name="FORCE_MAX_LATENCY_COUNT_WIDTH" value="0" />
<parameter name="ENABLE_NON_DESTRUCTIVE_CALIB" value="false" />
<parameter name="TRACKING_ERROR_TEST" value="false" />
<parameter name="TRACKING_WATCH_TEST" value="false" />
<parameter name="MARGIN_VARIATION_TEST" value="false" />
<parameter name="EXTRA_SETTINGS" value="" />
<parameter name="MEM_DEVICE" value="MISSING_MODEL" />
<parameter name="FORCE_SYNTHESIS_LANGUAGE" value="" />
<parameter name="FORCED_NUM_WRITE_FR_CYCLE_SHIFTS" value="0" />
<parameter name="SEQUENCER_TYPE" value="NIOS" />
<parameter name="ADVERTIZE_SEQUENCER_SW_BUILD_FILES" value="false" />
<parameter name="FORCED_NON_LDC_ADDR_CMD_MEM_CK_INVERT" value="false" />
<parameter name="PHY_ONLY" value="false" />
<parameter name="SEQ_MODE" value="0" />
```

```
<parameter name="ADVANCED_CK_PHASES" value="false" />
<parameter name="COMMAND_PHASE" value="0.0" />
<parameter name="MEM_CK_PHASE" value="0.0" />
<parameter name="P2C_READ_CLOCK_ADD_PHASE" value="0.0" />
<parameter name="C2P_WRITE_CLOCK_ADD_PHASE" value="0.0" />
<parameter name="ACV_PHY_CLK_ADD_FR_PHASE" value="0.0" />
<parameter name="MEM_VOLTAGE" value="1.5V DDR3" />
<parameter name="PLL_LOCATION" value="Top_Bottom" />
<parameter name="SKIP_MEM_INIT" value="true" />
<parameter name="READ_DQ_DQS_CLOCK_SOURCE" value="INVERTED_DQS_BUS" />
<parameter name="DQ_INPUT_REG_USE_CLKN" value="false" />
<parameter name="DQS_DQSN_MODE" value="DIFFERENTIAL" />
<parameter name="AFI_DEBUG_INFO_WIDTH" value="32" />
<parameter name="CALIBRATION_MODE" value="Skip" />
<parameter name="NIOS_ROM_DATA_WIDTH" value="32" />
<parameter name="READ_FIFO_SIZE" value="8" />
<parameter name="PHY_CSR_ENABLED" value="false" />
<parameter name="PHY_CSR_CONNECTION" value="INTERNAL_JTAG" />
<parameter name="USER_DEBUG_LEVEL" value="1" />
<parameter name="TIMING_BOARD_DERATE_METHOD" value="AUTO" />
<parameter name="TIMING_BOARD_CK_CKN_SLEW_RATE" value="2.0" />
<parameter name="TIMING_BOARD_AC_SLEW_RATE" value="1.0" />
<parameter name="TIMING_BOARD_DQS_DQSN_SLEW_RATE" value="2.0" />
<parameter name="TIMING_BOARD_DQ_SLEW_RATE" value="1.0" />
<parameter name="TIMING_BOARD_TIS" value="0.0" />
<parameter name="TIMING_BOARD_TIH" value="0.0" />
<parameter name="TIMING_BOARD_TDS" value="0.0" />
<parameter name="TIMING_BOARD_TDH" value="0.0" />
<parameter name="TIMING_BOARD_ISI_METHOD" value="AUTO" />
<parameter name="TIMING_BOARD_AC_EYE_REDUCTION_SU" value="0.0" />
<parameter name="TIMING_BOARD_AC_EYE_REDUCTION_H" value="0.0" />
<parameter name="TIMING_BOARD_DQ_EYE_REDUCTION" value="0.0" />
<parameter name="TIMING_BOARD_DELTA_DQS_ARRIVAL_TIME" value="0.0" />
<parameter name="PACKAGE_DESKEW" value="false" />
<parameter name="AC_PACKAGE_DESKEW" value="false" />
<parameter name="TIMING_BOARD_MAX_CK_DELAY" value="0.6" />
<parameter name="TIMING_BOARD_MAX_DQS_DELAY" value="0.6" />
<parameter name="TIMING_BOARD_SKEW_CKDQS_DIMM_MIN" value="-0.01" />
<parameter name="TIMING_BOARD_SKEW_CKDQS_DIMM_MAX" value="0.01" />
<parameter name="TIMING_BOARD_SKEW_BETWEEN_DIMMS" value="0.05" />
<parameter name="TIMING_BOARD_SKEW_WITHIN_DQS" value="0.02" />
<parameter name="TIMING_BOARD_SKEW_BETWEEN_DQS" value="0.02" />
<parameter name="TIMING_BOARD_DQ_TO_DQS_SKEW" value="0.0" />
<parameter name="TIMING_BOARD_AC_SKEW" value="0.02" />
<parameter name="TIMING_BOARD_AC_TO_CK_SKEW" value="0.0" />
<parameter name="RATE" value="Full" />
<parameter name="MEM_CLK_FREQ" value="400.0" />
<parameter name="USE_MEM_CLK_FREQ" value="false" />
<parameter name="FORCE_DQS_TRACKING" value="AUTO" />
<parameter name="FORCE_SHADOW_REGS" value="AUTO" />
<parameter name="MRS_MIRROR_PING_PONG_ATSO" value="false" />
<parameter name="SYS_INFO_DEVICE_FAMILY" value="Cyclone V" />
<parameter name="PARSE_FRIENDLY_DEVICE_FAMILY_PARAM_VALID" value="false" />
<parameter name="PARSE_FRIENDLY_DEVICE_FAMILY_PARAM" value="" />
<parameter name="DEVICE_FAMILY_PARAM" value="" />
<parameter name="SPEED_GRADE" value="7" />
<parameter name="IS_ES_DEVICE" value="false" />
<parameter name="DISABLE_CHILD_MESSAGING" value="false" />
<parameter name="HARD_EMIF" value="true" />
<parameter name="HHP_HPS" value="true" />
<parameter name="HHP_HPS_VERIFICATION" value="false" />
<parameter name="HHP_HPS_SIMULATION" value="false" />
<parameter name="HPS_PROTOCOL" value="DDR3" />
<parameter name="CUT_NEW_FAMILY_TIMING" value="true" />
<parameter name="ENABLE_EXPORT_SEQ_DEBUG_BRIDGE" value="false" />
<parameter name="CORE_DEBUG_CONNECTION" value="EXPORT" />
<parameter name="ADD_EXTERNAL_SEQ_DEBUG_NIOS" value="false" />
<parameter name="ED_EXPORT_SEQ_DEBUG" value="false" />
```

```
678   <parameter name="ADD_EFFICIENCY_MONITOR" value="false" />
679   <parameter name="ENABLE_ABS_RAM_MEM_INIT" value="false" />
680   <parameter name="ABS_RAM_MEM_INIT_FILENAME" value="meminit" />
681   <parameter name="DLL_SHARING_MODE" value="None" />
682   <parameter name="NUM_DLL_SHARING_INTERFACES" value="1" />
683   <parameter name="OCT_SHARING_MODE" value="None" />
684   <parameter name="NUM_OCT_SHARING_INTERFACES" value="1" />
685   <parameter name="MPU_EVENTS_Enable" value="false" />
686   <parameter name="GP_Enable" value="false" />
687   <parameter name="DEBUGAPB_Enable" value="false" />
688   <parameter name="STM_Enable" value="false" />
689   <parameter name="CTI_Enable" value="false" />
690   <parameter name="TPIUFPGA_Enable" value="false" />
691   <parameter name="BOOTFROMFPGA_Enable" value="false" />
692   <parameter name="TEST_Enable" value="false" />
693   <parameter name="HLGPI_Enable" value="false" />
694   <parameter name="BSEL_EN" value="false" />
695   <parameter name="BSEL" value="1" />
696   <parameter name="CSEL_EN" value="false" />
697   <parameter name="CSEL" value="0" />
698   <parameter name="F2S_Width" value="2" />
699   <parameter name="S2F_Width" value="2" />
700   <parameter name="LWH2F_Enable" value="true" />
701   <parameter name="F2SDRAM_Type" value="" />
702   <parameter name="F2SDRAM_Width" value="" />
703   <parameter name="S2FCLK_COLDRST_Enable" value="false" />
704   <parameter name="S2FCLK_PENDINGRST_Enable" value="false" />
705   <parameter name="F2SCLK_DBGRST_Enable" value="false" />
706   <parameter name="F2SCLK_WARMRST_Enable" value="false" />
707   <parameter name="F2SCLK_COLDRST_Enable" value="false" />
708   <parameter name="DMA_Enable">No,No,No,No,No,No,No,No</parameter>
709   <parameter name="F2SINTERRUPT_Enable" value="true" />
710   <parameter name="S2FINTERRUPT_CAN_Enable" value="false" />
711   <parameter name="S2FINTERRUPT_CLOCKPERIPHERAL_Enable" value="false" />
712   <parameter name="S2FINTERRUPT_CTI_Enable" value="false" />
713   <parameter name="S2FINTERRUPT_DMA_Enable" value="false" />
714   <parameter name="S2FINTERRUPT_EMAC_Enable" value="false" />
715   <parameter name="S2FINTERRUPT_FPGAMANAGER_Enable" value="false" />
716   <parameter name="S2FINTERRUPT_GPIO_Enable" value="false" />
717   <parameter name="S2FINTERRUPT_I2CEMAC_Enable" value="false" />
718   <parameter name="S2FINTERRUPT_I2CPERIPHERAL_Enable" value="false" />
719   <parameter name="S2FINTERRUPT_L4TIMER_Enable" value="false" />
720   <parameter name="S2FINTERRUPT_NAND_Enable" value="false" />
721   <parameter name="S2FINTERRUPT_OSCTIMER_Enable" value="false" />
722   <parameter name="S2FINTERRUPT_QSPI_Enable" value="false" />
723   <parameter name="S2FINTERRUPT_SDMMC_Enable" value="false" />
724   <parameter name="S2FINTERRUPT_SPIMASTER_Enable" value="false" />
725   <parameter name="S2FINTERRUPT_SPISLAVE_Enable" value="false" />
726   <parameter name="S2FINTERRUPT_UART_Enable" value="false" />
727   <parameter name="S2FINTERRUPT_USB_Enable" value="false" />
728   <parameter name="S2FINTERRUPT_WATCHDOG_Enable" value="false" />
729   <parameter name="EMAC0_PinMuxing" value="Unused" />
730   <parameter name="EMAC0_Mode" value="N/A" />
731   <parameter name="EMAC1_PinMuxing" value="HPS I/O Set 0" />
732   <parameter name="EMAC1_Mode" value="RGMII" />
733   <parameter name="NAND_PinMuxing" value="Unused" />
734   <parameter name="NAND_Mode" value="N/A" />
735   <parameter name="QSPI_PinMuxing" value="HPS I/O Set 0" />
736   <parameter name="QSPI_Mode" value="1 SS" />
737   <parameter name="SDIO_PinMuxing" value="HPS I/O Set 0" />
738   <parameter name="SDIO_Mode" value="4-bit Data" />
739   <parameter name="USB0_PinMuxing" value="Unused" />
740   <parameter name="USB0_Mode" value="N/A" />
741   <parameter name="USB1_PinMuxing" value="HPS I/O Set 0" />
742   <parameter name="USB1_Mode" value="SDR" />
743   <parameter name="SPIM0_PinMuxing" value="HPS I/O Set 0" />
744   <parameter name="SPIM0_Mode" value="Single Slave Select" />
745   <parameter name="SPIM1_PinMuxing" value="Unused" />
```

```
<parameter name="SPIM1_Mode" value="N/A" />
<parameter name="SPIS0_PinMuxing" value="Unused" />
<parameter name="SPIS0_Mode" value="N/A" />
<parameter name="SPIS1_PinMuxing" value="Unused" />
<parameter name="SPIS1_Mode" value="N/A" />
<parameter name="UART0_PinMuxing" value="HPS I/O Set 2" />
<parameter name="UART0_Mode" value="No Flow Control" />
<parameter name="UART1_PinMuxing" value="Unused" />
<parameter name="UART1_Mode" value="N/A" />
<parameter name="I2C0_PinMuxing" value="HPS I/O Set 1" />
<parameter name="I2C0_Mode" value="I2C" />
<parameter name="I2C1_PinMuxing" value="Unused" />
<parameter name="I2C1_Mode" value="N/A" />
<parameter name="I2C2_PinMuxing" value="Unused" />
<parameter name="I2C2_Mode" value="N/A" />
<parameter name="I2C3_PinMuxing" value="Unused" />
<parameter name="I2C3_Mode" value="N/A" />
<parameter name="CAN0_PinMuxing" value="HPS I/O Set 0" />
<parameter name="CAN0_Mode" value="CAN" />
<parameter name="CAN1_PinMuxing" value="Unused" />
<parameter name="CAN1_Mode" value="N/A" />
<parameter name="TRACE_PinMuxing" value="HPS I/O Set 0" />
<parameter name="TRACE_Mode" value="HPS" />
<parameter name="GPIO_Enable">No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,
    No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,
    No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,
    No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No</
    parameter>
<parameter name="LOANIO_Enable">No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No
    ,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No
    ,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No
    ,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No</
    parameter>
<parameter name="S2FCLK_USER0CLK_Enable" value="false" />
<parameter name="S2FCLK_USER0CLK_FREQ" value="50" />
<parameter name="S2FCLK_USER1CLK_Enable" value="false" />
<parameter name="S2FCLK_USER1CLK_FREQ" value="100" />
<parameter name="S2FCLK_USER2CLK_Enable" value="false" />
<parameter name="S2FCLK_USER2CLK_FREQ" value="100" />
<parameter name="F2SCLK_PERIPHCLK_Enable" value="false" />
<parameter name="F2SCLK_PERIPHCLK_FREQ" value="100" />
<parameter name="F2SCLK_SDRAMCLK_Enable" value="false" />
<parameter name="F2SCLK_SDRAMCLK_FREQ" value="100" />
<parameter name="F2H_AXI_CLOCK_FREQ" value="50000000" />
<parameter name="H2F_AXI_CLOCK_FREQ" value="50000000" />
<parameter name="H2F_LW_AXI_CLOCK_FREQ" value="50000000" />
<parameter name="F2H_SDRAM0_CLOCK_FREQ" value="100" />
<parameter name="F2H_SDRAM1_CLOCK_FREQ" value="100" />
<parameter name="F2H_SDRAM2_CLOCK_FREQ" value="100" />
<parameter name="F2H_SDRAM3_CLOCK_FREQ" value="100" />
<parameter name="F2H_SDRAM4_CLOCK_FREQ" value="100" />
<parameter name="F2H_SDRAM5_CLOCK_FREQ" value="100" />
<parameter name="H2F_CTI_CLOCK_FREQ" value="100" />
<parameter name="H2F_TPIU_CLOCK_IN_FREQ" value="100" />
<parameter name="H2F_DEBUG_APB_CLOCK_FREQ" value="100" />
<parameter
    name="FPGA_PERIPHERAL_INPUT_CLOCK_FREQ_EMAC_PTP_REF_CLOCK"
    value="100" />
<parameter name="FPGA_PERIPHERAL_INPUT_CLOCK_FREQ_EMAC0_RX_CLK_IN" value="100" />
<parameter name="FPGA_PERIPHERAL_INPUT_CLOCK_FREQ_EMAC0_TX_CLK_IN" value="100" />
<parameter name="FPGA_PERIPHERAL_OUTPUT_CLOCK_FREQ_EMAC0_MD_CLK" value="100" />
<parameter name="FPGA_PERIPHERAL_OUTPUT_CLOCK_FREQ_EMAC0_GTX_CLK" value="100" />
<parameter name="FPGA_PERIPHERAL_INPUT_CLOCK_FREQ_EMAC1_RX_CLK_IN" value="100" />
<parameter name="FPGA_PERIPHERAL_INPUT_CLOCK_FREQ_EMAC1_TX_CLK_IN" value="100" />
<parameter name="FPGA_PERIPHERAL_OUTPUT_CLOCK_FREQ_EMAC1_MD_CLK" value="100" />
<parameter name="FPGA_PERIPHERAL_OUTPUT_CLOCK_FREQ_EMAC1_GTX_CLK" value="100" />
<parameter name="FPGA_PERIPHERAL_OUTPUT_CLOCK_FREQ_QSPI_SCLK_OUT" value="100" />
<parameter name="FPGA_PERIPHERAL_INPUT_CLOCK_FREQ_SDIO_CLK_IN" value="100" />
```

```
  <parameter name="FPGA_PERIPHERAL_OUTPUT_CLOCK_FREQ_SDIO_CCLK" value="100" />
  <parameter name="FPGA_PERIPHERAL_INPUT_CLOCK_FREQ_USB0_CLK_IN" value="100" />
  <parameter name="FPGA_PERIPHERAL_INPUT_CLOCK_FREQ_USB1_CLK_IN" value="100" />
  <parameter name="FPGA_PERIPHERAL_OUTPUT_CLOCK_FREQ_SPIM0_SCLK_OUT" value="100" />
  <parameter name="FPGA_PERIPHERAL_OUTPUT_CLOCK_FREQ_SPIM1_SCLK_OUT" value="100" />
  <parameter name="FPGA_PERIPHERAL_INPUT_CLOCK_FREQ_SPIS0_SCLK_IN" value="100" />
  <parameter name="FPGA_PERIPHERAL_INPUT_CLOCK_FREQ_SPIS1_SCLK_IN" value="100" />
  <parameter name="FPGA_PERIPHERAL_INPUT_CLOCK_FREQ_I2C0_SCL_IN" value="100" />
  <parameter name="FPGA_PERIPHERAL_OUTPUT_CLOCK_FREQ_I2C0_CLK" value="100" />
  <parameter name="FPGA_PERIPHERAL_INPUT_CLOCK_FREQ_I2C1_SCL_IN" value="100" />
  <parameter name="FPGA_PERIPHERAL_OUTPUT_CLOCK_FREQ_I2C1_CLK" value="100" />
  <parameter name="FPGA_PERIPHERAL_INPUT_CLOCK_FREQ_I2C2_SCL_IN" value="100" />
  <parameter name="FPGA_PERIPHERAL_OUTPUT_CLOCK_FREQ_I2C2_CLK" value="100" />
  <parameter name="FPGA_PERIPHERAL_INPUT_CLOCK_FREQ_I2C3_SCL_IN" value="100" />
  <parameter name="FPGA_PERIPHERAL_OUTPUT_CLOCK_FREQ_I2C3_CLK" value="100" />
  <parameter name="device_name" value="5CSXFC6D6F31C8ES" />
  <parameter
     name="quartus_ini_hps_ip_enable_all_peripheral_fpga_interfaces"
     value="false" />
  <parameter
     name="quartus_ini_hps_ip_enable_emac0_peripheral_fpga_interface"
     value="false" />
  <parameter name="quartus_ini_hps_ip_enable_test_interface" value="false" />
  <parameter name="quartus_ini_hps_ip_fast_f2sdram_sim_model" value="false" />
  <parameter name="quartus_ini_hps_ip_suppress_sdram_synth" value="false" />
  <parameter name="quartus_ini_hps_ip_enable_loanio" value="false" />
  <parameter
     name="quartus_ini_hps_ip_enable_low_speed_serial_fpga_interfaces"
     value="false" />
  <parameter name="quartus_ini_hps_ip_enable_bsel_csel" value="false" />
 </module>
 <module
    kind="altera_jtag_avalon_master"
    version="13.0"
    enabled="1"
    name="fpga_only_master">
  <parameter name="USE_PLI" value="0" />
  <parameter name="PLI_PORT" value="50000" />
  <parameter name="COMPONENT_CLOCK" value="0" />
  <parameter name="FAST_VER" value="0" />
  <parameter name="FIFO_DEPTHS" value="2" />
  <parameter name="AUTO_DEVICE_FAMILY" value="Cyclone V" />
  <parameter name="AUTO_DEVICE" value="5CSXFC6D6F31C8ES" />
 </module>
 <module
    kind="intr_capturer"
    version="100.99.98.97"
    enabled="1"
    name="intr_capturer_0">
  <parameter name="NUM_INTR" value="32" />
  <parameter name="AUTO_CLOCK_CLOCK_RATE" value="50000000" />
 </module>
 <module kind="clock_source" version="13.0" enabled="1" name="ext_clk_50">
  <parameter name="clockFrequency" value="50000000" />
  <parameter name="clockFrequencyKnown" value="true" />
  <parameter name="inputClockFrequency" value="0" />
  <parameter name="resetSynchronousEdges" value="NONE" />
 </module>
 <module
    kind="altera_jtag_avalon_master"
    version="13.0"
    enabled="1"
    name="hps_only_master">
  <parameter name="USE_PLI" value="0" />
  <parameter name="PLI_PORT" value="50000" />
  <parameter name="COMPONENT_CLOCK" value="0" />
  <parameter name="FAST_VER" value="0" />
  <parameter name="FIFO_DEPTHS" value="2" />
```

```
 <parameter name="AUTO_DEVICE_FAMILY" value="Cyclone V" />
 <parameter name="AUTO_DEVICE" value="5CSXFC6D6F31C8ES" />
</module>
<module
  kind="altera_avalon_jtag_uart"
  version="13.0.1.99.2"
  enabled="1"
  name="jtag_uart">
 <parameter name="allowMultipleConnections" value="true" />
 <parameter name="hubInstanceID" value="0" />
 <parameter name="readBufferDepth" value="64" />
 <parameter name="readIRQThreshold" value="8" />
 <parameter name="simInputCharacterStream" value="" />
 <parameter name="simInteractiveOptions">INTERACTIVE_ASCII_OUTPUT</parameter>
 <parameter name="useRegistersForReadBuffer" value="false" />
 <parameter name="useRegistersForWriteBuffer" value="false" />
 <parameter name="useRelativePathForSimFile" value="false" />
 <parameter name="writeBufferDepth" value="64" />
 <parameter name="writeIRQThreshold" value="8" />
 <parameter name="avalonSpec" value="2.0" />
</module>
<module
  kind="altera_avalon_pio"
  version="13.0.1.99.2"
  enabled="1"
  name="button_pio">
 <parameter name="bitClearingEdgeCapReg" value="true" />
 <parameter name="bitModifyingOutReg" value="false" />
 <parameter name="captureEdge" value="true" />
 <parameter name="direction" value="Input" />
 <parameter name="edgeType" value="FALLING" />
 <parameter name="generateIRQ" value="true" />
 <parameter name="irqType" value="EDGE" />
 <parameter name="resetValue" value="0" />
 <parameter name="simDoTestBenchWiring" value="false" />
 <parameter name="simDrivenValue" value="0" />
 <parameter name="width" value="2" />
 <parameter name="clockRate" value="50000000" />
</module>
<module
  kind="altera_avalon_pio"
  version="13.0.1.99.2"
  enabled="1"
  name="dipsw_pio">
 <parameter name="bitClearingEdgeCapReg" value="true" />
 <parameter name="bitModifyingOutReg" value="false" />
 <parameter name="captureEdge" value="true" />
 <parameter name="direction" value="Input" />
 <parameter name="edgeType" value="ANY" />
 <parameter name="generateIRQ" value="true" />
 <parameter name="irqType" value="EDGE" />
 <parameter name="resetValue" value="0" />
 <parameter name="simDoTestBenchWiring" value="false" />
 <parameter name="simDrivenValue" value="0" />
 <parameter name="width" value="4" />
 <parameter name="clockRate" value="50000000" />
</module>
<module
  kind="altera_avalon_pio"
  version="13.0.1.99.2"
  enabled="1"
  name="led_pio">
 <parameter name="bitClearingEdgeCapReg" value="false" />
 <parameter name="bitModifyingOutReg" value="true" />
 <parameter name="captureEdge" value="false" />
 <parameter name="direction" value="InOut" />
 <parameter name="edgeType" value="RISING" />
 <parameter name="generateIRQ" value="false" />
```

```
  <parameter name="irqType" value="LEVEL" />
  <parameter name="resetValue" value="0" />
  <parameter name="simDoTestBenchWiring" value="false" />
  <parameter name="simDrivenValue" value="0" />
  <parameter name="width" value="4" />
  <parameter name="clockRate" value="50000000" />
 </module>
 <module
    kind="altera_mem_if_ddr3_emif"
    version="13.0"
    enabled="1"
    name="mem_if_ddr3_fpga">
  <parameter name="MEM_VENDOR" value="Micron" />
  <parameter name="MEM_FORMAT" value="DISCRETE" />
  <parameter name="RDIMM_CONFIG" value="0000000000000000" />
  <parameter name="LRDIMM_EXTENDED_CONFIG">0x0000000000000000000</parameter>
  <parameter name="DISCRETE_FLY_BY" value="true" />
  <parameter name="DEVICE_DEPTH" value="1" />
  <parameter name="MEM_MIRROR_ADDRESSING" value="0" />
  <parameter name="MEM_CLK_FREQ_MAX" value="800.0" />
  <parameter name="MEM_ROW_ADDR_WIDTH" value="15" />
  <parameter name="MEM_COL_ADDR_WIDTH" value="8" />
  <parameter name="MEM_DQ_WIDTH" value="32" />
  <parameter name="MEM_DQ_PER_DQS" value="8" />
  <parameter name="MEM_BANKADDR_WIDTH" value="3" />
  <parameter name="MEM_IF_DM_PINS_EN" value="true" />
  <parameter name="MEM_IF_DQSN_EN" value="true" />
  <parameter name="MEM_NUMBER_OF_DIMMS" value="1" />
  <parameter name="MEM_NUMBER_OF_RANKS_PER_DIMM" value="1" />
  <parameter name="MEM_NUMBER_OF_RANKS_PER_DEVICE" value="1" />
  <parameter name="MEM_RANK_MULTIPLICATION_FACTOR" value="1" />
  <parameter name="MEM_CK_WIDTH" value="1" />
  <parameter name="MEM_CS_WIDTH" value="1" />
  <parameter name="MEM_CLK_EN_WIDTH" value="1" />
  <parameter name="ALTMEMPHY_COMPATIBLE_MODE" value="false" />
  <parameter name="NEXTGEN" value="true" />
  <parameter name="MEM_IF_BOARD_BASE_DELAY" value="10" />
  <parameter name="MEM_IF_SIM_VALID_WINDOW" value="0" />
  <parameter name="MEM_GUARANTEED_WRITE_INIT" value="false" />
  <parameter name="MEM_VERBOSE" value="true" />
  <parameter name="PINGPONGPHY_EN" value="false" />
  <parameter name="REFRESH_BURST_VALIDATION" value="false" />
  <parameter name="MEM_BL" value="OTF" />
  <parameter name="MEM_BT" value="Sequential" />
  <parameter name="MEM_ASR" value="Manual" />
  <parameter name="MEM_SRT" value="Normal" />
  <parameter name="MEM_PD" value="DLL off" />
  <parameter name="MEM_DRV_STR" value="RZQ/6" />
  <parameter name="MEM_DLL_EN" value="true" />
  <parameter name="MEM_RTT_NOM" value="RZQ/6" />
  <parameter name="MEM_RTT_WR" value="RZQ/4" />
  <parameter name="MEM_WTCL" value="5" />
  <parameter name="MEM_ATCL" value="Disabled" />
  <parameter name="MEM_TCL" value="6" />
  <parameter name="MEM_AUTO_LEVELING_MODE" value="true" />
  <parameter name="MEM_USER_LEVELING_MODE" value="Leveling" />
  <parameter name="MEM_INIT_EN" value="false" />
  <parameter name="MEM_INIT_FILE" value="" />
  <parameter name="DAT_DATA_WIDTH" value="32" />
  <parameter name="TIMING_TIS" value="350" />
  <parameter name="TIMING_TIH" value="275" />
  <parameter name="TIMING_TDS" value="125" />
  <parameter name="TIMING_TDH" value="150" />
  <parameter name="TIMING_TDQSQ" value="200" />
  <parameter name="TIMING_TQH" value="0.38" />
  <parameter name="TIMING_TDQSCK" value="400" />
  <parameter name="TIMING_TDQSCKDS" value="450" />
  <parameter name="TIMING_TDQSCKDM" value="900" />
```

```
<parameter name="TIMING_TDQSCKDL" value="1200" />
<parameter name="TIMING_TDQSS" value="0.25" />
<parameter name="TIMING_TQSH" value="0.38" />
<parameter name="TIMING_TDSH" value="0.2" />
<parameter name="TIMING_TDSS" value="0.2" />
<parameter name="MEM_TINIT_US" value="500" />
<parameter name="MEM_TMRD_CK" value="4" />
<parameter name="MEM_TRAS_NS" value="35.0" />
<parameter name="MEM_TRCD_NS" value="13.75" />
<parameter name="MEM_TRP_NS" value="13.75" />
<parameter name="MEM_TREFI_US" value="7.8" />
<parameter name="MEM_TRFC_NS" value="260.0" />
<parameter name="CFG_TCCD_NS" value="2.5" />
<parameter name="MEM_TWR_NS" value="15.0" />
<parameter name="MEM_TWTR" value="4" />
<parameter name="MEM_TFAW_NS" value="40.0" />
<parameter name="MEM_TRRD_NS" value="7.5" />
<parameter name="MEM_TRTP_NS" value="7.5" />
<parameter name="RATE" value="Full" />
<parameter name="MEM_CLK_FREQ" value="400.0" />
<parameter name="USE_MEM_CLK_FREQ" value="false" />
<parameter name="FORCE_DQS_TRACKING" value="AUTO" />
<parameter name="FORCE_SHADOW_REGS" value="AUTO" />
<parameter name="MRS_MIRROR_PING_PONG_ATSO" value="false" />
<parameter name="SYS_INFO_DEVICE_FAMILY" value="Cyclone V" />
<parameter name="PARSE_FRIENDLY_DEVICE_FAMILY_PARAM_VALID" value="false" />
<parameter name="PARSE_FRIENDLY_DEVICE_FAMILY_PARAM" value="" />
<parameter name="DEVICE_FAMILY_PARAM" value="" />
<parameter name="SPEED_GRADE" value="7" />
<parameter name="IS_ES_DEVICE" value="false" />
<parameter name="DISABLE_CHILD_MESSAGING" value="false" />
<parameter name="HARD_EMIF" value="true" />
<parameter name="HHP_HPS" value="false" />
<parameter name="HHP_HPS_VERIFICATION" value="false" />
<parameter name="HHP_HPS_SIMULATION" value="false" />
<parameter name="HPS_PROTOCOL" value="DEFAULT" />
<parameter name="CUT_NEW_FAMILY_TIMING" value="true" />
<parameter name="POWER_OF_TWO_BUS" value="true" />
<parameter name="SOPC_COMPAT_RESET" value="false" />
<parameter name="AVL_MAX_SIZE" value="4" />
<parameter name="BYTE_ENABLE" value="true" />
<parameter name="ENABLE_CTRL_AVALON_INTERFACE" value="true" />
<parameter name="CTL_DEEP_POWERDN_EN" value="false" />
<parameter name="CTL_SELF_REFRESH_EN" value="false" />
<parameter name="AUTO_POWERDN_EN" value="false" />
<parameter name="AUTO_PD_CYCLES" value="0" />
<parameter name="CTL_USR_REFRESH_EN" value="false" />
<parameter name="CTL_AUTOPCH_EN" value="false" />
<parameter name="CTL_ZQCAL_EN" value="false" />
<parameter name="ADDR_ORDER" value="0" />
<parameter name="CTL_LOOK_AHEAD_DEPTH" value="4" />
<parameter name="CONTROLLER_LATENCY" value="5" />
<parameter name="CFG_REORDER_DATA" value="false" />
<parameter name="STARVE_LIMIT" value="10" />
<parameter name="CTL_CSR_ENABLED" value="false" />
<parameter name="CTL_CSR_CONNECTION" value="INTERNAL_JTAG" />
<parameter name="CTL_ECC_ENABLED" value="false" />
<parameter name="CTL_HRB_ENABLED" value="false" />
<parameter name="CTL_ECC_AUTO_CORRECTION_ENABLED" value="false" />
<parameter name="MULTICAST_EN" value="false" />
<parameter name="CTL_DYNAMIC_BANK_ALLOCATION" value="false" />
<parameter name="CTL_DYNAMIC_BANK_NUM" value="4" />
<parameter name="DEBUG_MODE" value="false" />
<parameter name="ENABLE_BURST_MERGE" value="false" />
<parameter name="CTL_ENABLE_BURST_INTERRUPT" value="false" />
<parameter name="CTL_ENABLE_BURST_TERMINATE" value="false" />
<parameter name="LOCAL_ID_WIDTH" value="8" />
<parameter name="WRBUFFER_ADDR_WIDTH" value="6" />
```

```
<parameter name="MAX_PENDING_WR_CMD" value="8" />
<parameter name="MAX_PENDING_RD_CMD" value="16" />
<parameter name="USE_MM_ADAPTOR" value="true" />
<parameter name="USE_AXI_ADAPTOR" value="false" />
<parameter name="HCX_COMPAT_MODE" value="false" />
<parameter name="CTL_CMD_QUEUE_DEPTH" value="8" />
<parameter name="CTL_CSR_READ_ONLY" value="1" />
<parameter name="CFG_DATA_REORDERING_TYPE" value="INTER_BANK" />
<parameter name="NUM_OF_PORTS" value="1" />
<parameter name="ENABLE_BONDING" value="false" />
<parameter name="ENABLE_USER_ECC" value="false" />
<parameter name="AVL_DATA_WIDTH_PORT" value="256,32,32,32,32,32" />
<parameter name="PRIORITY_PORT" value="1,1,1,1,1,1" />
<parameter name="WEIGHT_PORT" value="0,0,0,0,0,0" />
<parameter name="CPORT_TYPE_PORT">Bidirectional,Bidirectional,
    Bidirectional,Bidirectional,Bidirectional</parameter>
<parameter name="ENABLE_EMIT_BFM_MASTER" value="false" />
<parameter name="FORCE_SEQUENCER_TCL_DEBUG_MODE" value="false" />
<parameter name="ENABLE_SEQUENCER_MARGINING_ON_BY_DEFAULT" value="false" />
<parameter name="REF_CLK_FREQ" value="100.0" />
<parameter name="REF_CLK_FREQ_PARAM_VALID" value="false" />
<parameter name="REF_CLK_FREQ_MIN_PARAM" value="0.0" />
<parameter name="REF_CLK_FREQ_MAX_PARAM" value="0.0" />
<parameter name="PLL_DR_CLK_FREQ_PARAM" value="0.0" />
<parameter name="PLL_DR_CLK_FREQ_SIM_STR_PARAM" value="" />
<parameter name="PLL_DR_CLK_PHASE_PS_PARAM" value="0" />
<parameter name="PLL_DR_CLK_PHASE_PS_SIM_STR_PARAM" value="" />
<parameter name="PLL_DR_CLK_MULT_PARAM" value="0" />
<parameter name="PLL_DR_CLK_DIV_PARAM" value="0" />
<parameter name="PLL_MEM_CLK_FREQ_PARAM" value="0.0" />
<parameter name="PLL_MEM_CLK_FREQ_SIM_STR_PARAM" value="" />
<parameter name="PLL_MEM_CLK_PHASE_PS_PARAM" value="0" />
<parameter name="PLL_MEM_CLK_PHASE_PS_SIM_STR_PARAM" value="" />
<parameter name="PLL_MEM_CLK_MULT_PARAM" value="0" />
<parameter name="PLL_MEM_CLK_DIV_PARAM" value="0" />
<parameter name="PLL_AFI_CLK_FREQ_PARAM" value="0.0" />
<parameter name="PLL_AFI_CLK_FREQ_SIM_STR_PARAM" value="" />
<parameter name="PLL_AFI_CLK_PHASE_PS_PARAM" value="0" />
<parameter name="PLL_AFI_CLK_PHASE_PS_SIM_STR_PARAM" value="" />
<parameter name="PLL_AFI_CLK_MULT_PARAM" value="0" />
<parameter name="PLL_AFI_CLK_DIV_PARAM" value="0" />
<parameter name="PLL_WRITE_CLK_FREQ_PARAM" value="0.0" />
<parameter name="PLL_WRITE_CLK_FREQ_SIM_STR_PARAM" value="" />
<parameter name="PLL_WRITE_CLK_PHASE_PS_PARAM" value="0" />
<parameter name="PLL_WRITE_CLK_PHASE_PS_SIM_STR_PARAM" value="" />
<parameter name="PLL_WRITE_CLK_MULT_PARAM" value="0" />
<parameter name="PLL_WRITE_CLK_DIV_PARAM" value="0" />
<parameter name="PLL_ADDR_CMD_CLK_FREQ_PARAM" value="0.0" />
<parameter name="PLL_ADDR_CMD_CLK_FREQ_SIM_STR_PARAM" value="" />
<parameter name="PLL_ADDR_CMD_CLK_PHASE_PS_PARAM" value="0" />
<parameter name="PLL_ADDR_CMD_CLK_PHASE_PS_SIM_STR_PARAM" value="" />
<parameter name="PLL_ADDR_CMD_CLK_MULT_PARAM" value="0" />
<parameter name="PLL_ADDR_CMD_CLK_DIV_PARAM" value="0" />
<parameter name="PLL_AFI_HALF_CLK_FREQ_PARAM" value="0.0" />
<parameter name="PLL_AFI_HALF_CLK_FREQ_SIM_STR_PARAM" value="" />
<parameter name="PLL_AFI_HALF_CLK_PHASE_PS_PARAM" value="0" />
<parameter name="PLL_AFI_HALF_CLK_PHASE_PS_SIM_STR_PARAM" value="" />
<parameter name="PLL_AFI_HALF_CLK_MULT_PARAM" value="0" />
<parameter name="PLL_AFI_HALF_CLK_DIV_PARAM" value="0" />
<parameter name="PLL_NIOS_CLK_FREQ_PARAM" value="0.0" />
<parameter name="PLL_NIOS_CLK_FREQ_SIM_STR_PARAM" value="" />
<parameter name="PLL_NIOS_CLK_PHASE_PS_PARAM" value="0" />
<parameter name="PLL_NIOS_CLK_PHASE_PS_SIM_STR_PARAM" value="" />
<parameter name="PLL_NIOS_CLK_MULT_PARAM" value="0" />
<parameter name="PLL_NIOS_CLK_DIV_PARAM" value="0" />
<parameter name="PLL_CONFIG_CLK_FREQ_PARAM" value="0.0" />
<parameter name="PLL_CONFIG_CLK_FREQ_SIM_STR_PARAM" value="" />
<parameter name="PLL_CONFIG_CLK_PHASE_PS_PARAM" value="0" />
```

```
<parameter name="PLL_CONFIG_CLK_PHASE_PS_SIM_STR_PARAM" value="" />
<parameter name="PLL_CONFIG_CLK_MULT_PARAM" value="0" />
<parameter name="PLL_CONFIG_CLK_DIV_PARAM" value="0" />
<parameter name="PLL_P2C_READ_CLK_FREQ_PARAM" value="0.0" />
<parameter name="PLL_P2C_READ_CLK_FREQ_SIM_STR_PARAM" value="" />
<parameter name="PLL_P2C_READ_CLK_PHASE_PS_PARAM" value="0" />
<parameter name="PLL_P2C_READ_CLK_PHASE_PS_SIM_STR_PARAM" value="" />
<parameter name="PLL_P2C_READ_CLK_MULT_PARAM" value="0" />
<parameter name="PLL_P2C_READ_CLK_DIV_PARAM" value="0" />
<parameter name="PLL_C2P_WRITE_CLK_FREQ_PARAM" value="0.0" />
<parameter name="PLL_C2P_WRITE_CLK_FREQ_SIM_STR_PARAM" value="" />
<parameter name="PLL_C2P_WRITE_CLK_PHASE_PS_PARAM" value="0" />
<parameter name="PLL_C2P_WRITE_CLK_PHASE_PS_SIM_STR_PARAM" value="" />
<parameter name="PLL_C2P_WRITE_CLK_MULT_PARAM" value="0" />
<parameter name="PLL_C2P_WRITE_CLK_DIV_PARAM" value="0" />
<parameter name="PLL_HR_CLK_FREQ_PARAM" value="0.0" />
<parameter name="PLL_HR_CLK_FREQ_SIM_STR_PARAM" value="" />
<parameter name="PLL_HR_CLK_PHASE_PS_PARAM" value="0" />
<parameter name="PLL_HR_CLK_PHASE_PS_SIM_STR_PARAM" value="" />
<parameter name="PLL_HR_CLK_MULT_PARAM" value="0" />
<parameter name="PLL_HR_CLK_DIV_PARAM" value="0" />
<parameter name="PLL_AFI_PHY_CLK_FREQ_PARAM" value="0.0" />
<parameter name="PLL_AFI_PHY_CLK_FREQ_SIM_STR_PARAM" value="" />
<parameter name="PLL_AFI_PHY_CLK_PHASE_PS_PARAM" value="0" />
<parameter name="PLL_AFI_PHY_CLK_PHASE_PS_SIM_STR_PARAM" value="" />
<parameter name="PLL_AFI_PHY_CLK_MULT_PARAM" value="0" />
<parameter name="PLL_AFI_PHY_CLK_DIV_PARAM" value="0" />
<parameter name="PLL_CLK_PARAM_VALID" value="false" />
<parameter name="ENABLE_EXTRA_REPORTING" value="false" />
<parameter name="NUM_EXTRA_REPORT_PATH" value="10" />
<parameter name="ENABLE_ISS_PROBES" value="false" />
<parameter name="CALIB_REG_WIDTH" value="8" />
<parameter name="USE_SEQUENCER_BFM" value="false" />
<parameter name="DEFAULT_FAST_SIM_MODEL" value="true" />
<parameter name="PLL_SHARING_MODE" value="None" />
<parameter name="NUM_PLL_SHARING_INTERFACES" value="1" />
<parameter name="EXPORT_AFI_HALF_CLK" value="false" />
<parameter name="ABSTRACT_REAL_COMPARE_TEST" value="false" />
<parameter name="INCLUDE_BOARD_DELAY_MODEL" value="false" />
<parameter name="INCLUDE_MULTIRANK_BOARD_DELAY_MODEL" value="false" />
<parameter name="USE_FAKE_PHY" value="false" />
<parameter name="FORCE_MAX_LATENCY_COUNT_WIDTH" value="0" />
<parameter name="ENABLE_NON_DESTRUCTIVE_CALIB" value="false" />
<parameter name="TRACKING_ERROR_TEST" value="false" />
<parameter name="TRACKING_WATCH_TEST" value="false" />
<parameter name="MARGIN_VARIATION_TEST" value="false" />
<parameter name="EXTRA_SETTINGS" value="" />
<parameter name="MEM_DEVICE" value="MISSING_MODEL" />
<parameter name="FORCE_SYNTHESIS_LANGUAGE" value="" />
<parameter name="FORCED_NUM_WRITE_FR_CYCLE_SHIFTS" value="0" />
<parameter name="SEQUENCER_TYPE" value="NIOS" />
<parameter name="ADVERTIZE_SEQUENCER_SW_BUILD_FILES" value="false" />
<parameter name="FORCED_NON_LDC_ADDR_CMD_MEM_CK_INVERT" value="false" />
<parameter name="PHY_ONLY" value="false" />
<parameter name="SEQ_MODE" value="0" />
<parameter name="ADVANCED_CK_PHASES" value="false" />
<parameter name="COMMAND_PHASE" value="0.0" />
<parameter name="MEM_CK_PHASE" value="0.0" />
<parameter name="P2C_READ_CLOCK_ADD_PHASE" value="0.0" />
<parameter name="C2P_WRITE_CLOCK_ADD_PHASE" value="0.0" />
<parameter name="ACV_PHY_CLK_ADD_FR_PHASE" value="0.0" />
<parameter name="MEM_VOLTAGE" value="1.5V DDR3" />
<parameter name="PLL_LOCATION" value="Top_Bottom" />
<parameter name="SKIP_MEM_INIT" value="true" />
<parameter name="READ_DQ_DQS_CLOCK_SOURCE" value="INVERTED_DQS_BUS" />
<parameter name="DQ_INPUT_REG_USE_CLKN" value="false" />
<parameter name="DQS_DQSN_MODE" value="DIFFERENTIAL" />
<parameter name="AFI_DEBUG_INFO_WIDTH" value="32" />
```

```
1213   <parameter name="CALIBRATION_MODE" value="Skip" />
1214   <parameter name="NIOS_ROM_DATA_WIDTH" value="32" />
1215   <parameter name="READ_FIFO_SIZE" value="8" />
1216   <parameter name="PHY_CSR_ENABLED" value="false" />
1217   <parameter name="PHY_CSR_CONNECTION" value="INTERNAL_JTAG" />
1218   <parameter name="USER_DEBUG_LEVEL" value="1" />
1219   <parameter name="TIMING_BOARD_DERATE_METHOD" value="AUTO" />
1220   <parameter name="TIMING_BOARD_CK_CKN_SLEW_RATE" value="2.0" />
1221   <parameter name="TIMING_BOARD_AC_SLEW_RATE" value="1.0" />
1222   <parameter name="TIMING_BOARD_DQS_DQSN_SLEW_RATE" value="2.0" />
1223   <parameter name="TIMING_BOARD_DQ_SLEW_RATE" value="1.0" />
1224   <parameter name="TIMING_BOARD_TIS" value="0.0" />
1225   <parameter name="TIMING_BOARD_TIH" value="0.0" />
1226   <parameter name="TIMING_BOARD_TDS" value="0.0" />
1227   <parameter name="TIMING_BOARD_TDH" value="0.0" />
1228   <parameter name="TIMING_BOARD_ISI_METHOD" value="AUTO" />
1229   <parameter name="TIMING_BOARD_AC_EYE_REDUCTION_SU" value="0.0" />
1230   <parameter name="TIMING_BOARD_AC_EYE_REDUCTION_H" value="0.0" />
1231   <parameter name="TIMING_BOARD_DQ_EYE_REDUCTION" value="0.0" />
1232   <parameter name="TIMING_BOARD_DELTA_DQS_ARRIVAL_TIME" value="0.0" />
1233   <parameter name="PACKAGE_DESKEW" value="false" />
1234   <parameter name="AC_PACKAGE_DESKEW" value="false" />
1235   <parameter name="TIMING_BOARD_MAX_CK_DELAY" value="0.29132963" />
1236   <parameter name="TIMING_BOARD_MAX_DQS_DELAY" value="0.286640405" />
1237   <parameter name="TIMING_BOARD_SKEW_CKDQS_DIMM_MIN" value="-0.005828107" />
1238   <parameter name="TIMING_BOARD_SKEW_CKDQS_DIMM_MAX" value="0.048251827" />
1239   <parameter name="TIMING_BOARD_SKEW_BETWEEN_DIMMS" value="0.05" />
1240   <parameter name="TIMING_BOARD_SKEW_WITHIN_DQS" value="0.001782034" />
1241   <parameter name="TIMING_BOARD_SKEW_BETWEEN_DQS" value="0.029609881" />
1242   <parameter name="TIMING_BOARD_DQ_TO_DQS_SKEW" value="0.0" />
1243   <parameter name="TIMING_BOARD_AC_SKEW" value="0.096664741" />
1244   <parameter name="TIMING_BOARD_AC_TO_CK_SKEW" value="0.026243846" />
1245   <parameter name="ENABLE_EXPORT_SEQ_DEBUG_BRIDGE" value="false" />
1246   <parameter name="CORE_DEBUG_CONNECTION" value="EXPORT" />
1247   <parameter name="ADD_EXTERNAL_SEQ_DEBUG_NIOS" value="false" />
1248   <parameter name="ED_EXPORT_SEQ_DEBUG" value="false" />
1249   <parameter name="ADD_EFFICIENCY_MONITOR" value="false" />
1250   <parameter name="ENABLE_ABS_RAM_MEM_INIT" value="false" />
1251   <parameter name="ABS_RAM_MEM_INIT_FILENAME" value="meminit" />
1252   <parameter name="DLL_SHARING_MODE" value="None" />
1253   <parameter name="NUM_DLL_SHARING_INTERFACES" value="1" />
1254   <parameter name="OCT_SHARING_MODE" value="None" />
1255   <parameter name="NUM_OCT_SHARING_INTERFACES" value="1" />
1256   <parameter name="AUTO_DEVICE" value="5CSXFC6D6F31C8ES" />
1257  </module>
1258  <module kind="clock_source" version="13.0" enabled="1" name="refclk_100">
1259   <parameter name="clockFrequency" value="100000000" />
1260   <parameter name="clockFrequencyKnown" value="true" />
1261   <parameter name="inputClockFrequency" value="0" />
1262   <parameter name="resetSynchronousEdges" value="NONE" />
1263  </module>
1264  <module kind="altera_pll" version="13.0" enabled="1" name="pll_0">
1265   <parameter name="device_family" value="Cyclone V" />
1266   <parameter name="gui_device_speed_grade" value="7" />
1267   <parameter name="gui_pll_mode" value="Integer-N PLL" />
1268   <parameter name="gui_reference_clock_frequency" value="400.0" />
1269   <parameter name="gui_channel_spacing" value="0.0" />
1270   <parameter name="gui_operation_mode" value="normal" />
1271   <parameter name="gui_feedback_clock" value="Global Clock" />
1272   <parameter name="gui_fractional_cout" value="32" />
1273   <parameter name="gui_dsm_out_sel" value="1st_order" />
1274   <parameter name="gui_use_locked" value="false" />
1275   <parameter name="gui_en_adv_params" value="false" />
1276   <parameter name="gui_number_of_clocks" value="1" />
1277   <parameter name="gui_multiply_factor" value="1" />
1278   <parameter name="gui_frac_multiply_factor" value="1" />
1279   <parameter name="gui_divide_factor_n" value="1" />
1280   <parameter name="gui_output_clock_frequency0" value="100.0" />
```

```
<parameter name="gui_divide_factor_c0" value="1" />
<parameter name="gui_actual_output_clock_frequency0" value="0 MHz" />
<parameter name="gui_ps_units0" value="ps" />
<parameter name="gui_phase_shift0" value="0" />
<parameter name="gui_phase_shift_deg0" value="0" />
<parameter name="gui_actual_phase_shift0" value="0" />
<parameter name="gui_duty_cycle0" value="50" />
<parameter name="gui_output_clock_frequency1" value="100.0" />
<parameter name="gui_divide_factor_c1" value="1" />
<parameter name="gui_actual_output_clock_frequency1" value="0 MHz" />
<parameter name="gui_ps_units1" value="ps" />
<parameter name="gui_phase_shift1" value="0" />
<parameter name="gui_phase_shift_deg1" value="0" />
<parameter name="gui_actual_phase_shift1" value="0" />
<parameter name="gui_duty_cycle1" value="50" />
<parameter name="gui_output_clock_frequency2" value="100.0" />
<parameter name="gui_divide_factor_c2" value="1" />
<parameter name="gui_actual_output_clock_frequency2" value="0 MHz" />
<parameter name="gui_ps_units2" value="ps" />
<parameter name="gui_phase_shift2" value="0" />
<parameter name="gui_phase_shift_deg2" value="0" />
<parameter name="gui_actual_phase_shift2" value="0" />
<parameter name="gui_duty_cycle2" value="50" />
<parameter name="gui_output_clock_frequency3" value="100.0" />
<parameter name="gui_divide_factor_c3" value="1" />
<parameter name="gui_actual_output_clock_frequency3" value="0 MHz" />
<parameter name="gui_ps_units3" value="ps" />
<parameter name="gui_phase_shift3" value="0" />
<parameter name="gui_phase_shift_deg3" value="0" />
<parameter name="gui_actual_phase_shift3" value="0" />
<parameter name="gui_duty_cycle3" value="50" />
<parameter name="gui_output_clock_frequency4" value="100.0" />
<parameter name="gui_divide_factor_c4" value="1" />
<parameter name="gui_actual_output_clock_frequency4" value="0 MHz" />
<parameter name="gui_ps_units4" value="ps" />
<parameter name="gui_phase_shift4" value="0" />
<parameter name="gui_phase_shift_deg4" value="0" />
<parameter name="gui_actual_phase_shift4" value="0" />
<parameter name="gui_duty_cycle4" value="50" />
<parameter name="gui_output_clock_frequency5" value="100.0" />
<parameter name="gui_divide_factor_c5" value="1" />
<parameter name="gui_actual_output_clock_frequency5" value="0 MHz" />
<parameter name="gui_ps_units5" value="ps" />
<parameter name="gui_phase_shift5" value="0" />
<parameter name="gui_phase_shift_deg5" value="0" />
<parameter name="gui_actual_phase_shift5" value="0" />
<parameter name="gui_duty_cycle5" value="50" />
<parameter name="gui_output_clock_frequency6" value="100.0" />
<parameter name="gui_divide_factor_c6" value="1" />
<parameter name="gui_actual_output_clock_frequency6" value="0 MHz" />
<parameter name="gui_ps_units6" value="ps" />
<parameter name="gui_phase_shift6" value="0" />
<parameter name="gui_phase_shift_deg6" value="0" />
<parameter name="gui_actual_phase_shift6" value="0" />
<parameter name="gui_duty_cycle6" value="50" />
<parameter name="gui_output_clock_frequency7" value="100.0" />
<parameter name="gui_divide_factor_c7" value="1" />
<parameter name="gui_actual_output_clock_frequency7" value="0 MHz" />
<parameter name="gui_ps_units7" value="ps" />
<parameter name="gui_phase_shift7" value="0" />
<parameter name="gui_phase_shift_deg7" value="0" />
<parameter name="gui_actual_phase_shift7" value="0" />
<parameter name="gui_duty_cycle7" value="50" />
<parameter name="gui_output_clock_frequency8" value="100.0" />
<parameter name="gui_divide_factor_c8" value="1" />
<parameter name="gui_actual_output_clock_frequency8" value="0 MHz" />
<parameter name="gui_ps_units8" value="ps" />
<parameter name="gui_phase_shift8" value="0" />
```

```
<parameter name="gui_phase_shift_deg8" value="0" />
<parameter name="gui_actual_phase_shift8" value="0" />
<parameter name="gui_duty_cycle8" value="50" />
<parameter name="gui_output_clock_frequency9" value="100.0" />
<parameter name="gui_divide_factor_c9" value="1" />
<parameter name="gui_actual_output_clock_frequency9" value="0 MHz" />
<parameter name="gui_ps_units9" value="ps" />
<parameter name="gui_phase_shift9" value="0" />
<parameter name="gui_phase_shift_deg9" value="0" />
<parameter name="gui_actual_phase_shift9" value="0" />
<parameter name="gui_duty_cycle9" value="50" />
<parameter name="gui_output_clock_frequency10" value="100.0" />
<parameter name="gui_divide_factor_c10" value="1" />
<parameter name="gui_actual_output_clock_frequency10" value="0 MHz" />
<parameter name="gui_ps_units10" value="ps" />
<parameter name="gui_phase_shift10" value="0" />
<parameter name="gui_phase_shift_deg10" value="0" />
<parameter name="gui_actual_phase_shift10" value="0" />
<parameter name="gui_duty_cycle10" value="50" />
<parameter name="gui_output_clock_frequency11" value="100.0" />
<parameter name="gui_divide_factor_c11" value="1" />
<parameter name="gui_actual_output_clock_frequency11" value="0 MHz" />
<parameter name="gui_ps_units11" value="ps" />
<parameter name="gui_phase_shift11" value="0" />
<parameter name="gui_phase_shift_deg11" value="0" />
<parameter name="gui_actual_phase_shift11" value="0" />
<parameter name="gui_duty_cycle11" value="50" />
<parameter name="gui_output_clock_frequency12" value="100.0" />
<parameter name="gui_divide_factor_c12" value="1" />
<parameter name="gui_actual_output_clock_frequency12" value="0 MHz" />
<parameter name="gui_ps_units12" value="ps" />
<parameter name="gui_phase_shift12" value="0" />
<parameter name="gui_phase_shift_deg12" value="0" />
<parameter name="gui_actual_phase_shift12" value="0" />
<parameter name="gui_duty_cycle12" value="50" />
<parameter name="gui_output_clock_frequency13" value="100.0" />
<parameter name="gui_divide_factor_c13" value="1" />
<parameter name="gui_actual_output_clock_frequency13" value="0 MHz" />
<parameter name="gui_ps_units13" value="ps" />
<parameter name="gui_phase_shift13" value="0" />
<parameter name="gui_phase_shift_deg13" value="0" />
<parameter name="gui_actual_phase_shift13" value="0" />
<parameter name="gui_duty_cycle13" value="50" />
<parameter name="gui_output_clock_frequency14" value="100.0" />
<parameter name="gui_divide_factor_c14" value="1" />
<parameter name="gui_actual_output_clock_frequency14" value="0 MHz" />
<parameter name="gui_ps_units14" value="ps" />
<parameter name="gui_phase_shift14" value="0" />
<parameter name="gui_phase_shift_deg14" value="0" />
<parameter name="gui_actual_phase_shift14" value="0" />
<parameter name="gui_duty_cycle14" value="50" />
<parameter name="gui_output_clock_frequency15" value="100.0" />
<parameter name="gui_divide_factor_c15" value="1" />
<parameter name="gui_actual_output_clock_frequency15" value="0 MHz" />
<parameter name="gui_ps_units15" value="ps" />
<parameter name="gui_phase_shift15" value="0" />
<parameter name="gui_phase_shift_deg15" value="0" />
<parameter name="gui_actual_phase_shift15" value="0" />
<parameter name="gui_duty_cycle15" value="50" />
<parameter name="gui_output_clock_frequency16" value="100.0" />
<parameter name="gui_divide_factor_c16" value="1" />
<parameter name="gui_actual_output_clock_frequency16" value="0 MHz" />
<parameter name="gui_ps_units16" value="ps" />
<parameter name="gui_phase_shift16" value="0" />
<parameter name="gui_phase_shift_deg16" value="0" />
<parameter name="gui_actual_phase_shift16" value="0" />
<parameter name="gui_duty_cycle16" value="50" />
<parameter name="gui_output_clock_frequency17" value="100.0" />
```

```
 <parameter name="gui_divide_factor_c17" value="1" />
 <parameter name="gui_actual_output_clock_frequency17" value="0 MHz" />
 <parameter name="gui_ps_units17" value="ps" />
 <parameter name="gui_phase_shift17" value="0" />
 <parameter name="gui_phase_shift_deg17" value="0" />
 <parameter name="gui_actual_phase_shift17" value="0" />
 <parameter name="gui_duty_cycle17" value="50" />
 <parameter name="gui_pll_auto_reset" value="Off" />
 <parameter name="gui_pll_bandwidth_preset" value="Auto" />
 <parameter name="gui_en_reconf" value="false" />
 <parameter name="gui_en_dps_ports" value="false" />
 <parameter name="gui_en_phout_ports" value="false" />
 <parameter name="gui_mif_generate" value="false" />
 <parameter name="gui_enable_mif_dps" value="false" />
 <parameter name="gui_dps_cntr" value="C0" />
 <parameter name="gui_dps_num" value="1" />
 <parameter name="gui_dps_dir" value="Positive" />
 <parameter name="gui_refclk_switch" value="false" />
 <parameter name="gui_refclk1_frequency" value="100.0" />
 <parameter name="gui_switchover_mode">Automatic Switchover</parameter>
 <parameter name="gui_switchover_delay" value="0" />
 <parameter name="gui_active_clk" value="false" />
 <parameter name="gui_clk_bad" value="false" />
 <parameter name="gui_enable_cascade_out" value="false" />
 <parameter name="gui_enable_cascade_in" value="false" />
 <parameter name="gui_pll_cascading_mode">Create an adjpllin signal to connect with an
     upstream PLL</parameter>
 <parameter name="AUTO_REFCLK_CLOCK_RATE" value="400000000" />
</module>
<module
  kind="altera_avalon_mm_clock_crossing_bridge"
  version="13.0"
  enabled="1"
  name="mm_clock_crossing_bridge_0">
 <parameter name="DATA_WIDTH" value="32" />
 <parameter name="SYMBOL_WIDTH" value="8" />
 <parameter name="ADDRESS_WIDTH" value="28" />
 <parameter name="ADDRESS_UNITS" value="WORDS" />
 <parameter name="MAX_BURST_SIZE" value="1" />
 <parameter name="COMMAND_FIFO_DEPTH" value="4" />
 <parameter name="RESPONSE_FIFO_DEPTH" value="4" />
 <parameter name="MASTER_SYNC_DEPTH" value="2" />
 <parameter name="SLAVE_SYNC_DEPTH" value="2" />
 <parameter name="AUTO_M0_CLK_CLOCK_RATE" value="100000000" />
 <parameter name="AUTO_S0_CLK_CLOCK_RATE" value="50000000" />
 <parameter name="AUTO_DEVICE_FAMILY" value="Cyclone V" />
</module>
<module kind="ImgDiff" version="1.0" enabled="0" name="ImgDiff_0">
 <parameter name="DATA_WIDTH" value="32" />
 <parameter name="ADDRESS_WIDTH" value="32" />
 <parameter name="MASTER_WRITE_BURST_CAPABLE" value="0" />
 <parameter name="MASTER_WRITE_MAXIMUM_BURST_COUNT" value="2" />
 <parameter name="MASTER_WRITE_BURST_COUNT_WIDTH" value="2" />
 <parameter name="MASTER_WRITE_FIFO_DEPTH" value="32" />
 <parameter name="MASTER_WRITE_FIFO_DEPTH_LOG2" value="5" />
 <parameter name="MASTER_WRITE_MEMORY_BASED_FIFO" value="1" />
 <parameter name="MASTER_READ_BURST_CAPABLE" value="0" />
 <parameter name="MASTER_READ_MAXIMUM_BURST_COUNT" value="2" />
 <parameter name="MASTER_READ_BURST_COUNT_WIDTH" value="2" />
 <parameter name="MASTER_READ_FIFO_DEPTH" value="32" />
 <parameter name="MASTER_READ_FIFO_DEPTH_LOG2" value="5" />
 <parameter name="MASTER_READ_MEMORY_BASED_FIFO" value="1" />
 <parameter name="IRQ_EN" value="0" />
 <parameter name="AUTO_CLK_IN_CLOCK_RATE" value="100000000" />
 <parameter name="AUTO_DEVICE_FAMILY" value="Cyclone V" />
</module>
<module
  kind="altera_avalon_mm_clock_crossing_bridge"
```

```
    version="13.0"
    enabled="1"
    name="mm_clock_crossing_bridge_1">
   <parameter name="DATA_WIDTH" value="32" />
   <parameter name="SYMBOL_WIDTH" value="8" />
   <parameter name="ADDRESS_WIDTH" value="10" />
   <parameter name="ADDRESS_UNITS" value="WORDS" />
   <parameter name="MAX_BURST_SIZE" value="1" />
   <parameter name="COMMAND_FIFO_DEPTH" value="4" />
   <parameter name="RESPONSE_FIFO_DEPTH" value="4" />
   <parameter name="MASTER_SYNC_DEPTH" value="2" />
   <parameter name="SLAVE_SYNC_DEPTH" value="2" />
   <parameter name="AUTO_M0_CLK_CLOCK_RATE" value="50000000" />
   <parameter name="AUTO_S0_CLK_CLOCK_RATE" value="100000000" />
   <parameter name="AUTO_DEVICE_FAMILY" value="Cyclone V" />
  </module>
  <module kind="IPTest" version="1.0" enabled="1" name="IPTest_0">
   <parameter name="DATA_WIDTH" value="32" />
   <parameter name="ADDRESS_WIDTH" value="32" />
   <parameter name="MASTER_WRITE_BURST_CAPABLE" value="0" />
   <parameter name="MASTER_WRITE_MAXIMUM_BURST_COUNT" value="2" />
   <parameter name="MASTER_WRITE_BURST_COUNT_WIDTH" value="2" />
   <parameter name="MASTER_WRITE_FIFO_DEPTH" value="32" />
   <parameter name="MASTER_WRITE_FIFO_DEPTH_LOG2" value="5" />
   <parameter name="MASTER_WRITE_MEMORY_BASED_FIFO" value="1" />
   <parameter name="MASTER_READ_BURST_CAPABLE" value="0" />
   <parameter name="MASTER_READ_MAXIMUM_BURST_COUNT" value="2" />
   <parameter name="MASTER_READ_BURST_COUNT_WIDTH" value="2" />
   <parameter name="MASTER_READ_FIFO_DEPTH" value="32" />
   <parameter name="MASTER_READ_FIFO_DEPTH_LOG2" value="5" />
   <parameter name="MASTER_READ_MEMORY_BASED_FIFO" value="1" />
   <parameter name="IRQ_EN" value="0" />
   <parameter name="AUTO_CLK_IN_CLOCK_RATE" value="100000000" />
   <parameter name="AUTO_DEVICE_FAMILY" value="Cyclone V" />
  </module>
  <connection
     kind="avalon"
     version="13.0"
     start="hps_0.h2f_lw_axi_master"
     end="sysid_qsys.control_slave">
   <parameter name="arbitrationPriority" value="1" />
   <parameter name="baseAddress" value="0x00010000" />
   <parameter name="defaultConnection" value="false" />
  </connection>
  <connection
     kind="avalon"
     version="13.0"
     start="hps_0.h2f_lw_axi_master"
     end="led_pio.s1">
   <parameter name="arbitrationPriority" value="1" />
   <parameter name="baseAddress" value="0x00010040" />
   <parameter name="defaultConnection" value="false" />
  </connection>
  <connection
     kind="avalon"
     version="13.0"
     start="hps_0.h2f_lw_axi_master"
     end="dipsw_pio.s1">
   <parameter name="arbitrationPriority" value="1" />
   <parameter name="baseAddress" value="0x00010080" />
   <parameter name="defaultConnection" value="false" />
  </connection>
  <connection
     kind="avalon"
     version="13.0"
     start="hps_0.h2f_lw_axi_master"
     end="button_pio.s1">
   <parameter name="arbitrationPriority" value="1" />
```

```
 <parameter name="baseAddress" value="0x000100c0" />
 <parameter name="defaultConnection" value="false" />
</connection>
<connection
   kind="interrupt"
   version="13.0"
   start="intr_capturer_0.interrupt_receiver"
   end="button_pio.irq">
 <parameter name="irqNumber" value="1" />
</connection>
<connection
   kind="interrupt"
   version="13.0"
   start="intr_capturer_0.interrupt_receiver"
   end="dipsw_pio.irq">
 <parameter name="irqNumber" value="0" />
</connection>
<connection
   kind="interrupt"
   version="13.0"
   start="intr_capturer_0.interrupt_receiver"
   end="jtag_uart.irq">
 <parameter name="irqNumber" value="2" />
</connection>
<connection
   kind="interrupt"
   version="13.0"
   start="hps_0.f2h_irq0"
   end="dipsw_pio.irq">
 <parameter name="irqNumber" value="0" />
</connection>
<connection
   kind="interrupt"
   version="13.0"
   start="hps_0.f2h_irq0"
   end="button_pio.irq">
 <parameter name="irqNumber" value="1" />
</connection>
<connection
   kind="interrupt"
   version="13.0"
   start="hps_0.f2h_irq0"
   end="jtag_uart.irq">
 <parameter name="irqNumber" value="2" />
</connection>
<connection
   kind="avalon"
   version="13.0"
   start="hps_0.h2f_lw_axi_master"
   end="jtag_uart.avalon_jtag_slave">
 <parameter name="arbitrationPriority" value="1" />
 <parameter name="baseAddress" value="0x00020000" />
 <parameter name="defaultConnection" value="false" />
</connection>
<connection
   kind="clock"
   version="13.0"
   start="ext_clk_50.clk"
   end="intr_capturer_0.clock" />
<connection
   kind="clock"
   version="13.0"
   start="ext_clk_50.clk"
   end="fpga_only_master.clk" />
<connection
   kind="clock"
   version="13.0"
   start="ext_clk_50.clk"
```

```
1620     end="jtag_uart.clk" />
1621   <connection
1622     kind="clock"
1623     version="13.0"
1624     start="ext_clk_50.clk"
1625     end="button_pio.clk" />
1626   <connection
1627     kind="clock"
1628     version="13.0"
1629     start="ext_clk_50.clk"
1630     end="dipsw_pio.clk" />
1631   <connection kind="clock" version="13.0" start="ext_clk_50.clk" end="led_pio.clk" />
1632   <connection
1633     kind="clock"
1634     version="13.0"
1635     start="ext_clk_50.clk"
1636     end="sysid_qsys.clk" />
1637   <connection
1638     kind="clock"
1639     version="13.0"
1640     start="ext_clk_50.clk"
1641     end="hps_0.h2f_lw_axi_clock" />
1642   <connection
1643     kind="clock"
1644     version="13.0"
1645     start="ext_clk_50.clk"
1646     end="hps_0.f2h_axi_clock" />
1647   <connection
1648     kind="avalon"
1649     version="13.0"
1650     start="fpga_only_master.master"
1651     end="jtag_uart.avalon_jtag_slave">
1652    <parameter name="arbitrationPriority" value="1" />
1653    <parameter name="baseAddress" value="0x00020000" />
1654    <parameter name="defaultConnection" value="false" />
1655   </connection>
1656   <connection
1657     kind="avalon"
1658     version="13.0"
1659     start="fpga_only_master.master"
1660     end="button_pio.s1">
1661    <parameter name="arbitrationPriority" value="1" />
1662    <parameter name="baseAddress" value="0x000100c0" />
1663    <parameter name="defaultConnection" value="false" />
1664   </connection>
1665   <connection
1666     kind="avalon"
1667     version="13.0"
1668     start="fpga_only_master.master"
1669     end="dipsw_pio.s1">
1670    <parameter name="arbitrationPriority" value="1" />
1671    <parameter name="baseAddress" value="0x00010080" />
1672    <parameter name="defaultConnection" value="false" />
1673   </connection>
1674   <connection
1675     kind="avalon"
1676     version="13.0"
1677     start="fpga_only_master.master"
1678     end="led_pio.s1">
1679    <parameter name="arbitrationPriority" value="1" />
1680    <parameter name="baseAddress" value="0x00010040" />
1681    <parameter name="defaultConnection" value="false" />
1682   </connection>
1683   <connection
1684     kind="avalon"
1685     version="13.0"
1686     start="fpga_only_master.master"
1687     end="sysid_qsys.control_slave">
```

```
 <parameter name="arbitrationPriority" value="1" />
 <parameter name="baseAddress" value="0x00010000" />
 <parameter name="defaultConnection" value="false" />
</connection>
<connection
   kind="avalon"
   version="13.0"
   start="fpga_only_master.master"
   end="intr_capturer_0.avalon_slave_0">
 <parameter name="arbitrationPriority" value="1" />
 <parameter name="baseAddress" value="0x00030000" />
 <parameter name="defaultConnection" value="false" />
</connection>
<connection
   kind="clock"
   version="13.0"
   start="ext_clk_50.clk"
   end="hps_only_master.clk" />
<connection
   kind="avalon"
   version="13.0"
   start="hps_only_master.master"
   end="hps_0.f2h_axi_slave">
 <parameter name="arbitrationPriority" value="1" />
 <parameter name="baseAddress" value="0x0000" />
 <parameter name="defaultConnection" value="false" />
</connection>
<connection
   kind="clock"
   version="13.0"
   start="ext_clk_50.clk"
   end="hps_0.h2f_axi_clock" />
<connection
   kind="reset"
   version="13.0"
   start="ext_clk_50.clk_reset"
   end="intr_capturer_0.reset_sink" />
<connection
   kind="reset"
   version="13.0"
   start="ext_clk_50.clk_reset"
   end="fpga_only_master.clk_reset" />
<connection
   kind="reset"
   version="13.0"
   start="ext_clk_50.clk_reset"
   end="jtag_uart.reset" />
<connection
   kind="reset"
   version="13.0"
   start="ext_clk_50.clk_reset"
   end="button_pio.reset" />
<connection
   kind="reset"
   version="13.0"
   start="ext_clk_50.clk_reset"
   end="dipsw_pio.reset" />
<connection
   kind="reset"
   version="13.0"
   start="ext_clk_50.clk_reset"
   end="led_pio.reset" />
<connection
   kind="reset"
   version="13.0"
   start="ext_clk_50.clk_reset"
   end="sysid_qsys.reset" />
<connection
```

```
1756      kind="reset"
1757      version="13.0"
1758      start="ext_clk_50.clk_reset"
1759      end="hps_only_master.clk_reset" />
1760    <connection
1761      kind="clock"
1762      version="13.0"
1763      start="refclk_100.clk"
1764      end="mem_if_ddr3_fpga.pll_ref_clk" />
1765    <connection
1766      kind="clock"
1767      version="13.0"
1768      start="mem_if_ddr3_fpga.afi_clk"
1769      end="pll_0.refclk" />
1770    <connection
1771      kind="reset"
1772      version="13.0"
1773      start="ext_clk_50.clk_reset"
1774      end="pll_0.reset" />
1775    <connection
1776      kind="reset"
1777      version="13.0"
1778      start="ext_clk_50.clk_reset"
1779      end="mem_if_ddr3_fpga.mp_cmd_reset_n_0" />
1780    <connection
1781      kind="reset"
1782      version="13.0"
1783      start="ext_clk_50.clk_reset"
1784      end="mem_if_ddr3_fpga.mp_rfifo_reset_n_0" />
1785    <connection
1786      kind="reset"
1787      version="13.0"
1788      start="ext_clk_50.clk_reset"
1789      end="mem_if_ddr3_fpga.mp_wfifo_reset_n_0" />
1790    <connection
1791      kind="reset"
1792      version="13.0"
1793      start="ext_clk_50.clk_reset"
1794      end="mem_if_ddr3_fpga.mp_rfifo_reset_n_1" />
1795    <connection
1796      kind="reset"
1797      version="13.0"
1798      start="ext_clk_50.clk_reset"
1799      end="mem_if_ddr3_fpga.mp_wfifo_reset_n_1" />
1800    <connection
1801      kind="reset"
1802      version="13.0"
1803      start="ext_clk_50.clk_reset"
1804      end="mem_if_ddr3_fpga.mp_rfifo_reset_n_2" />
1805    <connection
1806      kind="reset"
1807      version="13.0"
1808      start="ext_clk_50.clk_reset"
1809      end="mem_if_ddr3_fpga.mp_wfifo_reset_n_2" />
1810    <connection
1811      kind="reset"
1812      version="13.0"
1813      start="ext_clk_50.clk_reset"
1814      end="mem_if_ddr3_fpga.mp_rfifo_reset_n_3" />
1815    <connection
1816      kind="reset"
1817      version="13.0"
1818      start="ext_clk_50.clk_reset"
1819      end="mem_if_ddr3_fpga.mp_wfifo_reset_n_3" />
1820    <connection
1821      kind="clock"
1822      version="13.0"
1823      start="pll_0.outclk0"
```

```
    end="mem_if_ddr3_fpga.mp_cmd_clk_0" />
  <connection
    kind="clock"
    version="13.0"
    start="pll_0.outclk0"
    end="mem_if_ddr3_fpga.mp_rfifo_clk_0" />
  <connection
    kind="clock"
    version="13.0"
    start="pll_0.outclk0"
    end="mem_if_ddr3_fpga.mp_wfifo_clk_0" />
  <connection
    kind="clock"
    version="13.0"
    start="pll_0.outclk0"
    end="mem_if_ddr3_fpga.mp_rfifo_clk_1" />
  <connection
    kind="clock"
    version="13.0"
    start="pll_0.outclk0"
    end="mem_if_ddr3_fpga.mp_rfifo_clk_2" />
  <connection
    kind="clock"
    version="13.0"
    start="pll_0.outclk0"
    end="mem_if_ddr3_fpga.mp_wfifo_clk_1" />
  <connection
    kind="clock"
    version="13.0"
    start="pll_0.outclk0"
    end="mem_if_ddr3_fpga.mp_wfifo_clk_2" />
  <connection
    kind="clock"
    version="13.0"
    start="pll_0.outclk0"
    end="mem_if_ddr3_fpga.mp_rfifo_clk_3" />
  <connection
    kind="clock"
    version="13.0"
    start="pll_0.outclk0"
    end="mem_if_ddr3_fpga.mp_wfifo_clk_3" />
  <connection
    kind="reset"
    version="13.0"
    start="ext_clk_50.clk_reset"
    end="refclk_100.clk_in_reset" />
  <connection
    kind="avalon"
    version="13.0"
    start="hps_0.h2f_axi_master"
    end="mm_clock_crossing_bridge_0.s0">
   <parameter name="arbitrationPriority" value="1" />
   <parameter name="baseAddress" value="0x0000" />
   <parameter name="defaultConnection" value="false" />
  </connection>
  <connection
    kind="avalon"
    version="13.0"
    start="mm_clock_crossing_bridge_0.m0"
    end="mem_if_ddr3_fpga.avl_0">
   <parameter name="arbitrationPriority" value="1" />
   <parameter name="baseAddress" value="0x0000" />
   <parameter name="defaultConnection" value="false" />
  </connection>
  <connection
    kind="clock"
    version="13.0"
    start="pll_0.outclk0"
```

```
    end="mm_clock_crossing_bridge_0.m0_clk" />
  <connection
    kind="clock"
    version="13.0"
    start="ext_clk_50.clk"
    end="mm_clock_crossing_bridge_0.s0_clk" />
  <connection
    kind="reset"
    version="13.0"
    start="ext_clk_50.clk_reset"
    end="mm_clock_crossing_bridge_0.s0_reset" />
  <connection
    kind="reset"
    version="13.0"
    start="refclk_100.clk_reset"
    end="mem_if_ddr3_fpga.soft_reset" />
  <connection
    kind="reset"
    version="13.0"
    start="refclk_100.clk_reset"
    end="mem_if_ddr3_fpga.global_reset" />
  <connection
    kind="reset"
    version="13.0"
    start="refclk_100.clk_reset"
    end="mm_clock_crossing_bridge_0.m0_reset" />
  <connection
    kind="reset"
    version="13.0"
    start="ext_clk_50.clk_reset"
    end="ImgDiff_0.clk_in_reset" />
  <connection
    kind="clock"
    version="13.0"
    start="pll_0.outclk0"
    end="ImgDiff_0.clk_in" />
  <connection
    kind="avalon"
    version="13.0"
    start="ImgDiff_0.master_read"
    end="mem_if_ddr3_fpga.avl_0">
   <parameter name="arbitrationPriority" value="1" />
   <parameter name="baseAddress" value="0x0000" />
   <parameter name="defaultConnection" value="false" />
  </connection>
  <connection
    kind="avalon"
    version="13.0"
    start="ImgDiff_0.master_write"
    end="mem_if_ddr3_fpga.avl_0">
   <parameter name="arbitrationPriority" value="1" />
   <parameter name="baseAddress" value="0x0000" />
   <parameter name="defaultConnection" value="false" />
  </connection>
  <connection
    kind="avalon"
    version="13.0"
    start="hps_0.h2f_lw_axi_master"
    end="mm_clock_crossing_bridge_1.s0">
   <parameter name="arbitrationPriority" value="1" />
   <parameter name="baseAddress" value="0x00030000" />
   <parameter name="defaultConnection" value="false" />
  </connection>
  <connection
    kind="avalon"
    version="13.0"
    start="mm_clock_crossing_bridge_1.m0"
    end="ImgDiff_0.csr">
```

```
 <parameter name="arbitrationPriority" value="1" />
 <parameter name="baseAddress" value="0x0000" />
 <parameter name="defaultConnection" value="false" />
</connection>
<connection
   kind="reset"
   version="13.0"
   start="ext_clk_50.clk_reset"
   end="mm_clock_crossing_bridge_1.s0_reset" />
<connection
   kind="clock"
   version="13.0"
   start="pll_0.outclk0"
   end="mm_clock_crossing_bridge_1.s0_clk" />
<connection
   kind="clock"
   version="13.0"
   start="ext_clk_50.clk"
   end="mm_clock_crossing_bridge_1.m0_clk" />
<connection
   kind="reset"
   version="13.0"
   start="ext_clk_50.clk_reset"
   end="mm_clock_crossing_bridge_1.m0_reset" />
<connection
   kind="avalon"
   version="13.0"
   start="mm_clock_crossing_bridge_1.m0"
   end="IPTest_0.csr">
 <parameter name="arbitrationPriority" value="1" />
 <parameter name="baseAddress" value="0x0800" />
 <parameter name="defaultConnection" value="false" />
</connection>
<connection
   kind="avalon"
   version="13.0"
   start="IPTest_0.master_write"
   end="mem_if_ddr3_fpga.avl_0">
 <parameter name="arbitrationPriority" value="1" />
 <parameter name="baseAddress" value="0x0000" />
 <parameter name="defaultConnection" value="false" />
</connection>
<connection
   kind="avalon"
   version="13.0"
   start="IPTest_0.master_read"
   end="mem_if_ddr3_fpga.avl_0">
 <parameter name="arbitrationPriority" value="1" />
 <parameter name="baseAddress" value="0x0000" />
 <parameter name="defaultConnection" value="false" />
</connection>
<connection
   kind="clock"
   version="13.0"
   start="pll_0.outclk0"
   end="IPTest_0.clk_in" />
<connection
   kind="reset"
   version="13.0"
   start="ext_clk_50.clk_reset"
   end="IPTest_0.clk_in_reset" />
<interconnectRequirement for="$system" name="qsys_mm.clockCrossingAdapter" value="FIFO" />
<interconnectRequirement for="$system" name="qsys_mm.maxAdditionalLatency" value="4" />
<interconnectRequirement for="hps_only_master.master" name="qsys_mm.security" value="SECURE" />
</system>
```

Listing 16.2: Qsys System - Code View (orion_system.qsys)

16.3.4 Top Level Verilog (orion_top.v)

```
//-----------------------------------------------------------------------------//
//  Title:        orion_top.v                                                  //
//  Rev:          Rev 1                                                        //
//-----------------------------------------------------------------------------//
//  Description: Senior Design Top Level File (based on golden_top.v  )        //
//-----------------------------------------------------------------------------//
//*************************ORIGIONAL COMMENTS*******************************//
//-----------------------------------------------------------------------------//
//  Title:        golden_top.v                                                 //
//  Rev:          Rev 5                                                        //
//  Created:      November 30, 2012                                            //
//-----------------------------------------------------------------------------//
//  Description: Cyclone V SX SoC pinout and IO Standard example design        //
//-----------------------------------------------------------------------------//
//  Revision History:                                                          //
//  Rev 0:        1st cut                                                      //
//-----------------------------------------------------------------------------
//------ 1 ------- 2 ------- 3 ------- 4 ------- 5 ------- 6 ------- 7 ------7
//------ 0 ------- 0 ------- 0 ------- 0 ------- 0 ------- 0 ------- 0 ------8
//-----------------------------------------------------------------------------
//Copyright 2012 Altera Corporation. All rights reserved.  Altera products
//are protected under numerous U.S. and foreign patents, maskwork rights,
//copyrights and other intellectual property laws.
//
//This reference design file, and your use thereof, is subject to and
//governed by the terms and conditions of the applicable Altera Reference
//Design License Agreement.  By using this reference design file, you
//indicate your acceptance of such terms and conditions between you and
//Altera Corporation.  In the event that you do not agree with such terms and
//conditions, you may not use the reference design file. Please promptly
//destroy any copies you have made.
//
//This reference design file being provided on an "as-is" basis and as an
//accommodation and therefore all warranties, representations or guarantees
//of any kind (whether express, implied or statutory) including, without
//limitation, warranties of merchantability, non-infringement, or fitness for
//a particular purpose, are specifically disclaimed.  By making this
//reference design file available, Altera expressly does not recommend,
//suggest or require that this reference design file be used in combination
//with any other product not provided by Altera.
//



`define    HPS
//`define    HPS_GPIO
`define     DDR3_FPGA
`define     DDR3_HPS
`define    HPS_USB
//`define I2C_FPGA
//`define MAX_FPGA

//`define    SDI_FPGA
`define    ENET_FPGA
//`define    PCIE_FPGA

//`define HSMA_XCVR
//`define HSMA_LVDS
//`define HSMA_PARALLEL
//`define HSMA_CMOS
```

```
61 //`define PCIE_XCVR
62 //`define SDI_XCVR
63
64 module orion_top (
65 //-------------------------------------------------------------------------//
66 /*
67 //User-IO-------------------------------//X pins  //-------------------------
68   output [3:0]    user_led_fpga,
69
70 //GPLL-CLK-----------------------------//X pins
71    input           clk_100m_fpga,       //2.5V    //100 MHz (2nd copy to max)
72    input           clk_50m_fpga,        //2.5V    //50MHz (2nd copy to max)
73
74   input           clk_top1,            //2.5V    //156.25 MHz adjustable
75    input           clk_bot1,            //1.5V    //100 MHz ajustable
76
77     output          ddr3_fpga_csn,       //SSTL15  //Chip Select
78   input           cpu_resetn          //2.5V    //NIOS CPU Reset Pushbutton
79 */
80 //-------------------------------------------------------------------------//
81 ////////////////// Hard Processor System (HPS) pins ///////////////////////
82 //-------------------------------------------------------------------------//
83
84 `ifdef  HPS
85 //HPS-CLK-INPUT-------------------------//2 pins   //-------------------------
86    //input           clk_osc1,            //3.3V     //Not for instantiation
87    //input           clk_osc2,            //3.3V     //Not for instantiation
88
89 //HPS-Reset----------------------------//2 pins  //-------------------------
90    //inout           mictor_rstn,         //3.3V    //Cold Reset
91    //input           hps_resetn,          //3.3V    //Warm Reset
92
93 //HPS-UART-----------------------------//2 pins   //-------------------------
94    input           uart_rx,            //3.3V LV  //UART Receive
95    output          uart_tx,            //3.3V     //UART Transmit
96
97
98 //HPS-I2C------------------------------//2 pins  //-------------------------
99   inout           i2c_scl_hps,         //3.3V    //HPS I2C Clock output
100  inout           i2c_sda_hps,         //3.3V    //HPS I2C Data Input/Output
101
102 //HPS-SPI-Bus--------------------------//4 pins  //-------------------------
103  output          spi_csn,             //3.3V    //Slave Sel 0 - LTC Analog
104  input           spi_miso,            //3.3V    //Master Input
105  output          spi_mosi,            //3.3V    //Master Output
106  output          spi_sck,             //3.3V    //Clock Output
107
108 //HPS-QSPI-Flash-----------------------//6 pins  //-------------------------
109  output          qspi_clk,            //3.3V    //Clock
110  inout [3:0]     qspi_io,             //3.3V    //Data
111  output          qspi_ss0,            //3.3V    //Select
112
113 //HPS-SD-Card-Flash--------------------//7 pins  //-------------------------
114  output          sd_clk,              //3.3V    //
115  inout           sd_cmd,              //3.3V    //
116  //output          sd_pwren,            //3.3V    //
117  inout   [3:0]   sd_dat,              //3.3V    //???????????????  should only be (3:0)
         ????????
118   //inout   [7:0]   sd_dat,              //3.3V    //???????????????  should only be (3:0)
         ????????
119
120
121 //HPS-Ethernet----10/100/1000----------//14 pins //-------------------------
122  output          enet_hps_gtx_clk,    //3.3V    //Gb Ethernet Clock
123  //input           enet_hps_intn,       //3.3V    //Placed on HPS GPIO
124  output          enet_hps_mdc,        //3.3V    //MDIO Clock (TR=0)
125  inout           enet_hps_mdio,       //3.3V    //MDIO Data (TR=0)
126  input         enet_hps_rx_clk,     //3.3V    //Receive Data
```

```
  input         enet_hps_rx_dv,       //3.3V    //Receive Data Valid / Cont
  input [3:0]     enet_hps_rxd,         //3.3V    //Receive Data
  output          enet_hps_tx_en,       //3.3V    //Transmit Data Enable / Cont
  output [3:0]    enet_hps_txd,         //3.3V    //Transmit Data

//HPS-CAN-BUS--------------------------//2 pins  //
   input           can_0_rx,             //3.3V    //HPS only
   output          can_0_tx,             //3.3V    //HPS only

//HPS-Trace----------------------------//10 pins //-------------------------
  //input           mictor_rstn,          //3.3V    //??????????????? not in Qsys - needed?
   output          trace_clk_mic,        //3.3V    //
   output [7:0]    trace_data,           //3.3V    //

`endif

`ifdef  HPS_GPIO
//HPS-User-IO--------------------------//X pins  //-------------------------
   inout  [3:0]    user_dipsw_hps,       //1.5V
   inout  [3:0]    user_led_hps,         //3.3V
   inout  [1:0]    user_pb_hps,          //1.5V
`endif

`ifdef    HPS_USB
//HPS-USB-OTG--------------------------//19 pins  //-------------------------
  input           usb_clk,              //3.3V      //Clock
  inout  [7:0]    usb_data,             //3.3V      //
  input           usb_nxt,              //3.3V      //
  input           usb_dir,              //3.3V      //
  output          usb_stp,              //3.3V      //
  //input           usb_empty,
  //input           usb_full,
  //output          usb_oen,
  //output          usb_rdn,
  //output          usb_resetn,
  //output          usb_scl,
  //inout           usb_sda,
  //output          usb_wrn,
`endif

//HPS-DDR3-400Mx40---------------------//78 pins //-------------------------
`ifdef  DDR3_HPS
   output [14:0]  ddr3_hps_a,           //SSTL15  //Address
   output [2:0]   ddr3_hps_ba,          //SSTL15  //Bank Address
   output         ddr3_hps_casn,        //SSTL15  //Column Address Strobe
   output         ddr3_hps_cke,         //SSTL15  //Clock Enable
   output         ddr3_hps_clk_n,       //SSTL15  //Diff Clock - Neg
   output         ddr3_hps_clk_p,       //SSTL15  //Diff Clock - Pos
   output         ddr3_hps_csn,         //SSTL15  //Chip Select
   output [4:0]   ddr3_hps_dm,          //SSTL15  //Data Write Mask
   inout  [39:0]  ddr3_hps_dq,          //SSTL15  //Data Bus
   inout   [4:0]   ddr3_hps_dqs_n,       //SSTL15  //Diff Data Strobe - Neg
  inout  [4:0]   ddr3_hps_dqs_p,        //SSTL15  //Diff Data Strobe - Pos
  output         ddr3_hps_odt,         //SSTL15  //On-Die Termination Enable
  output         ddr3_hps_rasn,        //SSTL15  //Row Address Strobe
  output         ddr3_hps_resetn,      //SSTL15  //Reset
  output         ddr3_hps_wen,         //SSTL15  //Write Enable
  input          ddr3_hps_rzq,         //OCT_rzqin
  //input          ddr3_rup,             //OCT_rup
  //input          ddr3_rdn,             //OCT_rdn
`else
   //output          ddr3_hps_csn,        //SSTL15  //Chip Select
`endif


//-----------------------------------------------------------------------//
////////////////////////////// FPGA Pins ///////////////////////////////////
//-----------------------------------------------------------------------//
```

```
195
196
197 //FPGA-GPLL-CLK-------------------------//X pins
198   input       clk_100m_fpga,          //2.5V     //100 MHz (2nd copy to max)
199   input       clk_50m_fpga,           //2.5V     //50MHz (2nd copy to max)
200   input       clk_25m_fpga,
201   input       clk_top1,               //2.5V     //156.25 MHz adjustable
202   input       clk_bot1,               //1.5V     //100 MHz adjustable
203 //    input           clk_enet_fpga_p,      //LVDS     //125 MHz fixed
204
205 //FPGA-User-IO-------------------------//11 pins //--------------------------
206   input       cpu_resetn,             //2.5V     //NIOS CPU Reset Pushbutton
207   input [3:0] user_dipsw_fpga,        //
208   output  [3:0] user_led_fpga,          //
209   input [1:0] user_pb_fpga,           //
210
211 //FPGA-DDR3-400Mx32---------------------//71 pins //--------------------------
212 `ifdef  DDR3_FPGA
213   output [14:0]  ddr3_fpga_a,           //SSTL15   //Address
214   output [2:0]   ddr3_fpga_ba,          //SSTL15   //Bank Address
215   output         ddr3_fpga_casn,        //SSTL15   //Column Address Strobe
216   output         ddr3_fpga_cke,         //SSTL15   //Clock Enable
217   output         ddr3_fpga_clk_n,       //SSTL15   //Diff Clock - Neg
218   output         ddr3_fpga_clk_p,       //SSTL15   //Diff Clock - Pos
219   output         ddr3_fpga_csn,         //SSTL15   //Chip Select
220   output [3:0]   ddr3_fpga_dm,          //SSTL15   //Data Write Mask
221   inout  [31:0]  ddr3_fpga_dq,          //SSTL15   //Data Bus
222   inout  [3:0]   ddr3_fpga_dqs_n,       //SSTL15   //Diff Data Strobe - Neg
223   inout  [3:0]   ddr3_fpga_dqs_p,       //SSTL15   //Diff Data Strobe - Pos
224   output         ddr3_fpga_odt,         //SSTL15   //On-Die Termination Enable
225   input          ddr3_fpga_rasn,        //SSTL15   //Row Address Strobe
226   input          ddr3_fpga_resetn,      //SSTL15   //Reset
227   input          ddr3_fpga_wen,         //SSTL15   //Write Enable
228   input          ddr3_fpga_rzq,         //OCT_rzqin
229 `else
230    output          ddr3_fpga_csn,        //SSTL15   //Chip Select
231 `endif
232
233
234 //FPGA-Ethernet1 ---10/100--------------//14 pins //--------------------------
235 `ifdef  ENET_FPGA
236   input         enet1_rx_clk,           //2.5V     //Receive Data Clock
237   input [3:0]     enet1_rx_d,           //2.5V     //Receive Data
238   input         enet1_rx_dv,            //2.5V     //Receive Data Valid
239   input         enet1_rx_error,         //2.5V     //Receive Data Error
240   output          enet1_tx_clk_fb,      //2.5V     //Transmit Clock Feedback
241   output [3:0]    enet1_tx_d,           //2.5V     //Transmit Data
242   output          enet1_tx_en,          //2.5V     //Transmit Data Enable
243   input           enet1_tx_error,       //2.5V     //Transmit Data Error
244   output          enet_dual_resetn,     //2.5V     //EtherCat PHY Reset
245   input         enet2_rx_clk,           //2.5V     //Receive Data Clock
246   input [3:0]     enet2_rx_d,           //2.5V     //Receive Data
247   input         enet2_rx_dv,            //2.5V     //Receive Data Valid
248   input         enet2_rx_error,         //2.5V     //Receive Data Error
249   output          enet2_tx_clk_fb,      //2.5V     //Transmit Clock Feedback
250   output [3:0]    enet2_tx_d,           //2.5V     //Transmit Data
251   output          enet2_tx_en,          //2.5V     //Transmit Data Enable
252   input           enet2_tx_error,       //2.5V     //Transmit Data Error
253   input       enet_fpga_mdc,
254   input       enet_fpga_mdio
255 `endif
256
257 //SDI-XCVR-Video-----------------------//X pins  //--------------------------
258 `ifdef  use_SDI_XCVR_refclk_p
259    input          clk_148_p,             //LVDS     //148.5MHz Prog. SDI VCXO
260 `endif
261 `ifdef  SDI_XCVR
262   input          gxb_rx_l4_p,           //PCML     //SDI Receiver (or SMA)
```

```
263   output          gxb_tx_l4_p,          //PCML     //SDI Transmitter (or SMA)
264 `endif
265
266 `ifdef  SDI_FPGA
267   //148.5M programmable VCXO in refclk section above
268   output       sdi_clk148_dn,        //2.5V     //VCXO pump up
269   output       sdi_clk148_up,        //2.5V     //VCXO pump down
270   input        sdi_fault,            //2.5V     //SDI Cable Driver Fault
271   output       sdi_rsti,             //2.5V     //SDI Cable Driver Reset
272   output       sdi_rx_bypass,        //2.5V     //SDI Equalizer Bypass
273   output       sdi_rx_en,            //2.5V     //SDI Equalizer Enable
274   output       sdi_tx_en,            //2.5V     //SDI Cable Driver Enable
275   output       sdi_tx_sd_hdn,        //2.5V     //SDI Cable Driver High-Def
276 `endif
277
278 `ifdef  I2C_FPGA
279   inout        i2c_sda_fpga,
280   inout        i2c_scl_fpga,
281 `endif
282
283
284 `ifdef  MAX_FPGA
285   output   max_fpga_miso,
286   input    max_fpga_mosi,
287   inout    max_fpga_sck,
288   inout    max_fpga_ssel,
289 `endif
290
291
292 //HSMC-Port-A-------------------------//107pins //-------------------------
293 `ifdef  use_HSMA_XCVR_refclk_p
294   input           refclk_ql2_p,        //LVDS     //HSMA Transceiver Refclk -reqs OCT
295 `endif
296 `ifdef  HSMA_XCVR
297     input [3:0]    hsma_rx_p,            //PCML?    //HSMA Receive Data -reqs OCT
298     output [3:0]   hsma_tx_p,            //PCML?    //HSMA Transmit Data
299 `endif
300
301 `ifdef  HSMA_CMOS
302  //Enable below for CMOS HSMC
303   //inout  [79:0]  hsma_d,             //2.5V     //HSMA CMOS Data Bus
304  //Enable below for LVDS HSMC
305    input           hsma_clk_in0,         //2.5V     //Primary single-ended CLKIN
306   input        hsma_clk_in_p1,    // LVCMOS signal
307   input        hsma_clk_in_n1,    // LVCMOS signal
308    input           hsma_clk_in_p2,       //LVDS     //Primary Source-Sync CLKIN
309    output          hsma_clk_out0,        //2.5V     //Primary single-ended CLKOUT
310   output       hsma_clk_out_p1,     // LVCMOS signal
311   output       hsma_clk_out_n1,     // LVCMOS signal
312    output          hsma_clk_out_p2,      //LVDS     //Primary Source-Sync CLKOUT
313    inout  [3:0]    hsma_d,               //2.5V     //Dedicated CMOS IO
314   input  [16:0]  hsma_rx_d_p,          //LVDS     //LVDS Sounce-Sync Input
315   output [16:0]  hsma_tx_d_p,          //LVDS     //LVDS Sounce-Sync Output
316   input          hsma_prsntn,          //2.5V     //HSMC Presence Detect Input
317   output         hsma_scl,             //2.5V     //SMBus Clock
318   inout          hsma_sda,             //2.5V     //SMBus Data
319 `endif
320
321 //PCI-Express-------------------------//X pins  //-------------------------
322 `ifdef  use_PCIE_XCVR_refclk_p
323    input           pcie_refclk_p,        //LVDS     //PCIe Refclk -reqs OCT
324 `endif
325 `ifdef  PCIE_XCVR
326    input  [3:0]    pcie_rx_p,            //PCML     //PCIe Receive Data-req'sOCT
327    output [3:0]    pcie_tx_p,            //PCML     //PCIe Transmit Data
328 `endif
329 //---Found multiple nets with the same name
330 `ifdef  PCIE_FPGA
```

```
331     input       pcie_perstn_in ,    // Pin W21 instance 0
332     output      pcie_perstn_out ,   // Pin AG6 instance 1
333      input        pcie_prsnt2_x1 ,
334    input        pcie_prsnt2_x4 ,
335    inout        pcie_smbclk ,
336    inout        pcie_smbdat ,
337    inout        pcie_waken
338  `endif
339
340  );
341
342
343  // Rev A board Ethernet reset can interrupt device configuration, to fix
344  // enet_dual_reset tristate as input or drive as output to 1
345  assign  enet_dual_resetn  = 1'b1;
346  // FPGA user dipswitches drive user LEDs
347  //assign    user_led_fpga     = ~user_dipsw_fpga;
348  // Disable FPGA DDR3
349  //assign    ddr3_fpga_csn     = 1'b1;
350
351
352
353
354  //*****From golden_top by Altera Corp.*****
355  // internal wires and registers declaration
356    wire [1:0] fpga_debounced_buttons;
357    wire [3:0] fpga_led_internal;
358    wire       hps_fpga_reset_n;
359
360  // connection of internal logics
361  //  assign fpga_led_internal = user_led_fpga;
362
363
364  // Debounce logic to clean out glitches within 1ms
365  debounce debounce_inst (
366    .clk                                  (clk_50m_fpga),
367    .reset_n                              (hps_fpga_reset_n),
368    .data_in                              (user_pb_fpga[0]),
369    .data_out                             (fpga_debounced_buttons)
370  );
371    defparam debounce_inst.WIDTH = 2;
372    defparam debounce_inst.POLARITY = "LOW";
373    defparam debounce_inst.TIMEOUT = 50000;               // at 50Mhz this is a debounce
         time of 1ms
374    defparam debounce_inst.TIMEOUT_WIDTH = 16;            // ceil(log2(TIMEOUT))
375
376  // Due to PLLs the clock pins need to be buffered when connected internal clock network
377  //wire clk_50m_net;
378  //wire clk_100m_net;
379
380  //clk_cntrl clk_cntrl_50m (
381  //   .inclk ( clk_50m_fpga ),
382  //   .outclk ( clk_50m_net )
383  //);
384  //clk_cntrl clk_cntrl_100m (
385  //   .inclk ( clk_100m_fpga ),
386  //   .outclk ( clk_100m_net )
387  //);
388
389  // **** IMPORTANT NOTE ****
390  // Need to use clk_bot1 (100 Mhz (adjustable) to drive PLL due to fitter constraint
391  // DO NOT use clk_100m_fpga as it is in the wrong IO block for the FPGA
392
393   orion_system u0 (
394      .memory_mem_a                         (ddr3_hps_a),
           //                            memory.mem_a
395      .memory_mem_ba                        (ddr3_hps_ba),
           //                               .mem_ba
```

```
396     .memory_mem_ck                                  (ddr3_hps_clk_p),
                                                    //                                  .mem_ck
397     .memory_mem_ck_n                                (ddr3_hps_clk_n),
                                                  //                                    .mem_ck_n
398     .memory_mem_cke                                 (ddr3_hps_cke),
            //                                      .mem_cke
399     .memory_mem_cs_n                                (ddr3_hps_csn),
            //                                      .mem_cs_n
400     .memory_mem_ras_n                               (ddr3_hps_rasn),
            //                                      .mem_ras_n
401     .memory_mem_cas_n                               (ddr3_hps_casn),
            //                                      .mem_cas_n
402     .memory_mem_we_n                                (ddr3_hps_wen),
            //                                      .mem_we_n
403     .memory_mem_reset_n                             (ddr3_hps_resetn),
            //                                      .mem_reset_n
404     .memory_mem_dq                                  (ddr3_hps_dq),
            //                                      .mem_dq
405     .memory_mem_dqs                                 (ddr3_hps_dqs_p),
                                                    //                                  .mem_dqs
406     .memory_mem_dqs_n                               (ddr3_hps_dqs_n),
            //                                      .mem_dqs_n
407     .memory_mem_odt                                 (ddr3_hps_odt),
            //                                      .mem_odt
408     .memory_mem_dm                                  (ddr3_hps_dm),
            //                                      .mem_dm
409     .memory_oct_rzqin                               (ddr3_hps_rzq),                     //
                                                  .oct_rzqin
410     .hps_0_hps_io_hps_io_emac1_inst_TX_CLK      (enet_hps_gtx_clk),     //
                                  hps_0_hps_io.hps_io_emac1_inst_TX_CLK
411     .hps_0_hps_io_hps_io_emac1_inst_TXD0        (enet_hps_txd[0]),      //
                                                .hps_io_emac1_inst_TXD0
412     .hps_0_hps_io_hps_io_emac1_inst_TXD1        (enet_hps_txd[1]),      //
                                                .hps_io_emac1_inst_TXD1
413     .hps_0_hps_io_hps_io_emac1_inst_TXD2        (enet_hps_txd[2]),      //
                                                .hps_io_emac1_inst_TXD2
414     .hps_0_hps_io_hps_io_emac1_inst_TXD3        (enet_hps_txd[3]),      //
                                                .hps_io_emac1_inst_TXD3
415     .hps_0_hps_io_hps_io_emac1_inst_RXD0        (enet_hps_rxd[0]),      //
                                                .hps_io_emac1_inst_RXD0
416     .hps_0_hps_io_hps_io_emac1_inst_MDIO        (enet_hps_mdio),        //
                                                .hps_io_emac1_inst_MDIO
417     .hps_0_hps_io_hps_io_emac1_inst_MDC         (enet_hps_mdc),         //
                                                .hps_io_emac1_inst_MDC
418     .hps_0_hps_io_hps_io_emac1_inst_RX_CTL      (enet_hps_rx_dv),       //
                                                .hps_io_emac1_inst_RX_CTL
419     .hps_0_hps_io_hps_io_emac1_inst_TX_CTL      (enet_hps_tx_en),       //
                                                .hps_io_emac1_inst_TX_CTL
420     .hps_0_hps_io_hps_io_emac1_inst_RX_CLK      (enet_hps_rx_clk),      //
                                                .hps_io_emac1_inst_RX_CLK
421     .hps_0_hps_io_hps_io_emac1_inst_RXD1        (enet_hps_rxd[1]),      //
                                                .hps_io_emac1_inst_RXD1
422     .hps_0_hps_io_hps_io_emac1_inst_RXD2        (enet_hps_rxd[2]),      //
                                                .hps_io_emac1_inst_RXD2
423     .hps_0_hps_io_hps_io_emac1_inst_RXD3        (enet_hps_rxd[3]),      //
                                                .hps_io_emac1_inst_RXD3
424     .hps_0_hps_io_hps_io_qspi_inst_IO0          (qspi_io[0]),           //
                                                .hps_io_qspi_inst_IO0
425     .hps_0_hps_io_hps_io_qspi_inst_IO1          (qspi_io[1]),           //
                                                .hps_io_qspi_inst_IO1
426     .hps_0_hps_io_hps_io_qspi_inst_IO2          (qspi_io[2]),           //
                                                .hps_io_qspi_inst_IO2
427     .hps_0_hps_io_hps_io_qspi_inst_IO3          (qspi_io[3]),           //
                                                .hps_io_qspi_inst_IO3
428     .hps_0_hps_io_hps_io_qspi_inst_SS0          (qspi_ss0),           //
                                                .hps_io_qspi_inst_SS0
429     .hps_0_hps_io_hps_io_qspi_inst_CLK          (qspi_clk),           //
                                                .hps_io_qspi_inst_CLK
```

```
430     .hps_0_hps_io_hps_io_sdio_inst_CMD          (sd_cmd),          //
                                    .hps_io_sdio_inst_CMD
431     //.hps_0_hps_io_hps_io_sdio_inst_PWREN       (sd_pwren),                    //
                                    .hps_io_sdio_inst_PWREN
432     .hps_0_hps_io_hps_io_sdio_inst_D0           (sd_dat[0]),          //
                                    .hps_io_sdio_inst_D0
433     .hps_0_hps_io_hps_io_sdio_inst_D1           (sd_dat[1]),          //
                                    .hps_io_sdio_inst_D1
434     .hps_0_hps_io_hps_io_sdio_inst_CLK          (sd_clk),          //
                                    .hps_io_sdio_inst_CLK
435     .hps_0_hps_io_hps_io_sdio_inst_D2           (sd_dat[2]),          //
                                    .hps_io_sdio_inst_D2
436     .hps_0_hps_io_hps_io_sdio_inst_D3           (sd_dat[3]),          //
                                    .hps_io_sdio_inst_D3
437     .hps_0_hps_io_hps_io_usb1_inst_D0           (usb_data[0]),                    //
                                    .hps_io_usb1_inst_D0
438      .hps_0_hps_io_hps_io_usb1_inst_D1          (usb_data[1]),                     //
                                    .hps_io_usb1_inst_D1
439      .hps_0_hps_io_hps_io_usb1_inst_D2          (usb_data[2]),                     //
                                    .hps_io_usb1_inst_D2
440      .hps_0_hps_io_hps_io_usb1_inst_D3          (usb_data[3]),                     //
                                    .hps_io_usb1_inst_D3
441      .hps_0_hps_io_hps_io_usb1_inst_D4          (usb_data[4]),                     //
                                    .hps_io_usb1_inst_D4
442      .hps_0_hps_io_hps_io_usb1_inst_D5          (usb_data[5]),                     //
                                    .hps_io_usb1_inst_D5
443      .hps_0_hps_io_hps_io_usb1_inst_D6          (usb_data[6]),                     //
                                    .hps_io_usb1_inst_D6
444      .hps_0_hps_io_hps_io_usb1_inst_D7          (usb_data[7]),                     //
                                    .hps_io_usb1_inst_D7
445      .hps_0_hps_io_hps_io_usb1_inst_CLK         (usb_clk),                    //
                                    .hps_io_usb1_inst_CLK
446      .hps_0_hps_io_hps_io_usb1_inst_STP         (usb_stp),                    //
                                    .hps_io_usb1_inst_STP
447      .hps_0_hps_io_hps_io_usb1_inst_DIR         (usb_dir),                    //
                                    .hps_io_usb1_inst_DIR
448      .hps_0_hps_io_hps_io_usb1_inst_NXT         (usb_nxt),                    //
                                    .hps_io_usb1_inst_NXT
449     .hps_0_hps_io_hps_io_spim0_inst_CLK         (spi_sck),          //
                                    .hps_io_spim0_inst_CLK
450     .hps_0_hps_io_hps_io_spim0_inst_MOSI        (spi_mosi),          //
                                    .hps_io_spim0_inst_MOSI
451     .hps_0_hps_io_hps_io_spim0_inst_MISO        (spi_miso),          //
                                    .hps_io_spim0_inst_MISO
452     .hps_0_hps_io_hps_io_spim0_inst_SS0         (spi_csn),          //
                                    .hps_io_spim0_inst_SS0
453     .hps_0_hps_io_hps_io_uart0_inst_RX          (uart_rx),           //
                                    .hps_io_uart0_inst_RX
454     .hps_0_hps_io_hps_io_uart0_inst_TX          (uart_tx),           //
                                    .hps_io_uart0_inst_TX
455     .hps_0_hps_io_hps_io_i2c0_inst_SDA          (i2c_sda_hps),           //
                                    .hps_io_i2c0_inst_SDA
456     .hps_0_hps_io_hps_io_i2c0_inst_SCL          (i2c_scl_hps),           //
                                    .hps_io_i2c0_inst_SCL
457     .hps_0_hps_io_hps_io_can0_inst_RX           (can_0_rx),          //
                                    .hps_io_can0_inst_RX
458     .hps_0_hps_io_hps_io_can0_inst_TX           (can_0_tx),          //
                                    .hps_io_can0_inst_TX
459     .hps_0_hps_io_hps_io_trace_inst_CLK         (trace_clk_mic),          //
                                    .hps_io_trace_inst_CLK
460     .hps_0_hps_io_hps_io_trace_inst_D0          (trace_data[0]),          //
                                    .hps_io_trace_inst_D0
461     .hps_0_hps_io_hps_io_trace_inst_D1          (trace_data[1]),          //
                                    .hps_io_trace_inst_D1
462     .hps_0_hps_io_hps_io_trace_inst_D2          (trace_data[2]),          //
                                    .hps_io_trace_inst_D2
463     .hps_0_hps_io_hps_io_trace_inst_D3          (trace_data[3]),          //
                                    .hps_io_trace_inst_D3
```

```
464     .hps_0_hps_io_hps_io_trace_inst_D4          (trace_data[4]),          //
                                    .hps_io_trace_inst_D4
465     .hps_0_hps_io_hps_io_trace_inst_D5          (trace_data[5]),          //
                                    .hps_io_trace_inst_D5
466     .hps_0_hps_io_hps_io_trace_inst_D6          (trace_data[6]),          //
                                    .hps_io_trace_inst_D6
467     .hps_0_hps_io_hps_io_trace_inst_D7          (trace_data[7]),          //
                                    .hps_io_trace_inst_D7
468 //    .hps_0_hps_io_hps_io_gpio_inst_GPIO04      (user_dipsw_hps[0]),                    //
                                  .hps_io_gpio_inst_GPIO04
469 //    .hps_0_hps_io_hps_io_gpio_inst_GPIO05      (user_dipsw_hps[1]),                    //
                                  .hps_io_gpio_inst_GPIO05
470 //    .hps_0_hps_io_hps_io_gpio_inst_GPIO06      (user_dipsw_hps[2]),                    //
                                  .hps_io_gpio_inst_GPIO06
471 //    .hps_0_hps_io_hps_io_gpio_inst_GPIO07      (user_dipsw_hps[3]),                    //
                                  .hps_io_gpio_inst_GPIO07
472 //    .hps_0_hps_io_hps_io_gpio_inst_GPIO08      (user_pb_hps[0]),                 //
                                  .hps_io_gpio_inst_GPIO08
473 //    .hps_0_hps_io_hps_io_gpio_inst_GPIO09      (user_pb_hps[1]),                 //
                                  .hps_io_gpio_inst_GPIO09
474 //    .hps_0_hps_io_hps_io_gpio_inst_GPIO41      (user_led_hps[3]),                  //
                                  .hps_io_gpio_inst_GPIO41
475 //    .hps_0_hps_io_hps_io_gpio_inst_GPIO42      (user_led_hps[2]),                  //
                                  .hps_io_gpio_inst_GPIO42
476 //    .hps_0_hps_io_hps_io_gpio_inst_GPIO43      (user_led_hps[1]),                  //
                                  .hps_io_gpio_inst_GPIO43
477 //    .hps_0_hps_io_hps_io_gpio_inst_GPIO44      (user_led_hps[0]),                  //
                                  .hps_io_gpio_inst_GPIO44
478     .clk_50_clk                                 (clk_50m_fpga),
                                  //                                clk_50.clk
479     .led_pio_external_connection_in_port        (user_led_fpga),        //
        led_pio_external_connection.in_port
480     .led_pio_external_connection_out_port       (user_led_fpga),       //
                                  .out_port
481     .dipsw_pio_external_connection_export       (user_dipsw_fpga),       //
        dipsw_pio_external_connection.export
482     .button_pio_external_connection_export      (fpga_debounced_buttons),     //
        button_pio_external_connection.export
483     .hps_0_h2f_reset_reset_n                    (hps_fpga_reset_n),                 //
                          hps_0_h2f_reset.reset_n
484     .fpga_memory_mem_a                          (ddr3_fpga_a),                    //
                               fpga_memory.mem_a
485     .fpga_memory_mem_ba                         (ddr3_fpga_ba),                   //
                                  .mem_ba
486     .fpga_memory_mem_ck                         (ddr3_fpga_clk_p),
        //                           .mem_ck
487     .fpga_memory_mem_ck_n                       (ddr3_fpga_clk_n),                  //
                                  .mem_ck_n
488     .fpga_memory_mem_cke                        (ddr3_fpga_cke),                  //
                                  .mem_cke
489     .fpga_memory_mem_cs_n                       (ddr3_fpga_csn),                  //
                                  .mem_cs_n
490     .fpga_memory_mem_dm                         (ddr3_fpga_dm),                   //
                                  .mem_dm
491     .fpga_memory_mem_ras_n                      (ddr3_fpga_rasn),                 //
                                  .mem_ras_n
492     .fpga_memory_mem_cas_n                      (ddr3_fpga_casn),                 //
                                  .mem_cas_n
493     .fpga_memory_mem_we_n                       (ddr3_fpga_wen),                  //
                                  .mem_we_n
494     .fpga_memory_mem_reset_n                    (ddr3_fpga_resetn),               //
                                  .mem_reset_n
495     .fpga_memory_mem_dq                         (ddr3_fpga_dq),                   //
                                  .mem_dq
496     .fpga_memory_mem_dqs                        (ddr3_fpga_dqs_p),                  //
                                   .mem_dqs
497     .fpga_memory_mem_dqs_n                      (ddr3_fpga_dqs_n),                //
                                  .mem_dqs_n
```

```
    .fpga_memory_mem_odt                            (ddr3_fpga_odt),                          //
                                                .mem_odt
    .fpga_memory_oct_rzqin                          (ddr3_fpga_rzq),                        //
                                            fpga_memory_oct.rzqin
    .refclk_100_clk                                 (clk_bot1),                               //
                                        refclk_100.clk
    .reset_50_reset_n                                                                         //
                                                                    reset_50.reset_n
    .mem_if_ddr3_fpga_status_local_init_done          (),                   //
        mem_if_ddr3_fpga_status.local_init_done
    .mem_if_ddr3_fpga_status_local_cal_success        (),                 //
                                                .local_cal_success
    .mem_if_ddr3_fpga_status_local_cal_fail           (),                   //
                                                .local_cal_fail
    .mem_if_ddr3_fpga_pll_sharing_pll_mem_clk         (),                 //
        mem_if_ddr3_fpga_pll_sharing.pll_mem_clk
    .mem_if_ddr3_fpga_pll_sharing_pll_write_clk       (),               //
                                                .pll_write_clk
    .mem_if_ddr3_fpga_pll_sharing_pll_write_clk_pre_phy_clk (), //
                                                .pll_write_clk_pre_phy_clk
    .mem_if_ddr3_fpga_pll_sharing_pll_addr_cmd_clk    (),             //
                                                .pll_addr_cmd_clk
    .mem_if_ddr3_fpga_pll_sharing_pll_locked          (),                   //
                                                .pll_locked
    .mem_if_ddr3_fpga_pll_sharing_pll_avl_clk         (),                 //
                                                .pll_avl_clk
    .mem_if_ddr3_fpga_pll_sharing_pll_config_clk      (),               //
                                                .pll_config_clk
    .mem_if_ddr3_fpga_pll_sharing_pll_mem_phy_clk     (),               //
                                                .pll_mem_phy_clk
    .mem_if_ddr3_fpga_pll_sharing_afi_phy_clk         (),                   //
                                                .afi_phy_clk
    .mem_if_ddr3_fpga_pll_sharing_pll_avl_phy_clk     ()                //
                                                .pll_avl_phy_clk
);


endmodule
```

Listing 16.3: Top Level Verilog (orion_top.v)

16.3.5 Timing Constraint File (timing.base.sdc)

```
#Timing constraint commands from DDR3 Demo from Board Test System by Altera Corporation
#Some Modifications by Christopher Yarp

derive_pll_clocks
derive_clock_uncertainty

#set_false_path -from * -to {sld_signaltap:auto_signaltap_0|*}
#set_false_path -from {sld_signaltap:auto_signaltap_0|*} -to *

#set sys_clock_hmc {u0|pll_0|altera_pll_i|general[0].gpll~PLL_OUTPUT_COUNTER|divclk}

#set_false_path -from [get_ports hps_fpga_reset_n] -to *

# Timing file copied from Holden Hardware Reference Design (GHRD)
#modified for ports in use

# 50MHz board input clock
create_clock -period 20 [get_ports clk_50m_fpga]
create_clock -period 10 [get_ports clk_100m_fpga]

# for enhancing USB BlasterII to be reliable, 25MHz
create_clock -name {altera_reserved_tck} -period 40 {altera_reserved_tck}
```

```
set_input_delay -clock altera_reserved_tck -clock_fall 3 [get_ports altera_reserved_tdi]
set_input_delay -clock altera_reserved_tck -clock_fall 3 [get_ports altera_reserved_tms]
set_output_delay -clock altera_reserved_tck -clock_fall 3 [get_ports altera_reserved_tdo]

# FPGA IO port constraints
set_false_path -from [get_ports {fpga_button_pio[0]}] -to *
set_false_path -from [get_ports {fpga_button_pio[1]}] -to *
set_false_path -from [get_ports {fpga_dipsw_pio[0]}] -to *
set_false_path -from [get_ports {fpga_dipsw_pio[1]}] -to *
set_false_path -from [get_ports {fpga_dipsw_pio[2]}] -to *
set_false_path -from [get_ports {fpga_dipsw_pio[3]}] -to *
set_false_path -from * -to [get_ports {fpga_led_pio[0]}]
set_false_path -from * -to [get_ports {fpga_led_pio[1]}]
set_false_path -from * -to [get_ports {fpga_led_pio[2]}]
set_false_path -from * -to [get_ports {fpga_led_pio[3]}]

# HPS peripherals port false path setting to workaround the unconstrained path (setting
    false_path for hps_0 ports will not affect the routing as it is hard silicon)
# **** outputs ****
set_false_path -from * -to [get_ports {uart_tx}]

set_false_path -from * -to [get_ports {spi_csn}]
set_false_path -from * -to [get_ports {spi_mosi}]
set_false_path -from * -to [get_ports {spi_sck}]

set_false_path -from * -to [get_ports {qspi_clk}]
set_false_path -from * -to [get_ports {qspi_ss0}]

set_false_path -from * -to [get_ports {sd_clk}]
set_false_path -from * -to [get_ports {sd_pwren}]

set_false_path -from * -to [get_ports {enet_hps_gtx_clk}]
set_false_path -from * -to [get_ports {enet_hps_mdc}]
set_false_path -from * -to [get_ports {enet_hps_tx_en}]
set_false_path -from * -to [get_ports {enet_hps_txd[0]}]
set_false_path -from * -to [get_ports {enet_hps_txd[1]}]
set_false_path -from * -to [get_ports {enet_hps_txd[2]}]
set_false_path -from * -to [get_ports {enet_hps_txd[3]}]

set_false_path -from * -to [get_ports {can_0_tx}]

set_false_path -from * -to [get_ports {trace_clk_mic}]
set_false_path -from * -to [get_ports {trace_data[0]}]
set_false_path -from * -to [get_ports {trace_data[1]}]
set_false_path -from * -to [get_ports {trace_data[2]}]
set_false_path -from * -to [get_ports {trace_data[3]}]
set_false_path -from * -to [get_ports {trace_data[4]}]
set_false_path -from * -to [get_ports {trace_data[5]}]
set_false_path -from * -to [get_ports {trace_data[6]}]
set_false_path -from * -to [get_ports {trace_data[7]}]

# **** inputs ****
set_false_path -from [get_ports {uart_rx}] -to *
set_false_path -from [get_ports {spi_miso}] -to *

set_false_path -from [get_ports {enet_hps_rx_clk}] -to *
set_false_path -from [get_ports {enet_hps_rx_dv}] -to *
set_false_path -from [get_ports {enet_hps_rxd[0]}] -to *
set_false_path -from [get_ports {enet_hps_rxd[1]}] -to *
set_false_path -from [get_ports {enet_hps_rxd[2]}] -to *
set_false_path -from [get_ports {enet_hps_rxd[3]}] -to *

set_false_path -from [get_ports {can_0_rx}] -to *

# **** inouts ****
set_false_path -from * -to [get_ports {i2c_scl_hps}]
set_false_path -from [get_ports {i2c_scl_hps}] -to *
set_false_path -from * -to [get_ports {i2c_sda_hps}]
```

```
set_false_path -from [get_ports {i2c_sda_hps}] -to *

set_false_path -from * -to [get_ports {qspi_io[0]}]
set_false_path -from [get_ports {qspi_io[0]}] -to *
set_false_path -from * -to [get_ports {qspi_io[1]}]
set_false_path -from [get_ports {qspi_io[1]}] -to *
set_false_path -from * -to [get_ports {qspi_io[2]}]
set_false_path -from [get_ports {qspi_io[2]}] -to *
set_false_path -from * -to [get_ports {qspi_io[3]}]
set_false_path -from [get_ports {qspi_io[3]}] -to *

set_false_path -from * -to [get_ports {sd_cmd}]
set_false_path -from [get_ports {sd_cmd}] -to *
set_false_path -from * -to [get_ports {sd_dat[0]}]
set_false_path -from [get_ports {sd_dat[0]}] -to *
set_false_path -from * -to [get_ports {sd_dat[1]}]
set_false_path -from [get_ports {sd_dat[1]}] -to *
set_false_path -from * -to [get_ports {sd_dat[2]}]
set_false_path -from [get_ports {sd_dat[2]}] -to *
set_false_path -from * -to [get_ports {sd_dat[3]}]
set_false_path -from [get_ports {sd_dat[3]}] -to *

set_false_path -from * -to [get_ports {enet_hps_mdio}]
set_false_path -from [get_ports {enet_hps_mdio}] -to *

#-----------------Orig----------------------------------------

## Copyright (C) 1991-2013 Altera Corporation
## Your use of Altera Corporation's design tools, logic functions
## and other software and tools, and its AMPP partner logic
## functions, and any output files from any of the foregoing
## (including device programming or simulation files), and any
## associated documentation or information are expressly subject
## to the terms and conditions of the Altera Program License
## Subscription Agreement, Altera MegaCore Function License
## Agreement, or other applicable license agreement, including,
## without limitation, that your use is for the sole purpose of
## programming logic devices manufactured by Altera and sold by
## Altera or its authorized distributors.  Please refer to the
## applicable agreement for further details.

#set_false_path -from [get_keepers {*user_dipsw*}]
#set_false_path -from [get_keepers {*user_pb*}]
#set_false_path -to   [get_keepers {*user_led*}]


#**************************************************************
# Set Multicycle Path
#**************************************************************



#**************************************************************
# Set Maximum Delay
#**************************************************************



#**************************************************************
# Set Minimum Delay
#**************************************************************



#**************************************************************
# Set Input Transition
#**************************************************************
```

Listing 16.4: Timing Constraint File (timing.base.sdc)

16.3.6 Quartus II Project File (orion_system.qpf)

```
# -------------------------------------------------------------------------- #
#
# Copyright (C) 1991-2012 Altera Corporation
# Your use of Altera Corporation's design tools, logic functions
# and other software and tools, and its AMPP partner logic
# functions, and any output files from any of the foregoing
# (including device programming or simulation files), and any
# associated documentation or information are expressly subject
# to the terms and conditions of the Altera Program License
# Subscription Agreement, Altera MegaCore Function License
# Agreement, or other applicable license agreement, including,
# without limitation, that your use is for the sole purpose of
# programming logic devices manufactured by Altera and sold by
# Altera or its authorized distributors.  Please refer to the
# applicable agreement for further details.
#
# -------------------------------------------------------------------------- #
#
# Quartus II 64-Bit
# Version 13.0 Internal Build 69 11/14/2012 SJ Full Version
# Date created = 18:45:45  November 15, 2012
#
# -------------------------------------------------------------------------- #

QUARTUS_VERSION = "13.0"
DATE = "18:45:45  November 15, 2012"

# Revisions

PROJECT_REVISION = "orion_system"
```

Listing 16.5: Quartus II Project File (orion_system.qpf)

16.3.7 Quartus II Settings File (orion_system.qsf)

```
# -------------------------------------------------------------------------- #
#
# Copyright (C) 1991-2012 Altera Corporation
# Your use of Altera Corporation's design tools, logic functions
# and other software and tools, and its AMPP partner logic
# functions, and any output files from any of the foregoing
# (including device programming or simulation files), and any
# associated documentation or information are expressly subject
# to the terms and conditions of the Altera Program License
# Subscription Agreement, Altera MegaCore Function License
# Agreement, or other applicable license agreement, including,
# without limitation, that your use is for the sole purpose of
# programming logic devices manufactured by Altera and sold by
# Altera or its authorized distributors.  Please refer to the
# applicable agreement for further details.
#
# -------------------------------------------------------------------------- #
#
# Quartus II 64-Bit
# Version 13.0 Internal Build 69 11/14/2012 SJ Full Version
# Date created = 18:45:45  November 15, 2012
#
# -------------------------------------------------------------------------- #
```

```
#
# Notes:
#
# 1) The default values for assignments are stored in the file:
#    bts_xcvr_assignment_defaults.qdf
#     If this file doesn't exist, see file:
#    assignment_defaults.qdf
#
# 2) Altera recommends that you do not modify this file. This
#     file is updated automatically by the Quartus II software
#     and any changes you make may be lost or overwritten.
#
# ------------------------------------------------------------------------- #


set_global_assignment -name FAMILY "Cyclone V"
set_global_assignment -name DEVICE 5CSXFC6D6F31C8ES
set_global_assignment -name ORIGINAL_QUARTUS_VERSION 12.1
set_global_assignment -name PROJECT_CREATION_TIME_DATE "22:55:46  OCTOBER 03, 2012"
set_global_assignment -name LAST_QUARTUS_VERSION 13.0
set_global_assignment -name PROJECT_OUTPUT_DIRECTORY output_files_sof_rpt
set_global_assignment -name ERROR_CHECK_FREQUENCY_DIVISOR 256
set_global_assignment -name EDA_SIMULATION_TOOL "ModelSim-Altera (Verilog)"
set_global_assignment -name EDA_TIME_SCALE "1 ps" -section_id eda_simulation
set_global_assignment -name EDA_OUTPUT_DATA_FORMAT "VERILOG HDL" -section_id
    eda_simulation
set_global_assignment -name TOP_LEVEL_ENTITY orion_top
set_global_assignment -name POWER_PRESET_COOLING_SOLUTION "23 MM HEAT SINK WITH 200 LFPM
    AIRFLOW"
set_global_assignment -name POWER_BOARD_THERMAL_MODEL "NONE (CONSERVATIVE)"
set_global_assignment -name UNIPHY_SEQUENCER_DQS_CONFIG_ENABLE ON
set_global_assignment -name OPTIMIZE_MULTI_CORNER_TIMING ON
set_global_assignment -name ECO_REGENERATE_REPORT ON
set_global_assignment -name STRATIX_DEVICE_IO_STANDARD "2.5 V"
set_global_assignment -name STRATIXV_CONFIGURATION_SCHEME "PASSIVE PARALLEL X16"
set_global_assignment -name USE_CONFIGURATION_DEVICE OFF
set_global_assignment -name GENERATE_RBF_FILE ON
set_global_assignment -name GENERATE_HEX_FILE ON
set_global_assignment -name CRC_ERROR_OPEN_DRAIN ON
set_global_assignment -name ON_CHIP_BITSTREAM_DECOMPRESSION OFF
set_global_assignment -name RESERVE_DATA15_THROUGH_DATA8_AFTER_CONFIGURATION "AS INPUT
    TRI-STATED"
set_global_assignment -name RESERVE_DATA7_THROUGH_DATA5_AFTER_CONFIGURATION "AS INPUT
    TRI-STATED"
set_global_assignment -name OUTPUT_IO_TIMING_NEAR_END_VMEAS "HALF VCCIO" -rise
set_global_assignment -name OUTPUT_IO_TIMING_NEAR_END_VMEAS "HALF VCCIO" -fall
set_global_assignment -name OUTPUT_IO_TIMING_FAR_END_VMEAS "HALF SIGNAL SWING" -rise
set_global_assignment -name OUTPUT_IO_TIMING_FAR_END_VMEAS "HALF SIGNAL SWING" -fall
set_location_assignment PIN_D25 -to clk_osc1
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to clk_osc1
set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_rzq -tag
    __hps_sdram_p0
set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_dq[0] -tag
    __hps_sdram_p0
set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_dq[1] -tag
    __hps_sdram_p0
set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_dq[2] -tag
    __hps_sdram_p0
set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_dq[3] -tag
    __hps_sdram_p0
set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_dq[4] -tag
    __hps_sdram_p0
set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_dq[5] -tag
    __hps_sdram_p0
set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_dq[6] -tag
    __hps_sdram_p0
set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_dq[7] -tag
    __hps_sdram_p0
```

```
79 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_dq[8] -tag
       __hps_sdram_p0
80 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_dq[9] -tag
       __hps_sdram_p0
81 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_dq[10] -tag
       __hps_sdram_p0
82 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_dq[11] -tag
       __hps_sdram_p0
83 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_dq[12] -tag
       __hps_sdram_p0
84 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_dq[13] -tag
       __hps_sdram_p0
85 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_dq[14] -tag
       __hps_sdram_p0
86 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_dq[15] -tag
       __hps_sdram_p0
87 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_dq[16] -tag
       __hps_sdram_p0
88 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_dq[17] -tag
       __hps_sdram_p0
89 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_dq[18] -tag
       __hps_sdram_p0
90 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_dq[19] -tag
       __hps_sdram_p0
91 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_dq[20] -tag
       __hps_sdram_p0
92 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_dq[21] -tag
       __hps_sdram_p0
93 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_dq[22] -tag
       __hps_sdram_p0
94 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_dq[23] -tag
       __hps_sdram_p0
95 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_dq[24] -tag
       __hps_sdram_p0
96 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_dq[25] -tag
       __hps_sdram_p0
97 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_dq[26] -tag
       __hps_sdram_p0
98 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_dq[27] -tag
       __hps_sdram_p0
99 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_dq[28] -tag
       __hps_sdram_p0
100 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_dq[29] -tag
       __hps_sdram_p0
101 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_dq[30] -tag
       __hps_sdram_p0
102 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_dq[31] -tag
       __hps_sdram_p0
103 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_dq[32] -tag
       __hps_sdram_p0
104 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_dq[33] -tag
       __hps_sdram_p0
105 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_dq[34] -tag
       __hps_sdram_p0
106 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_dq[35] -tag
       __hps_sdram_p0
107 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_dq[36] -tag
       __hps_sdram_p0
108 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_dq[37] -tag
       __hps_sdram_p0
109 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_dq[38] -tag
       __hps_sdram_p0
110 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_dq[39] -tag
       __hps_sdram_p0
111 set_instance_assignment -name IO_STANDARD "DIFFERENTIAL 1.5-V SSTL CLASS I" -to
       ddr3_hps_dqs_p[0] -tag __hps_sdram_p0
112 set_instance_assignment -name IO_STANDARD "DIFFERENTIAL 1.5-V SSTL CLASS I" -to
       ddr3_hps_dqs_p[1] -tag __hps_sdram_p0
```

```
113 set_instance_assignment -name IO_STANDARD "DIFFERENTIAL 1.5-V SSTL CLASS I" -to
        ddr3_hps_dqs_p[2] -tag __hps_sdram_p0
114 set_instance_assignment -name IO_STANDARD "DIFFERENTIAL 1.5-V SSTL CLASS I" -to
        ddr3_hps_dqs_p[3] -tag __hps_sdram_p0
115 set_instance_assignment -name IO_STANDARD "DIFFERENTIAL 1.5-V SSTL CLASS I" -to
        ddr3_hps_dqs_p[4] -tag __hps_sdram_p0
116 set_instance_assignment -name IO_STANDARD "DIFFERENTIAL 1.5-V SSTL CLASS I" -to
        ddr3_hps_dqs_n[0] -tag __hps_sdram_p0
117 set_instance_assignment -name IO_STANDARD "DIFFERENTIAL 1.5-V SSTL CLASS I" -to
        ddr3_hps_dqs_n[1] -tag __hps_sdram_p0
118 set_instance_assignment -name IO_STANDARD "DIFFERENTIAL 1.5-V SSTL CLASS I" -to
        ddr3_hps_dqs_n[2] -tag __hps_sdram_p0
119 set_instance_assignment -name IO_STANDARD "DIFFERENTIAL 1.5-V SSTL CLASS I" -to
        ddr3_hps_dqs_n[3] -tag __hps_sdram_p0
120 set_instance_assignment -name IO_STANDARD "DIFFERENTIAL 1.5-V SSTL CLASS I" -to
        ddr3_hps_dqs_n[4] -tag __hps_sdram_p0
121 set_instance_assignment -name IO_STANDARD "DIFFERENTIAL 1.5-V SSTL CLASS I" -to
        ddr3_hps_clk_p -tag __hps_sdram_p0
122 set_instance_assignment -name IO_STANDARD "DIFFERENTIAL 1.5-V SSTL CLASS I" -to
        ddr3_hps_clk_n -tag __hps_sdram_p0
123 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_a[0] -tag
        __hps_sdram_p0
124 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_a[10] -tag
        __hps_sdram_p0
125 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_a[11] -tag
        __hps_sdram_p0
126 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_a[12] -tag
        __hps_sdram_p0
127 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_a[13] -tag
        __hps_sdram_p0
128 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_a[14] -tag
        __hps_sdram_p0
129 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_a[1] -tag
        __hps_sdram_p0
130 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_a[2] -tag
        __hps_sdram_p0
131 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_a[3] -tag
        __hps_sdram_p0
132 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_a[4] -tag
        __hps_sdram_p0
133 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_a[5] -tag
        __hps_sdram_p0
134 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_a[6] -tag
        __hps_sdram_p0
135 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_a[7] -tag
        __hps_sdram_p0
136 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_a[8] -tag
        __hps_sdram_p0
137 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_a[9] -tag
        __hps_sdram_p0
138 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_ba[0] -tag
        __hps_sdram_p0
139 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_ba[1] -tag
        __hps_sdram_p0
140 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_ba[2] -tag
        __hps_sdram_p0
141 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_casn -tag
        __hps_sdram_p0
142 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_cke -tag
        __hps_sdram_p0
143 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_csn -tag
        __hps_sdram_p0
144 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_odt -tag
        __hps_sdram_p0
145 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_rasn -tag
        __hps_sdram_p0
146 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_wen -tag
        __hps_sdram_p0
```

```
147 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_resetn -tag
        __hps_sdram_p0
148 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_dm[0] -tag
        __hps_sdram_p0
149 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_dm[1] -tag
        __hps_sdram_p0
150 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_dm[2] -tag
        __hps_sdram_p0
151 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_dm[3] -tag
        __hps_sdram_p0
152 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_hps_dm[4] -tag
        __hps_sdram_p0
153 set_location_assignment PIN_F26 -to ddr3_hps_a[0]
154 set_location_assignment PIN_G30 -to ddr3_hps_a[1]
155 set_location_assignment PIN_F28 -to ddr3_hps_a[2]
156 set_location_assignment PIN_F30 -to ddr3_hps_a[3]
157 set_location_assignment PIN_J25 -to ddr3_hps_a[4]
158 set_location_assignment PIN_J27 -to ddr3_hps_a[5]
159 set_location_assignment PIN_F29 -to ddr3_hps_a[6]
160 set_location_assignment PIN_E28 -to ddr3_hps_a[7]
161 set_location_assignment PIN_H27 -to ddr3_hps_a[8]
162 set_location_assignment PIN_G26 -to ddr3_hps_a[9]
163 set_location_assignment PIN_D29 -to ddr3_hps_a[10]
164 set_location_assignment PIN_C30 -to ddr3_hps_a[11]
165 set_location_assignment PIN_B30 -to ddr3_hps_a[12]
166 set_location_assignment PIN_C29 -to ddr3_hps_a[13]
167 set_location_assignment PIN_H25 -to ddr3_hps_a[14]
168 set_location_assignment PIN_E29 -to ddr3_hps_ba[0]
169 set_location_assignment PIN_J24 -to ddr3_hps_ba[1]
170 set_location_assignment PIN_J23 -to ddr3_hps_ba[2]
171 set_location_assignment PIN_K28 -to ddr3_hps_dm[0]
172 set_location_assignment PIN_M28 -to ddr3_hps_dm[1]
173 set_location_assignment PIN_R28 -to ddr3_hps_dm[2]
174 set_location_assignment PIN_W30 -to ddr3_hps_dm[3]
175 set_location_assignment PIN_W27 -to ddr3_hps_dm[4]
176 set_location_assignment PIN_K23 -to ddr3_hps_dq[0]
177 set_location_assignment PIN_K22 -to ddr3_hps_dq[1]
178 set_location_assignment PIN_H30 -to ddr3_hps_dq[2]
179 set_location_assignment PIN_G28 -to ddr3_hps_dq[3]
180 set_location_assignment PIN_L25 -to ddr3_hps_dq[4]
181 set_location_assignment PIN_L24 -to ddr3_hps_dq[5]
182 set_location_assignment PIN_J30 -to ddr3_hps_dq[6]
183 set_location_assignment PIN_J29 -to ddr3_hps_dq[7]
184 set_location_assignment PIN_K26 -to ddr3_hps_dq[8]
185 set_location_assignment PIN_L26 -to ddr3_hps_dq[9]
186 set_location_assignment PIN_K29 -to ddr3_hps_dq[10]
187 set_location_assignment PIN_K27 -to ddr3_hps_dq[11]
188 set_location_assignment PIN_M26 -to ddr3_hps_dq[12]
189 set_location_assignment PIN_M27 -to ddr3_hps_dq[13]
190 set_location_assignment PIN_L28 -to ddr3_hps_dq[14]
191 set_location_assignment PIN_M30 -to ddr3_hps_dq[15]
192 set_location_assignment PIN_U26 -to ddr3_hps_dq[16]
193 set_location_assignment PIN_T26 -to ddr3_hps_dq[17]
194 set_location_assignment PIN_N29 -to ddr3_hps_dq[18]
195 set_location_assignment PIN_N28 -to ddr3_hps_dq[19]
196 set_location_assignment PIN_P26 -to ddr3_hps_dq[20]
197 set_location_assignment PIN_P27 -to ddr3_hps_dq[21]
198 set_location_assignment PIN_N27 -to ddr3_hps_dq[22]
199 set_location_assignment PIN_R29 -to ddr3_hps_dq[23]
200 set_location_assignment PIN_P24 -to ddr3_hps_dq[24]
201 set_location_assignment PIN_P25 -to ddr3_hps_dq[25]
202 set_location_assignment PIN_T29 -to ddr3_hps_dq[26]
203 set_location_assignment PIN_T28 -to ddr3_hps_dq[27]
204 set_location_assignment PIN_R27 -to ddr3_hps_dq[28]
205 set_location_assignment PIN_R26 -to ddr3_hps_dq[29]
206 set_location_assignment PIN_V30 -to ddr3_hps_dq[30]
207 set_location_assignment PIN_W29 -to ddr3_hps_dq[31]
208 set_location_assignment PIN_W26 -to ddr3_hps_dq[32]
```

```
set_location_assignment PIN_R24 -to ddr3_hps_dq[33]
set_location_assignment PIN_U27 -to ddr3_hps_dq[34]
set_location_assignment PIN_V28 -to ddr3_hps_dq[35]
set_location_assignment PIN_T25 -to ddr3_hps_dq[36]
set_location_assignment PIN_U25 -to ddr3_hps_dq[37]
set_location_assignment PIN_V27 -to ddr3_hps_dq[38]
set_location_assignment PIN_Y29 -to ddr3_hps_dq[39]
set_location_assignment PIN_M19 -to ddr3_hps_dqs_n[0]
set_location_assignment PIN_N24 -to ddr3_hps_dqs_n[1]
set_location_assignment PIN_R18 -to ddr3_hps_dqs_n[2]
set_location_assignment PIN_R21 -to ddr3_hps_dqs_n[3]
set_location_assignment PIN_T23 -to ddr3_hps_dqs_n[4]
set_location_assignment PIN_N18 -to ddr3_hps_dqs_p[0]
set_location_assignment PIN_N25 -to ddr3_hps_dqs_p[1]
set_location_assignment PIN_R19 -to ddr3_hps_dqs_p[2]
set_location_assignment PIN_R22 -to ddr3_hps_dqs_p[3]
set_location_assignment PIN_T24 -to ddr3_hps_dqs_p[4]
set_location_assignment PIN_E27 -to ddr3_hps_casn
set_location_assignment PIN_L29 -to ddr3_hps_cke
set_location_assignment PIN_L23 -to ddr3_hps_clk_n
set_location_assignment PIN_M23 -to ddr3_hps_clk_p
set_location_assignment PIN_H24 -to ddr3_hps_csn
set_location_assignment PIN_H28 -to ddr3_hps_odt
set_location_assignment PIN_D30 -to ddr3_hps_rasn
set_location_assignment PIN_P30 -to ddr3_hps_resetn
set_location_assignment PIN_C28 -to ddr3_hps_wen
set_location_assignment PIN_D27 -to ddr3_hps_rzq
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to clk_osc2
set_location_assignment PIN_F25 -to clk_osc2
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to mictor_rstn
set_location_assignment PIN_C27 -to mictor_rstn
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to hps_resetn
set_location_assignment PIN_F23 -to hps_resetn
set_instance_assignment -name IO_STANDARD "1.5 V" -to user_dipsw_hps
set_location_assignment PIN_N30 -to user_dipsw_hps[0]
set_location_assignment PIN_P29 -to user_dipsw_hps[1]
set_location_assignment PIN_P22 -to user_dipsw_hps[2]
set_location_assignment PIN_V20 -to user_dipsw_hps[3]
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to user_led_hps
set_location_assignment PIN_E17 -to user_led_hps[0]
set_location_assignment PIN_E18 -to user_led_hps[1]
set_location_assignment PIN_G17 -to user_led_hps[2]
set_location_assignment PIN_C18 -to user_led_hps[3]
set_instance_assignment -name IO_STANDARD "1.5 V" -to user_pb_hps
set_location_assignment PIN_T30 -to user_pb_hps[0]
set_location_assignment PIN_U28 -to user_pb_hps[1]
set_location_assignment PIN_T21 -to user_pb_hps[2]
set_location_assignment PIN_U20 -to user_pb_hps[3]
set_location_assignment PIN_E24 -to uart_rx
set_location_assignment PIN_D24 -to uart_tx
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to uart_rx
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to uart_tx
set_location_assignment PIN_D22 -to i2c_scl_hps
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to i2c_scl_hps
set_location_assignment PIN_C23 -to i2c_sda_hps
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to i2c_sda_hps
set_location_assignment PIN_H20 -to spi_csn
set_location_assignment PIN_B23 -to spi_miso
set_location_assignment PIN_C22 -to spi_mosi
set_location_assignment PIN_A23 -to spi_sck
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to spi_csn
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to spi_miso
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to spi_mosi
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to spi_sck
set_location_assignment PIN_D19 -to qspi_clk
set_location_assignment PIN_C20 -to qspi_io[0]
set_location_assignment PIN_H18 -to qspi_io[1]
set_location_assignment PIN_A19 -to qspi_io[2]
```

```
set_location_assignment PIN_E19 -to qspi_io[3]
set_location_assignment PIN_A18 -to qspi_ss0
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to qspi_clk
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to qspi_io
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to qspi_ss0
set_location_assignment PIN_B16 -to sd_dat[3]
set_location_assignment PIN_A16 -to sd_clk
set_location_assignment PIN_F18 -to sd_cmd
set_location_assignment PIN_G18 -to sd_dat[0]
set_location_assignment PIN_C17 -to sd_dat[1]
set_location_assignment PIN_D17 -to sd_dat[2]
set_location_assignment PIN_B17 -to sd_pwren
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to sd_clk
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to sd_cmd
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to sd_pwren
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to sd_dat
set_location_assignment PIN_N16 -to usb_clk
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to usb_clk
set_location_assignment PIN_E16 -to usb_data[0]
set_location_assignment PIN_G16 -to usb_data[1]
set_location_assignment PIN_D16 -to usb_data[2]
set_location_assignment PIN_D14 -to usb_data[3]
set_location_assignment PIN_A15 -to usb_data[4]
set_location_assignment PIN_C14 -to usb_data[5]
set_location_assignment PIN_D15 -to usb_data[6]
set_location_assignment PIN_M17 -to usb_data[7]
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to usb_data
set_location_assignment PIN_A14 -to usb_nxt
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to usb_nxt
set_location_assignment PIN_E14 -to usb_dir
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to usb_stp
set_location_assignment PIN_C15 -to usb_stp
set_location_assignment PIN_H19 -to enet_hps_gtx_clk
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to enet_hps_gtx_clk
set_location_assignment PIN_B21 -to enet_hps_mdc
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to enet_hps_mdc
set_location_assignment PIN_E21 -to enet_hps_mdio
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to enet_hps_mdio
set_location_assignment PIN_G20 -to enet_hps_rx_clk
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to enet_hps_rx_clk
set_location_assignment PIN_K17 -to enet_hps_rx_dv
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to enet_hps_rx_dv
set_location_assignment PIN_A21 -to enet_hps_rxd[0]
set_location_assignment PIN_B20 -to enet_hps_rxd[1]
set_location_assignment PIN_B18 -to enet_hps_rxd[2]
set_location_assignment PIN_D21 -to enet_hps_rxd[3]
set_location_assignment PIN_A20 -to enet_hps_tx_en
set_location_assignment PIN_F20 -to enet_hps_txd[0]
set_location_assignment PIN_J19 -to enet_hps_txd[1]
set_location_assignment PIN_F21 -to enet_hps_txd[2]
set_location_assignment PIN_F19 -to enet_hps_txd[3]
set_instance_assignment -name IO_STANDARD "2.5 V" -to enet_hps_rxd
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to enet_hps_tx_en
set_instance_assignment -name IO_STANDARD "2.5 V" -to enet_hps_txd
set_location_assignment PIN_C19 -to enet_hps_intn
set_location_assignment PIN_B22 -to can_0_rx
set_location_assignment PIN_G22 -to can_0_tx
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to can_0_rx
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to can_0_tx
set_location_assignment PIN_B26 -to trace_clk_mic
set_location_assignment PIN_B25 -to trace_data[0]
set_location_assignment PIN_C25 -to trace_data[1]
set_location_assignment PIN_A25 -to trace_data[2]
set_location_assignment PIN_H23 -to trace_data[3]
set_location_assignment PIN_A24 -to trace_data[4]
set_location_assignment PIN_G21 -to trace_data[5]
set_location_assignment PIN_C24 -to trace_data[6]
set_location_assignment PIN_E23 -to trace_data[7]
```

```
345 set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to trace_clk_mic
346 set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to trace_data
347 set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to usb_dir
348 set_location_assignment PIN_AB27 -to clk_100m_fpga
349 set_instance_assignment -name IO_STANDARD "2.5 V" -to clk_100m_fpga
350 set_location_assignment PIN_AC18 -to clk_50m_fpga
351 set_instance_assignment -name IO_STANDARD "1.5 V" -to clk_50m_fpga
352 set_location_assignment PIN_AF14 -to clk_bot1
353 set_instance_assignment -name IO_STANDARD "1.5 V" -to clk_bot1
354 set_location_assignment PIN_Y26 -to clk_enet_fpga_p
355 set_instance_assignment -name IO_STANDARD LVDS -to clk_enet_fpga_p
356 set_location_assignment PIN_AA26 -to clk_top1
357 set_instance_assignment -name IO_STANDARD "2.5 V" -to clk_top1
358 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_a
359 set_location_assignment PIN_AJ14 -to ddr3_fpga_a[0]
360 set_location_assignment PIN_AK14 -to ddr3_fpga_a[1]
361 set_location_assignment PIN_AH12 -to ddr3_fpga_a[2]
362 set_location_assignment PIN_AJ12 -to ddr3_fpga_a[3]
363 set_location_assignment PIN_AG15 -to ddr3_fpga_a[4]
364 set_location_assignment PIN_AH15 -to ddr3_fpga_a[5]
365 set_location_assignment PIN_AK12 -to ddr3_fpga_a[6]
366 set_location_assignment PIN_AK13 -to ddr3_fpga_a[7]
367 set_location_assignment PIN_AH13 -to ddr3_fpga_a[8]
368 set_location_assignment PIN_AH14 -to ddr3_fpga_a[9]
369 set_location_assignment PIN_AJ9 -to ddr3_fpga_a[10]
370 set_location_assignment PIN_AK9 -to ddr3_fpga_a[11]
371 set_location_assignment PIN_AK7 -to ddr3_fpga_a[12]
372 set_location_assignment PIN_AK8 -to ddr3_fpga_a[13]
373 set_location_assignment PIN_AG12 -to ddr3_fpga_a[14]
374 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_ba
375 set_location_assignment PIN_AH10 -to ddr3_fpga_ba[0]
376 set_location_assignment PIN_AJ11 -to ddr3_fpga_ba[1]
377 set_location_assignment PIN_AK11 -to ddr3_fpga_ba[2]
378 set_location_assignment PIN_AH7 -to ddr3_fpga_casn
379 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_casn -tag
      __orion_system_mem_if_ddr3_fpga_p0
380 set_location_assignment PIN_AJ21 -to ddr3_fpga_cke
381 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_cke -tag
      __orion_system_mem_if_ddr3_fpga_p0
382 set_location_assignment PIN_AA15 -to ddr3_fpga_clk_n
383 set_instance_assignment -name IO_STANDARD "DIFFERENTIAL 1.5-V SSTL CLASS I" -to
      ddr3_fpga_clk_n -tag __orion_system_mem_if_ddr3_fpga_p0
384 set_location_assignment PIN_AA14 -to ddr3_fpga_clk_p
385 set_instance_assignment -name IO_STANDARD "DIFFERENTIAL 1.5-V SSTL CLASS I" -to
      ddr3_fpga_clk_p -tag __orion_system_mem_if_ddr3_fpga_p0
386 set_location_assignment PIN_AB15 -to ddr3_fpga_csn
387 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_csn -tag
      __orion_system_mem_if_ddr3_fpga_p0
388 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_dm
389 set_location_assignment PIN_AH17 -to ddr3_fpga_dm[0]
390 set_location_assignment PIN_AG23 -to ddr3_fpga_dm[1]
391 set_location_assignment PIN_AK23 -to ddr3_fpga_dm[2]
392 set_location_assignment PIN_AJ27 -to ddr3_fpga_dm[3]
393 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_dq
394 set_location_assignment PIN_AF18 -to ddr3_fpga_dq[0]
395 set_location_assignment PIN_AE17 -to ddr3_fpga_dq[1]
396 set_location_assignment PIN_AG16 -to ddr3_fpga_dq[2]
397 set_location_assignment PIN_AF16 -to ddr3_fpga_dq[3]
398 set_location_assignment PIN_AH20 -to ddr3_fpga_dq[4]
399 set_location_assignment PIN_AG21 -to ddr3_fpga_dq[5]
400 set_location_assignment PIN_AJ16 -to ddr3_fpga_dq[6]
401 set_location_assignment PIN_AH18 -to ddr3_fpga_dq[7]
402 set_location_assignment PIN_AK18 -to ddr3_fpga_dq[8]
403 set_location_assignment PIN_AJ17 -to ddr3_fpga_dq[9]
404 set_location_assignment PIN_AG18 -to ddr3_fpga_dq[10]
405 set_location_assignment PIN_AK19 -to ddr3_fpga_dq[11]
406 set_location_assignment PIN_AG20 -to ddr3_fpga_dq[12]
407 set_location_assignment PIN_AF19 -to ddr3_fpga_dq[13]
```

```
set_location_assignment PIN_AJ20 -to ddr3_fpga_dq[14]
set_location_assignment PIN_AH24 -to ddr3_fpga_dq[15]
set_location_assignment PIN_AE19 -to ddr3_fpga_dq[16]
set_location_assignment PIN_AE18 -to ddr3_fpga_dq[17]
set_location_assignment PIN_AG22 -to ddr3_fpga_dq[18]
set_location_assignment PIN_AK22 -to ddr3_fpga_dq[19]
set_location_assignment PIN_AF21 -to ddr3_fpga_dq[20]
set_location_assignment PIN_AF20 -to ddr3_fpga_dq[21]
set_location_assignment PIN_AH23 -to ddr3_fpga_dq[22]
set_location_assignment PIN_AK24 -to ddr3_fpga_dq[23]
set_location_assignment PIN_AF24 -to ddr3_fpga_dq[24]
set_location_assignment PIN_AF23 -to ddr3_fpga_dq[25]
set_location_assignment PIN_AJ24 -to ddr3_fpga_dq[26]
set_location_assignment PIN_AK26 -to ddr3_fpga_dq[27]
set_location_assignment PIN_AE23 -to ddr3_fpga_dq[28]
set_location_assignment PIN_AE22 -to ddr3_fpga_dq[29]
set_location_assignment PIN_AG25 -to ddr3_fpga_dq[30]
set_location_assignment PIN_AK27 -to ddr3_fpga_dq[31]
set_instance_assignment -name IO_STANDARD "DIFFERENTIAL 1.5-V SSTL CLASS I" -to
    ddr3_fpga_dqs_n
set_instance_assignment -name IO_STANDARD "DIFFERENTIAL 1.5-V SSTL CLASS I" -to
    ddr3_fpga_dqs_p
set_location_assignment PIN_AE16 -to ddr3_fpga_odt
set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_odt -tag
    __orion_system_mem_if_ddr3_fpga_p0
set_location_assignment PIN_AH8 -to ddr3_fpga_rasn
set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_rasn
set_location_assignment PIN_AK21 -to ddr3_fpga_resetn
set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_resetn
set_location_assignment PIN_AJ6 -to ddr3_fpga_wen
set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_wen
set_location_assignment PIN_C12 -to hsma_tx_d_p[16]
set_location_assignment PIN_B13 -to hsma_tx_d_p[15]
set_location_assignment PIN_C10 -to hsma_tx_d_p[14]
set_location_assignment PIN_C8 -to hsma_tx_d_p[13]
set_location_assignment PIN_A9 -to hsma_tx_d_p[12]
set_location_assignment PIN_C7 -to hsma_tx_d_p[11]
set_location_assignment PIN_A6 -to hsma_tx_d_p[10]
set_location_assignment PIN_D5 -to hsma_tx_d_p[9]
set_location_assignment PIN_A4 -to hsma_tx_d_p[8]
set_location_assignment PIN_C3 -to hsma_tx_d_p[7]
set_location_assignment PIN_B2 -to hsma_tx_d_p[6]
set_location_assignment PIN_D2 -to hsma_tx_d_p[5]
set_location_assignment PIN_E1 -to hsma_tx_d_p[4]
set_location_assignment PIN_E3 -to hsma_tx_d_p[3]
set_location_assignment PIN_E4 -to hsma_tx_d_p[2]
set_location_assignment PIN_D6 -to hsma_tx_d_p[1]
set_location_assignment PIN_E8 -to hsma_tx_d_p[0]
set_instance_assignment -name IO_STANDARD LVDS -to hsma_tx_d_p
set_location_assignment PIN_B11 -to hsma_tx_d_n[16]
set_location_assignment PIN_A13 -to hsma_tx_d_n[15]
set_location_assignment PIN_C9 -to hsma_tx_d_n[14]
set_location_assignment PIN_B8 -to hsma_tx_d_n[13]
set_location_assignment PIN_A8 -to hsma_tx_d_n[12]
set_location_assignment PIN_B7 -to hsma_tx_d_n[11]
set_location_assignment PIN_A5 -to hsma_tx_d_n[10]
set_location_assignment PIN_C4 -to hsma_tx_d_n[9]
set_location_assignment PIN_A3 -to hsma_tx_d_n[8]
set_location_assignment PIN_B3 -to hsma_tx_d_n[7]
set_location_assignment PIN_B1 -to hsma_tx_d_n[6]
set_location_assignment PIN_C2 -to hsma_tx_d_n[5]
set_location_assignment PIN_D1 -to hsma_tx_d_n[4]
set_location_assignment PIN_E2 -to hsma_tx_d_n[3]
set_location_assignment PIN_D4 -to hsma_tx_d_n[2]
set_location_assignment PIN_C5 -to hsma_tx_d_n[1]
set_location_assignment PIN_D7 -to hsma_tx_d_n[0]
set_location_assignment PIN_AH2 -to hsma_sda
set_location_assignment PIN_AA12 -to hsma_scl
```

```
set_instance_assignment -name IO_STANDARD "2.5 V" -to hsma_scl
set_location_assignment PIN_J2 -to hsma_rx_p[3]
set_location_assignment PIN_L2 -to hsma_rx_p[2]
set_location_assignment PIN_N2 -to hsma_rx_p[1]
set_location_assignment PIN_R2 -to hsma_rx_p[0]
set_location_assignment PIN_J1 -to hsma_rx_n[3]
set_location_assignment PIN_L1 -to hsma_rx_n[2]
set_location_assignment PIN_N1 -to hsma_rx_n[1]
set_location_assignment PIN_R1 -to hsma_rx_n[0]
set_location_assignment PIN_F15 -to hsma_rx_d_p[16]
set_location_assignment PIN_C13 -to hsma_rx_d_p[15]
set_location_assignment PIN_F13 -to hsma_rx_d_p[14]
set_location_assignment PIN_E12 -to hsma_rx_d_p[13]
set_location_assignment PIN_D11 -to hsma_rx_d_p[12]
set_location_assignment PIN_E9 -to hsma_rx_d_p[11]
set_location_assignment PIN_B6 -to hsma_rx_d_p[10]
set_location_assignment PIN_F11 -to hsma_rx_d_p[9]
set_location_assignment PIN_F9 -to hsma_rx_d_p[8]
set_location_assignment PIN_G10 -to hsma_rx_d_p[7]
set_location_assignment PIN_H8 -to hsma_rx_d_p[6]
set_location_assignment PIN_J7 -to hsma_rx_d_p[5]
set_location_assignment PIN_G12 -to hsma_rx_d_p[4]
set_location_assignment PIN_K7 -to hsma_rx_d_p[3]
set_location_assignment PIN_J10 -to hsma_rx_d_p[2]
set_location_assignment PIN_K12 -to hsma_rx_d_p[1]
set_location_assignment PIN_H14 -to hsma_rx_d_p[0]
set_instance_assignment -name IO_STANDARD LVDS -to hsma_rx_d_p
set_location_assignment PIN_F14 -to hsma_rx_d_n[16]
set_location_assignment PIN_B12 -to hsma_rx_d_n[15]
set_location_assignment PIN_E13 -to hsma_rx_d_n[14]
set_location_assignment PIN_D12 -to hsma_rx_d_n[13]
set_location_assignment PIN_D10 -to hsma_rx_d_n[12]
set_location_assignment PIN_D9 -to hsma_rx_d_n[11]
set_location_assignment PIN_B5 -to hsma_rx_d_n[10]
set_location_assignment PIN_E11 -to hsma_rx_d_n[9]
set_location_assignment PIN_F8 -to hsma_rx_d_n[8]
set_location_assignment PIN_F10 -to hsma_rx_d_n[7]
set_location_assignment PIN_G8 -to hsma_rx_d_n[6]
set_location_assignment PIN_H7 -to hsma_rx_d_n[5]
set_location_assignment PIN_G11 -to hsma_rx_d_n[4]
set_location_assignment PIN_K8 -to hsma_rx_d_n[3]
set_location_assignment PIN_J9 -to hsma_rx_d_n[2]
set_location_assignment PIN_J12 -to hsma_rx_d_n[1]
set_location_assignment PIN_G13 -to hsma_rx_d_n[0]
set_location_assignment PIN_AD12 -to hsma_prsntn
set_instance_assignment -name IO_STANDARD "2.5 V" -to hsma_prsntn
set_location_assignment PIN_AG1 -to hsma_d[3]
set_location_assignment PIN_AG7 -to hsma_d[2]
set_location_assignment PIN_AF8 -to hsma_d[1]
set_location_assignment PIN_AF9 -to hsma_d[0]
set_instance_assignment -name IO_STANDARD "2.5 V" -to hsma_d
set_location_assignment PIN_A10 -to hsma_clk_out0
set_instance_assignment -name IO_STANDARD "2.5 V" -to hsma_clk_out0
set_location_assignment PIN_E7 -to hsma_clk_out_p2
set_instance_assignment -name IO_STANDARD LVDS -to hsma_clk_out_p2
set_location_assignment PIN_E6 -to hsma_clk_out_n2
set_location_assignment PIN_K14 -to hsma_clk_in0
set_instance_assignment -name IO_STANDARD "2.5 V" -to hsma_clk_in0
set_location_assignment PIN_H15 -to hsma_clk_in_p2
set_instance_assignment -name IO_STANDARD LVDS -to hsma_clk_in_p2
set_location_assignment PIN_G15 -to hsma_clk_in_n2
set_location_assignment PIN_AB22 -to enet1_tx_en
set_location_assignment PIN_AB26 -to enet1_tx_d[3]
set_location_assignment PIN_AA25 -to enet1_tx_d[2]
set_location_assignment PIN_Y21 -to enet1_tx_d[1]
set_location_assignment PIN_W20 -to enet1_tx_d[0]
set_location_assignment PIN_W25 -to enet1_tx_clk_fb
set_location_assignment PIN_AE28 -to enet1_rx_error
```

```
set_location_assignment PIN_Y23 -to enet1_rx_dv
set_location_assignment PIN_AE27 -to enet1_rx_d[3]
set_location_assignment PIN_AB25 -to enet1_rx_d[2]
set_location_assignment PIN_AA24 -to enet1_rx_d[1]
set_location_assignment PIN_AB23 -to enet1_rx_d[0]
set_location_assignment PIN_Y24 -to enet1_rx_clk
set_location_assignment PIN_W24 -to enet2_tx_en
set_location_assignment PIN_V23 -to enet2_tx_d[3]
set_location_assignment PIN_AF28 -to enet2_tx_d[2]
set_location_assignment PIN_AG28 -to enet2_tx_d[1]
set_location_assignment PIN_AG27 -to enet2_tx_d[0]
set_location_assignment PIN_AG30 -to enet2_tx_clk_fb
set_location_assignment PIN_V25 -to enet2_rx_error
set_location_assignment PIN_AC28 -to enet2_rx_dv
set_location_assignment PIN_AC27 -to enet2_rx_d[3]
set_location_assignment PIN_AD26 -to enet2_rx_d[2]
set_location_assignment PIN_AF30 -to enet2_rx_d[1]
set_location_assignment PIN_AF29 -to enet2_rx_d[0]
set_location_assignment PIN_AH30 -to enet2_rx_clk
set_location_assignment PIN_AC29 -to sdi_tx_sd_hdn
set_location_assignment PIN_AA30 -to sdi_tx_en
set_location_assignment PIN_AA28 -to sdi_rx_en
set_location_assignment PIN_AB28 -to sdi_rx_bypass
set_location_assignment PIN_AB30 -to sdi_rsti
set_location_assignment PIN_AD30 -to sdi_fault
set_location_assignment PIN_AB12 -to sdi_clk148_up
set_location_assignment PIN_AF6 -to sdi_clk148_dn
set_location_assignment PIN_AG10 -to user_dipsw_fpga[0]
set_location_assignment PIN_AH9 -to user_dipsw_fpga[1]
set_location_assignment PIN_AF11 -to user_dipsw_fpga[2]
set_location_assignment PIN_AG11 -to user_dipsw_fpga[3]
set_instance_assignment -name IO_STANDARD "1.5 V" -to user_dipsw_fpga[0]
set_instance_assignment -name IO_STANDARD "1.5 V" -to user_dipsw_fpga[1]
set_instance_assignment -name IO_STANDARD "1.5 V" -to user_dipsw_fpga[2]
set_instance_assignment -name IO_STANDARD "1.5 V" -to user_dipsw_fpga[3]
set_location_assignment PIN_AK2 -to user_led_fpga[0]
set_location_assignment PIN_Y16 -to user_led_fpga[1]
set_location_assignment PIN_W15 -to user_led_fpga[2]
set_location_assignment PIN_AB17 -to user_led_fpga[3]
set_instance_assignment -name IO_STANDARD "1.5 V" -to user_led_fpga[0]
set_instance_assignment -name IO_STANDARD "1.5 V" -to user_led_fpga[1]
set_instance_assignment -name IO_STANDARD "1.5 V" -to user_led_fpga[2]
set_instance_assignment -name IO_STANDARD "1.5 V" -to user_led_fpga[3]
set_location_assignment PIN_AA13 -to user_pb_fpga[0]
set_location_assignment PIN_AB13 -to user_pb_fpga[1]
set_instance_assignment -name IO_STANDARD "1.5 V" -to user_pb_fpga[0]
set_instance_assignment -name IO_STANDARD "1.5 V" -to user_pb_fpga[1]
set_location_assignment PIN_AG17 -to ddr3_fpga_rzq
set_instance_assignment -name IO_STANDARD "SSTL-15" -to ddr3_fpga_rzq -tag
    __orion_system_mem_if_ddr3_fpga_p0
#---Found multiple nets with the same name
set_location_assignment PIN_W21 -to pcie_perstn_in
set_location_assignment PIN_AG6 -to pcie_perstn_out
set_location_assignment PIN_W7 -to pcie_refclk_n
set_location_assignment PIN_W8 -to pcie_refclk_p
set_location_assignment PIN_AE1 -to pcie_rx_n[0]
set_location_assignment PIN_AC1 -to pcie_rx_n[1]
set_location_assignment PIN_AA1 -to pcie_rx_n[2]
set_location_assignment PIN_W1 -to pcie_rx_n[3]
set_location_assignment PIN_AE2 -to pcie_rx_p[0]
set_location_assignment PIN_AC2 -to pcie_rx_p[1]
set_location_assignment PIN_AA2 -to pcie_rx_p[2]
set_location_assignment PIN_W2 -to pcie_rx_p[3]
set_location_assignment PIN_AD3 -to pcie_tx_n[0]
set_location_assignment PIN_AB3 -to pcie_tx_n[1]
set_location_assignment PIN_Y3 -to pcie_tx_n[2]
set_location_assignment PIN_V3 -to pcie_tx_n[3]
set_location_assignment PIN_AD4 -to pcie_tx_p[0]
```

```
608 set_location_assignment PIN_AB4 -to pcie_tx_p[1]
609 set_location_assignment PIN_Y4 -to pcie_tx_p[2]
610 set_location_assignment PIN_V4 -to pcie_tx_p[3]
611 set_location_assignment PIN_AF13 -to usb_b2_clk
612 set_location_assignment PIN_AK28 -to usb_b2_data[0]
613 set_location_assignment PIN_AD20 -to usb_b2_data[1]
614 set_location_assignment PIN_AD21 -to usb_b2_data[2]
615 set_location_assignment PIN_Y19 -to usb_b2_data[3]
616 set_location_assignment PIN_AA20 -to usb_b2_data[4]
617 set_location_assignment PIN_AH27 -to usb_b2_data[5]
618 set_location_assignment PIN_AF25 -to usb_b2_data[6]
619 set_location_assignment PIN_AC22 -to usb_b2_data[7]
620 set_instance_assignment -name IO_STANDARD "1.5 V" -to usb_b2_clk
621 set_instance_assignment -name IO_STANDARD "1.5 V" -to usb_b2_data
622 set_location_assignment PIN_AD27 -to cpu_resetn
623 set_location_assignment PIN_AG5 -to enet1_tx_error
624 set_location_assignment PIN_AH5 -to enet2_tx_error
625 set_location_assignment PIN_AJ1 -to enet_dual_resetn
626 set_instance_assignment -name IO_STANDARD "1.5-V PCML" -to pcie_tx_p
627 set_instance_assignment -name IO_STANDARD "1.5-V PCML" -to pcie_rx_p
628 set_instance_assignment -name IO_STANDARD "1.5-V PCML" -to hsma_tx_p
629 set_instance_assignment -name IO_STANDARD "1.5-V PCML" -to hsma_rx_p
630 set_instance_assignment -name IO_STANDARD "1.5-V PCML" -to gxb_rx_l4_p
631 set_instance_assignment -name IO_STANDARD "1.5-V PCML" -to gxb_tx_l4_p
632 set_instance_assignment -name IO_STANDARD LVDS -to clk_148_p
633 set_instance_assignment -name IO_STANDARD LVDS -to pcie_refclk_p
634 set_location_assignment PIN_T9 -to clk_148_p
635 set_global_assignment -name ENABLE_SIGNALTAP OFF
636 set_global_assignment -name USE_SIGNALTAP_FILE output_files_sof_rpt/stp1.stp
637 set_global_assignment -name MIN_CORE_JUNCTION_TEMP 0
638 set_global_assignment -name MAX_CORE_JUNCTION_TEMP 85
639 set_location_assignment PIN_G7 -to i2c_scl_fpga
640 set_location_assignment PIN_F6 -to i2c_sda_fpga
641 set_location_assignment PIN_AJ4 -to usb_empty
642 set_location_assignment PIN_AK3 -to usb_full
643 set_location_assignment PIN_AE14 -to usb_oen
644 set_location_assignment PIN_AJ5 -to usb_rdn
645 set_location_assignment PIN_AD14 -to usb_resetn
646 set_location_assignment PIN_AK4 -to usb_scl
647 set_location_assignment PIN_AE13 -to usb_sda
648 set_location_assignment PIN_AK6 -to usb_wrn
649 set_location_assignment PIN_H12 -to enet_fpga_mdc
650 set_location_assignment PIN_H13 -to enet_fpga_mdio
651 set_location_assignment PIN_T8 -to clk_148_n
652 set_location_assignment PIN_Y27 -to clk_enet_fpga_n
653 set_location_assignment PIN_U1 -to gxb_rx_l4_n
654 set_location_assignment PIN_U2 -to gxb_rx_l4_p
655 set_location_assignment PIN_T3 -to gxb_tx_l4_n
656 set_location_assignment PIN_T4 -to gxb_tx_l4_p
657 set_location_assignment PIN_P8 -to refclk_ql2_n
658 set_location_assignment PIN_P9 -to refclk_ql2_p
659 set_location_assignment PIN_AE29 -to pcie_smbclk
660 set_location_assignment PIN_J14 -to pcie_smbdat
661 set_location_assignment PIN_W22 -to pcie_waken
662 set_location_assignment PIN_AD25 -to max_fpga_miso
663 set_location_assignment PIN_AE26 -to max_fpga_mosi
664 set_location_assignment PIN_AJ29 -to max_fpga_sck
665 set_location_assignment PIN_AC25 -to max_fpga_ssel
666 set_location_assignment PIN_AD29 -to pcie_prsnt2_x1
667 set_location_assignment PIN_A11 -to pcie_prsnt2_x4
668 set_instance_assignment -name IO_STANDARD LVDS -to refclk_ql2_p
669 set_global_assignment -name PARTITION_NETLIST_TYPE SOURCE -section_id Top
670 set_global_assignment -name PARTITION_FITTER_PRESERVATION_LEVEL PLACEMENT_AND_ROUTING
        -section_id Top
671 set_global_assignment -name PARTITION_COLOR 16764057 -section_id Top
672 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[0] -tag __hps_sdram_p0
```

```
673 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[0] -tag __hps_sdram_p0
674 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[1] -tag __hps_sdram_p0
675 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[1] -tag __hps_sdram_p0
676 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[2] -tag __hps_sdram_p0
677 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[2] -tag __hps_sdram_p0
678 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[3] -tag __hps_sdram_p0
679 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[3] -tag __hps_sdram_p0
680 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[4] -tag __hps_sdram_p0
681 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[4] -tag __hps_sdram_p0
682 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[5] -tag __hps_sdram_p0
683 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[5] -tag __hps_sdram_p0
684 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[6] -tag __hps_sdram_p0
685 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[6] -tag __hps_sdram_p0
686 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[7] -tag __hps_sdram_p0
687 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[7] -tag __hps_sdram_p0
688 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[8] -tag __hps_sdram_p0
689 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[8] -tag __hps_sdram_p0
690 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[9] -tag __hps_sdram_p0
691 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[9] -tag __hps_sdram_p0
692 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[10] -tag __hps_sdram_p0
693 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[10] -tag __hps_sdram_p0
694 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[11] -tag __hps_sdram_p0
695 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[11] -tag __hps_sdram_p0
696 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[12] -tag __hps_sdram_p0
697 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[12] -tag __hps_sdram_p0
698 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[13] -tag __hps_sdram_p0
699 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[13] -tag __hps_sdram_p0
700 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[14] -tag __hps_sdram_p0
701 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[14] -tag __hps_sdram_p0
702 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[15] -tag __hps_sdram_p0
703 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[15] -tag __hps_sdram_p0
704 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[16] -tag __hps_sdram_p0
705 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[16] -tag __hps_sdram_p0
706 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[17] -tag __hps_sdram_p0
```

```
707 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[17] -tag __hps_sdram_p0
708 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[18] -tag __hps_sdram_p0
709 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[18] -tag __hps_sdram_p0
710 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[19] -tag __hps_sdram_p0
711 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[19] -tag __hps_sdram_p0
712 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[20] -tag __hps_sdram_p0
713 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[20] -tag __hps_sdram_p0
714 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[21] -tag __hps_sdram_p0
715 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[21] -tag __hps_sdram_p0
716 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[22] -tag __hps_sdram_p0
717 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[22] -tag __hps_sdram_p0
718 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[23] -tag __hps_sdram_p0
719 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[23] -tag __hps_sdram_p0
720 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[24] -tag __hps_sdram_p0
721 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[24] -tag __hps_sdram_p0
722 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[25] -tag __hps_sdram_p0
723 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[25] -tag __hps_sdram_p0
724 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[26] -tag __hps_sdram_p0
725 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[26] -tag __hps_sdram_p0
726 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[27] -tag __hps_sdram_p0
727 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[27] -tag __hps_sdram_p0
728 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[28] -tag __hps_sdram_p0
729 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[28] -tag __hps_sdram_p0
730 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[29] -tag __hps_sdram_p0
731 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[29] -tag __hps_sdram_p0
732 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[30] -tag __hps_sdram_p0
733 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[30] -tag __hps_sdram_p0
734 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[31] -tag __hps_sdram_p0
735 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[31] -tag __hps_sdram_p0
736 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[32] -tag __hps_sdram_p0
737 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[32] -tag __hps_sdram_p0
738 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[33] -tag __hps_sdram_p0
739 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[33] -tag __hps_sdram_p0
740 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[34] -tag __hps_sdram_p0
```

```
741 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[34] -tag __hps_sdram_p0
742 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[35] -tag __hps_sdram_p0
743 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[35] -tag __hps_sdram_p0
744 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[36] -tag __hps_sdram_p0
745 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[36] -tag __hps_sdram_p0
746 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[37] -tag __hps_sdram_p0
747 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[37] -tag __hps_sdram_p0
748 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[38] -tag __hps_sdram_p0
749 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[38] -tag __hps_sdram_p0
750 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[39] -tag __hps_sdram_p0
751 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dq[39] -tag __hps_sdram_p0
752 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dqs_p[0] -tag __hps_sdram_p0
753 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dqs_p[0] -tag __hps_sdram_p0
754 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dqs_p[1] -tag __hps_sdram_p0
755 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dqs_p[1] -tag __hps_sdram_p0
756 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dqs_p[2] -tag __hps_sdram_p0
757 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dqs_p[2] -tag __hps_sdram_p0
758 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dqs_p[3] -tag __hps_sdram_p0
759 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dqs_p[3] -tag __hps_sdram_p0
760 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dqs_p[4] -tag __hps_sdram_p0
761 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dqs_p[4] -tag __hps_sdram_p0
762 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dqs_n[0] -tag __hps_sdram_p0
763 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dqs_n[0] -tag __hps_sdram_p0
764 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dqs_n[1] -tag __hps_sdram_p0
765 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dqs_n[1] -tag __hps_sdram_p0
766 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dqs_n[2] -tag __hps_sdram_p0
767 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dqs_n[2] -tag __hps_sdram_p0
768 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dqs_n[3] -tag __hps_sdram_p0
769 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dqs_n[3] -tag __hps_sdram_p0
770 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dqs_n[4] -tag __hps_sdram_p0
771 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dqs_n[4] -tag __hps_sdram_p0
772 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITHOUT CALIBRATION" -to
        ddr3_hps_clk_p -tag __hps_sdram_p0
773 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITHOUT CALIBRATION" -to
        ddr3_hps_clk_n -tag __hps_sdram_p0
774 set_instance_assignment -name CURRENT_STRENGTH_NEW "MAXIMUM CURRENT" -to ddr3_hps_a[0]
        -tag __hps_sdram_p0
```

```
775 set_instance_assignment -name CURRENT_STRENGTH_NEW "MAXIMUM CURRENT" -to ddr3_hps_a[10]
        -tag __hps_sdram_p0
776 set_instance_assignment -name CURRENT_STRENGTH_NEW "MAXIMUM CURRENT" -to ddr3_hps_a[11]
        -tag __hps_sdram_p0
777 set_instance_assignment -name CURRENT_STRENGTH_NEW "MAXIMUM CURRENT" -to ddr3_hps_a[12]
        -tag __hps_sdram_p0
778 set_instance_assignment -name CURRENT_STRENGTH_NEW "MAXIMUM CURRENT" -to ddr3_hps_a[13]
        -tag __hps_sdram_p0
779 set_instance_assignment -name CURRENT_STRENGTH_NEW "MAXIMUM CURRENT" -to ddr3_hps_a[14]
        -tag __hps_sdram_p0
780 set_instance_assignment -name CURRENT_STRENGTH_NEW "MAXIMUM CURRENT" -to ddr3_hps_a[1]
        -tag __hps_sdram_p0
781 set_instance_assignment -name CURRENT_STRENGTH_NEW "MAXIMUM CURRENT" -to ddr3_hps_a[2]
        -tag __hps_sdram_p0
782 set_instance_assignment -name CURRENT_STRENGTH_NEW "MAXIMUM CURRENT" -to ddr3_hps_a[3]
        -tag __hps_sdram_p0
783 set_instance_assignment -name CURRENT_STRENGTH_NEW "MAXIMUM CURRENT" -to ddr3_hps_a[4]
        -tag __hps_sdram_p0
784 set_instance_assignment -name CURRENT_STRENGTH_NEW "MAXIMUM CURRENT" -to ddr3_hps_a[5]
        -tag __hps_sdram_p0
785 set_instance_assignment -name CURRENT_STRENGTH_NEW "MAXIMUM CURRENT" -to ddr3_hps_a[6]
        -tag __hps_sdram_p0
786 set_instance_assignment -name CURRENT_STRENGTH_NEW "MAXIMUM CURRENT" -to ddr3_hps_a[7]
        -tag __hps_sdram_p0
787 set_instance_assignment -name CURRENT_STRENGTH_NEW "MAXIMUM CURRENT" -to ddr3_hps_a[8]
        -tag __hps_sdram_p0
788 set_instance_assignment -name CURRENT_STRENGTH_NEW "MAXIMUM CURRENT" -to ddr3_hps_a[9]
        -tag __hps_sdram_p0
789 set_instance_assignment -name CURRENT_STRENGTH_NEW "MAXIMUM CURRENT" -to ddr3_hps_ba[0]
        -tag __hps_sdram_p0
790 set_instance_assignment -name CURRENT_STRENGTH_NEW "MAXIMUM CURRENT" -to ddr3_hps_ba[1]
        -tag __hps_sdram_p0
791 set_instance_assignment -name CURRENT_STRENGTH_NEW "MAXIMUM CURRENT" -to ddr3_hps_ba[2]
        -tag __hps_sdram_p0
792 set_instance_assignment -name CURRENT_STRENGTH_NEW "MAXIMUM CURRENT" -to ddr3_hps_casn
        -tag __hps_sdram_p0
793 set_instance_assignment -name CURRENT_STRENGTH_NEW "MAXIMUM CURRENT" -to ddr3_hps_cke -tag
        __hps_sdram_p0
794 set_instance_assignment -name CURRENT_STRENGTH_NEW "MAXIMUM CURRENT" -to ddr3_hps_csn -tag
        __hps_sdram_p0
795 set_instance_assignment -name CURRENT_STRENGTH_NEW "MAXIMUM CURRENT" -to ddr3_hps_odt -tag
        __hps_sdram_p0
796 set_instance_assignment -name CURRENT_STRENGTH_NEW "MAXIMUM CURRENT" -to ddr3_hps_rasn
        -tag __hps_sdram_p0
797 set_instance_assignment -name CURRENT_STRENGTH_NEW "MAXIMUM CURRENT" -to ddr3_hps_wen -tag
        __hps_sdram_p0
798 set_instance_assignment -name CURRENT_STRENGTH_NEW "MAXIMUM CURRENT" -to ddr3_hps_resetn
        -tag __hps_sdram_p0
799 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dm[0] -tag __hps_sdram_p0
800 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dm[1] -tag __hps_sdram_p0
801 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dm[2] -tag __hps_sdram_p0
802 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dm[3] -tag __hps_sdram_p0
803 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_hps_dm[4] -tag __hps_sdram_p0
804 set_instance_assignment -name ENABLE_BENEFICIAL_SKEW_OPTIMIZATION_FOR_NON_GLOBAL_CLOCKS ON
        -to u0|hps_0|hps_io|border|hps_sdram_inst -tag __hps_sdram_p0
805 set_instance_assignment -name INPUT_TERMINATION DIFFERENTIAL -to hsma_rx_d_p
806 set_instance_assignment -name INPUT_TERMINATION DIFFERENTIAL -to hsma_clk_in_p2
807 set_instance_assignment -name INPUT_TERMINATION "OCT 100 OHMS" -to clk_148_p
808 set_instance_assignment -name INPUT_TERMINATION "OCT 100 OHMS" -to pcie_refclk_p
809 set_instance_assignment -name INPUT_TERMINATION "OCT 100 OHMS" -to refclk_q12_p
810 set_location_assignment PIN_AA16 -to clk_25m_fpga
811 set_instance_assignment -name IO_STANDARD "1.5 V" -to clk_25m_fpga
812 set_location_assignment PIN_AJ2 -to hsma_clk_out_p1
```

```
813 set_location_assignment PIN_AC12 -to hsma_clk_out_n1
814 set_location_assignment PIN_AG2 -to hsma_clk_in_p1
815 set_location_assignment PIN_AH3 -to hsma_clk_in_n1
816
817 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_a[0] -tag
      __orion_system_mem_if_ddr3_fpga_p0
818 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_a[1] -tag
      __orion_system_mem_if_ddr3_fpga_p0
819 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_a[2] -tag
      __orion_system_mem_if_ddr3_fpga_p0
820 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_a[3] -tag
      __orion_system_mem_if_ddr3_fpga_p0
821 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_a[5] -tag
      __orion_system_mem_if_ddr3_fpga_p0
822 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_a[4] -tag
      __orion_system_mem_if_ddr3_fpga_p0
823 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_a[6] -tag
      __orion_system_mem_if_ddr3_fpga_p0
824 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_a[7] -tag
      __orion_system_mem_if_ddr3_fpga_p0
825 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_a[8] -tag
      __orion_system_mem_if_ddr3_fpga_p0
826 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_a[9] -tag
      __orion_system_mem_if_ddr3_fpga_p0
827 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_a[10] -tag
      __orion_system_mem_if_ddr3_fpga_p0
828 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_a[12] -tag
      __orion_system_mem_if_ddr3_fpga_p0
829 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_a[11] -tag
      __orion_system_mem_if_ddr3_fpga_p0
830 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_a[13] -tag
      __orion_system_mem_if_ddr3_fpga_p0
831 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_a[14] -tag
      __orion_system_mem_if_ddr3_fpga_p0
832 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_ba[2] -tag
      __orion_system_mem_if_ddr3_fpga_p0
833 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_ba[0] -tag
      __orion_system_mem_if_ddr3_fpga_p0
834 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_ba[1] -tag
      __orion_system_mem_if_ddr3_fpga_p0
835 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_dm[0] -tag
      __orion_system_mem_if_ddr3_fpga_p0
836 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_dm[1] -tag
      __orion_system_mem_if_ddr3_fpga_p0
837 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_dm[2] -tag
      __orion_system_mem_if_ddr3_fpga_p0
838 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_dm[3] -tag
      __orion_system_mem_if_ddr3_fpga_p0
839 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_dq[31] -tag
      __orion_system_mem_if_ddr3_fpga_p0
840 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_dq[0] -tag
      __orion_system_mem_if_ddr3_fpga_p0
841 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_dq[1] -tag
      __orion_system_mem_if_ddr3_fpga_p0
842 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_dq[2] -tag
      __orion_system_mem_if_ddr3_fpga_p0
843 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_dq[3] -tag
      __orion_system_mem_if_ddr3_fpga_p0
844 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_dq[4] -tag
      __orion_system_mem_if_ddr3_fpga_p0
845 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_dq[5] -tag
      __orion_system_mem_if_ddr3_fpga_p0
846 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_dq[6] -tag
      __orion_system_mem_if_ddr3_fpga_p0
847 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_dq[7] -tag
      __orion_system_mem_if_ddr3_fpga_p0
848 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_dq[8] -tag
      __orion_system_mem_if_ddr3_fpga_p0
```

```
849 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_dq[9] -tag
        __orion_system_mem_if_ddr3_fpga_p0
850 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_dq[10] -tag
        __orion_system_mem_if_ddr3_fpga_p0
851 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_dq[11] -tag
        __orion_system_mem_if_ddr3_fpga_p0
852 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_dq[12] -tag
        __orion_system_mem_if_ddr3_fpga_p0
853 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_dq[13] -tag
        __orion_system_mem_if_ddr3_fpga_p0
854 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_dq[14] -tag
        __orion_system_mem_if_ddr3_fpga_p0
855 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_dq[15] -tag
        __orion_system_mem_if_ddr3_fpga_p0
856 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_dq[16] -tag
        __orion_system_mem_if_ddr3_fpga_p0
857 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_dq[17] -tag
        __orion_system_mem_if_ddr3_fpga_p0
858 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_dq[18] -tag
        __orion_system_mem_if_ddr3_fpga_p0
859 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_dq[19] -tag
        __orion_system_mem_if_ddr3_fpga_p0
860 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_dq[20] -tag
        __orion_system_mem_if_ddr3_fpga_p0
861 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_dq[21] -tag
        __orion_system_mem_if_ddr3_fpga_p0
862 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_dq[22] -tag
        __orion_system_mem_if_ddr3_fpga_p0
863 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_dq[23] -tag
        __orion_system_mem_if_ddr3_fpga_p0
864 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_dq[24] -tag
        __orion_system_mem_if_ddr3_fpga_p0
865 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_dq[25] -tag
        __orion_system_mem_if_ddr3_fpga_p0
866 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_dq[26] -tag
        __orion_system_mem_if_ddr3_fpga_p0
867 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_dq[27] -tag
        __orion_system_mem_if_ddr3_fpga_p0
868 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_dq[28] -tag
        __orion_system_mem_if_ddr3_fpga_p0
869 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_dq[29] -tag
        __orion_system_mem_if_ddr3_fpga_p0
870 set_instance_assignment -name IO_STANDARD "SSTL-15 CLASS I" -to ddr3_fpga_dq[30] -tag
        __orion_system_mem_if_ddr3_fpga_p0
871 set_instance_assignment -name IO_STANDARD "DIFFERENTIAL 1.5-V SSTL CLASS I" -to
        ddr3_fpga_dqs_n[0] -tag __orion_system_mem_if_ddr3_fpga_p0
872 set_instance_assignment -name IO_STANDARD "DIFFERENTIAL 1.5-V SSTL CLASS I" -to
        ddr3_fpga_dqs_n[1] -tag __orion_system_mem_if_ddr3_fpga_p0
873 set_instance_assignment -name IO_STANDARD "DIFFERENTIAL 1.5-V SSTL CLASS I" -to
        ddr3_fpga_dqs_n[2] -tag __orion_system_mem_if_ddr3_fpga_p0
874 set_instance_assignment -name IO_STANDARD "DIFFERENTIAL 1.5-V SSTL CLASS I" -to
        ddr3_fpga_dqs_n[3] -tag __orion_system_mem_if_ddr3_fpga_p0
875 set_instance_assignment -name IO_STANDARD "DIFFERENTIAL 1.5-V SSTL CLASS I" -to
        ddr3_fpga_dqs_p[0] -tag __orion_system_mem_if_ddr3_fpga_p0
876 set_instance_assignment -name IO_STANDARD "DIFFERENTIAL 1.5-V SSTL CLASS I" -to
        ddr3_fpga_dqs_p[1] -tag __orion_system_mem_if_ddr3_fpga_p0
877 set_instance_assignment -name IO_STANDARD "DIFFERENTIAL 1.5-V SSTL CLASS I" -to
        ddr3_fpga_dqs_p[2] -tag __orion_system_mem_if_ddr3_fpga_p0
878 set_instance_assignment -name IO_STANDARD "DIFFERENTIAL 1.5-V SSTL CLASS I" -to
        ddr3_fpga_dqs_p[3] -tag __orion_system_mem_if_ddr3_fpga_p0
879 set_instance_assignment -name IO_STANDARD LVDS -to pcie_refclk_n
880 set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to enet_hps_rxd[0]
881 set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to enet_hps_rxd[1]
882 set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to enet_hps_rxd[2]
883 set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to enet_hps_rxd[3]
884 set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to enet_hps_txd[3]
885 set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to enet_hps_txd[0]
886 set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to enet_hps_txd[1]
```

```
887 set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to enet_hps_txd[2]
888 set_instance_assignment -name IO_STANDARD "1.5-V PCML" -to hsma_tx_p[3]
889 set_instance_assignment -name IO_STANDARD "1.5-V PCML" -to hsma_tx_p[2]
890 set_instance_assignment -name IO_STANDARD "1.5-V PCML" -to hsma_tx_p[1]
891 set_instance_assignment -name IO_STANDARD "1.5-V PCML" -to hsma_tx_p[0]
892 set_instance_assignment -name IO_STANDARD "1.5-V PCML" -to hsma_tx_n[3]
893 set_instance_assignment -name IO_STANDARD "1.5-V PCML" -to hsma_tx_n[2]
894 set_instance_assignment -name IO_STANDARD "1.5-V PCML" -to hsma_tx_n[1]
895 set_instance_assignment -name IO_STANDARD "1.5-V PCML" -to hsma_tx_n[0]
896 set_instance_assignment -name IO_STANDARD "2.5 V" -to hsma_sda
897 set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to qspi_io[0]
898 set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to qspi_io[1]
899 set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to qspi_io[2]
900 set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to qspi_io[3]
901 set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to sd_dat[0]
902 set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to sd_dat[1]
903 set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to sd_dat[2]
904 set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to sd_dat[3]
905 set_instance_assignment -name IO_STANDARD "1.8 V" -to user_pb_hps[0]
906 set_instance_assignment -name IO_STANDARD "1.8 V" -to user_pb_hps[1]
907 set_instance_assignment -name IO_STANDARD "1.8 V" -to user_dipsw_hps[0]
908 set_instance_assignment -name IO_STANDARD "1.8 V" -to user_dipsw_hps[1]
909 set_instance_assignment -name IO_STANDARD "1.8 V" -to user_dipsw_hps[2]
910 set_instance_assignment -name IO_STANDARD "1.8 V" -to user_dipsw_hps[3]
911 set_instance_assignment -name IO_STANDARD "1.8 V" -to user_pb_hps[2]
912 set_instance_assignment -name IO_STANDARD "1.8 V" -to user_pb_hps[3]
913 set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to usb_wrn
914 set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to usb_sda
915 set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to usb_scl
916 set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to usb_resetn
917 set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to usb_rdn
918 set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to usb_oen
919 set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to usb_full
920 set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to usb_empty
921 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
      ddr3_fpga_dq[0] -tag __orion_system_mem_if_ddr3_fpga_p0
922 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
      ddr3_fpga_dq[0] -tag __orion_system_mem_if_ddr3_fpga_p0
923 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
      ddr3_fpga_dq[1] -tag __orion_system_mem_if_ddr3_fpga_p0
924 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
      ddr3_fpga_dq[1] -tag __orion_system_mem_if_ddr3_fpga_p0
925 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
      ddr3_fpga_dq[2] -tag __orion_system_mem_if_ddr3_fpga_p0
926 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
      ddr3_fpga_dq[2] -tag __orion_system_mem_if_ddr3_fpga_p0
927 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
      ddr3_fpga_dq[3] -tag __orion_system_mem_if_ddr3_fpga_p0
928 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
      ddr3_fpga_dq[3] -tag __orion_system_mem_if_ddr3_fpga_p0
929 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
      ddr3_fpga_dq[4] -tag __orion_system_mem_if_ddr3_fpga_p0
930 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
      ddr3_fpga_dq[4] -tag __orion_system_mem_if_ddr3_fpga_p0
931 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
      ddr3_fpga_dq[5] -tag __orion_system_mem_if_ddr3_fpga_p0
932 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
      ddr3_fpga_dq[5] -tag __orion_system_mem_if_ddr3_fpga_p0
933 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
      ddr3_fpga_dq[6] -tag __orion_system_mem_if_ddr3_fpga_p0
934 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
      ddr3_fpga_dq[6] -tag __orion_system_mem_if_ddr3_fpga_p0
935 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
      ddr3_fpga_dq[7] -tag __orion_system_mem_if_ddr3_fpga_p0
936 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
      ddr3_fpga_dq[7] -tag __orion_system_mem_if_ddr3_fpga_p0
937 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
      ddr3_fpga_dq[8] -tag __orion_system_mem_if_ddr3_fpga_p0
```

```
938 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_fpga_dq[8] -tag __orion_system_mem_if_ddr3_fpga_p0
939 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_fpga_dq[9] -tag __orion_system_mem_if_ddr3_fpga_p0
940 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_fpga_dq[9] -tag __orion_system_mem_if_ddr3_fpga_p0
941 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_fpga_dq[10] -tag __orion_system_mem_if_ddr3_fpga_p0
942 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_fpga_dq[10] -tag __orion_system_mem_if_ddr3_fpga_p0
943 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_fpga_dq[11] -tag __orion_system_mem_if_ddr3_fpga_p0
944 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_fpga_dq[11] -tag __orion_system_mem_if_ddr3_fpga_p0
945 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_fpga_dq[12] -tag __orion_system_mem_if_ddr3_fpga_p0
946 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_fpga_dq[12] -tag __orion_system_mem_if_ddr3_fpga_p0
947 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_fpga_dq[13] -tag __orion_system_mem_if_ddr3_fpga_p0
948 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_fpga_dq[13] -tag __orion_system_mem_if_ddr3_fpga_p0
949 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_fpga_dq[14] -tag __orion_system_mem_if_ddr3_fpga_p0
950 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_fpga_dq[14] -tag __orion_system_mem_if_ddr3_fpga_p0
951 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_fpga_dq[15] -tag __orion_system_mem_if_ddr3_fpga_p0
952 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_fpga_dq[15] -tag __orion_system_mem_if_ddr3_fpga_p0
953 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_fpga_dq[16] -tag __orion_system_mem_if_ddr3_fpga_p0
954 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_fpga_dq[16] -tag __orion_system_mem_if_ddr3_fpga_p0
955 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_fpga_dq[17] -tag __orion_system_mem_if_ddr3_fpga_p0
956 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_fpga_dq[17] -tag __orion_system_mem_if_ddr3_fpga_p0
957 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_fpga_dq[18] -tag __orion_system_mem_if_ddr3_fpga_p0
958 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_fpga_dq[18] -tag __orion_system_mem_if_ddr3_fpga_p0
959 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_fpga_dq[19] -tag __orion_system_mem_if_ddr3_fpga_p0
960 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_fpga_dq[19] -tag __orion_system_mem_if_ddr3_fpga_p0
961 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_fpga_dq[20] -tag __orion_system_mem_if_ddr3_fpga_p0
962 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_fpga_dq[20] -tag __orion_system_mem_if_ddr3_fpga_p0
963 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_fpga_dq[21] -tag __orion_system_mem_if_ddr3_fpga_p0
964 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_fpga_dq[21] -tag __orion_system_mem_if_ddr3_fpga_p0
965 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_fpga_dq[22] -tag __orion_system_mem_if_ddr3_fpga_p0
966 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_fpga_dq[22] -tag __orion_system_mem_if_ddr3_fpga_p0
967 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_fpga_dq[23] -tag __orion_system_mem_if_ddr3_fpga_p0
968 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_fpga_dq[23] -tag __orion_system_mem_if_ddr3_fpga_p0
969 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_fpga_dq[24] -tag __orion_system_mem_if_ddr3_fpga_p0
970 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
        ddr3_fpga_dq[24] -tag __orion_system_mem_if_ddr3_fpga_p0
971 set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
        ddr3_fpga_dq[25] -tag __orion_system_mem_if_ddr3_fpga_p0
```

```
set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
    ddr3_fpga_dq[25] -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
    ddr3_fpga_dq[26] -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
    ddr3_fpga_dq[26] -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
    ddr3_fpga_dq[27] -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
    ddr3_fpga_dq[27] -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
    ddr3_fpga_dq[28] -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
    ddr3_fpga_dq[28] -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
    ddr3_fpga_dq[29] -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
    ddr3_fpga_dq[29] -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
    ddr3_fpga_dq[30] -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
    ddr3_fpga_dq[30] -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
    ddr3_fpga_dq[31] -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
    ddr3_fpga_dq[31] -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
    ddr3_fpga_dqs_p[0] -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
    ddr3_fpga_dqs_p[0] -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
    ddr3_fpga_dqs_p[1] -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
    ddr3_fpga_dqs_p[1] -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
    ddr3_fpga_dqs_p[2] -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
    ddr3_fpga_dqs_p[2] -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
    ddr3_fpga_dqs_p[3] -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
    ddr3_fpga_dqs_p[3] -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
    ddr3_fpga_dqs_n[0] -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
    ddr3_fpga_dqs_n[0] -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
    ddr3_fpga_dqs_n[1] -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
    ddr3_fpga_dqs_n[1] -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
    ddr3_fpga_dqs_n[2] -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
    ddr3_fpga_dqs_n[2] -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name INPUT_TERMINATION "PARALLEL 50 OHM WITH CALIBRATION" -to
    ddr3_fpga_dqs_n[3] -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
    ddr3_fpga_dqs_n[3] -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITHOUT CALIBRATION" -to
    ddr3_fpga_clk_p -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name D5_DELAY 2 -to ddr3_fpga_clk_p -tag
    __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITHOUT CALIBRATION" -to
    ddr3_fpga_clk_n -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name D5_DELAY 2 -to ddr3_fpga_clk_n -tag
    __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name CURRENT_STRENGTH_NEW "MAXIMUM CURRENT" -to ddr3_fpga_a[0]
    -tag __orion_system_mem_if_ddr3_fpga_p0
```

```
1006 set_instance_assignment -name CURRENT_STRENGTH_NEW "MAXIMUM CURRENT" -to ddr3_fpga_a[10]
     -tag __orion_system_mem_if_ddr3_fpga_p0
1007 set_instance_assignment -name CURRENT_STRENGTH_NEW "MAXIMUM CURRENT" -to ddr3_fpga_a[11]
     -tag __orion_system_mem_if_ddr3_fpga_p0
1008 set_instance_assignment -name CURRENT_STRENGTH_NEW "MAXIMUM CURRENT" -to ddr3_fpga_a[12]
     -tag __orion_system_mem_if_ddr3_fpga_p0
1009 set_instance_assignment -name CURRENT_STRENGTH_NEW "MAXIMUM CURRENT" -to ddr3_fpga_a[13]
     -tag __orion_system_mem_if_ddr3_fpga_p0
1010 set_instance_assignment -name CURRENT_STRENGTH_NEW "MAXIMUM CURRENT" -to ddr3_fpga_a[14]
     -tag __orion_system_mem_if_ddr3_fpga_p0
1011 set_instance_assignment -name CURRENT_STRENGTH_NEW "MAXIMUM CURRENT" -to ddr3_fpga_a[1]
     -tag __orion_system_mem_if_ddr3_fpga_p0
1012 set_instance_assignment -name CURRENT_STRENGTH_NEW "MAXIMUM CURRENT" -to ddr3_fpga_a[2]
     -tag __orion_system_mem_if_ddr3_fpga_p0
1013 set_instance_assignment -name CURRENT_STRENGTH_NEW "MAXIMUM CURRENT" -to ddr3_fpga_a[3]
     -tag __orion_system_mem_if_ddr3_fpga_p0
1014 set_instance_assignment -name CURRENT_STRENGTH_NEW "MAXIMUM CURRENT" -to ddr3_fpga_a[4]
     -tag __orion_system_mem_if_ddr3_fpga_p0
1015 set_instance_assignment -name CURRENT_STRENGTH_NEW "MAXIMUM CURRENT" -to ddr3_fpga_a[5]
     -tag __orion_system_mem_if_ddr3_fpga_p0
1016 set_instance_assignment -name CURRENT_STRENGTH_NEW "MAXIMUM CURRENT" -to ddr3_fpga_a[6]
     -tag __orion_system_mem_if_ddr3_fpga_p0
1017 set_instance_assignment -name CURRENT_STRENGTH_NEW "MAXIMUM CURRENT" -to ddr3_fpga_a[7]
     -tag __orion_system_mem_if_ddr3_fpga_p0
1018 set_instance_assignment -name CURRENT_STRENGTH_NEW "MAXIMUM CURRENT" -to ddr3_fpga_a[8]
     -tag __orion_system_mem_if_ddr3_fpga_p0
1019 set_instance_assignment -name CURRENT_STRENGTH_NEW "MAXIMUM CURRENT" -to ddr3_fpga_a[9]
     -tag __orion_system_mem_if_ddr3_fpga_p0
1020 set_instance_assignment -name CURRENT_STRENGTH_NEW "MAXIMUM CURRENT" -to ddr3_fpga_ba[0]
     -tag __orion_system_mem_if_ddr3_fpga_p0
1021 set_instance_assignment -name CURRENT_STRENGTH_NEW "MAXIMUM CURRENT" -to ddr3_fpga_ba[1]
     -tag __orion_system_mem_if_ddr3_fpga_p0
1022 set_instance_assignment -name CURRENT_STRENGTH_NEW "MAXIMUM CURRENT" -to ddr3_fpga_ba[2]
     -tag __orion_system_mem_if_ddr3_fpga_p0
1023 set_instance_assignment -name CURRENT_STRENGTH_NEW "MAXIMUM CURRENT" -to ddr3_fpga_casn
     -tag __orion_system_mem_if_ddr3_fpga_p0
1024 set_instance_assignment -name CURRENT_STRENGTH_NEW "MAXIMUM CURRENT" -to ddr3_fpga_cke
     -tag __orion_system_mem_if_ddr3_fpga_p0
1025 set_instance_assignment -name CURRENT_STRENGTH_NEW "MAXIMUM CURRENT" -to ddr3_fpga_csn
     -tag __orion_system_mem_if_ddr3_fpga_p0
1026 set_instance_assignment -name CURRENT_STRENGTH_NEW "MAXIMUM CURRENT" -to ddr3_fpga_odt
     -tag __orion_system_mem_if_ddr3_fpga_p0
1027 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
     ddr3_fpga_dm[0] -tag __orion_system_mem_if_ddr3_fpga_p0
1028 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
     ddr3_fpga_dm[1] -tag __orion_system_mem_if_ddr3_fpga_p0
1029 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
     ddr3_fpga_dm[2] -tag __orion_system_mem_if_ddr3_fpga_p0
1030 set_instance_assignment -name OUTPUT_TERMINATION "SERIES 50 OHM WITH CALIBRATION" -to
     ddr3_fpga_dm[3] -tag __orion_system_mem_if_ddr3_fpga_p0
1031 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_dq[0] -tag
     __orion_system_mem_if_ddr3_fpga_p0
1032 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_dq[1] -tag
     __orion_system_mem_if_ddr3_fpga_p0
1033 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_dq[2] -tag
     __orion_system_mem_if_ddr3_fpga_p0
1034 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_dq[3] -tag
     __orion_system_mem_if_ddr3_fpga_p0
1035 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_dq[4] -tag
     __orion_system_mem_if_ddr3_fpga_p0
1036 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_dq[5] -tag
     __orion_system_mem_if_ddr3_fpga_p0
1037 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_dq[6] -tag
     __orion_system_mem_if_ddr3_fpga_p0
1038 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_dq[7] -tag
     __orion_system_mem_if_ddr3_fpga_p0
1039 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_dq[8] -tag
     __orion_system_mem_if_ddr3_fpga_p0
```

```
1040 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_dq[9] -tag
         __orion_system_mem_if_ddr3_fpga_p0
1041 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_dq[10] -tag
         __orion_system_mem_if_ddr3_fpga_p0
1042 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_dq[11] -tag
         __orion_system_mem_if_ddr3_fpga_p0
1043 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_dq[12] -tag
         __orion_system_mem_if_ddr3_fpga_p0
1044 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_dq[13] -tag
         __orion_system_mem_if_ddr3_fpga_p0
1045 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_dq[14] -tag
         __orion_system_mem_if_ddr3_fpga_p0
1046 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_dq[15] -tag
         __orion_system_mem_if_ddr3_fpga_p0
1047 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_dq[16] -tag
         __orion_system_mem_if_ddr3_fpga_p0
1048 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_dq[17] -tag
         __orion_system_mem_if_ddr3_fpga_p0
1049 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_dq[18] -tag
         __orion_system_mem_if_ddr3_fpga_p0
1050 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_dq[19] -tag
         __orion_system_mem_if_ddr3_fpga_p0
1051 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_dq[20] -tag
         __orion_system_mem_if_ddr3_fpga_p0
1052 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_dq[21] -tag
         __orion_system_mem_if_ddr3_fpga_p0
1053 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_dq[22] -tag
         __orion_system_mem_if_ddr3_fpga_p0
1054 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_dq[23] -tag
         __orion_system_mem_if_ddr3_fpga_p0
1055 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_dq[24] -tag
         __orion_system_mem_if_ddr3_fpga_p0
1056 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_dq[25] -tag
         __orion_system_mem_if_ddr3_fpga_p0
1057 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_dq[26] -tag
         __orion_system_mem_if_ddr3_fpga_p0
1058 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_dq[27] -tag
         __orion_system_mem_if_ddr3_fpga_p0
1059 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_dq[28] -tag
         __orion_system_mem_if_ddr3_fpga_p0
1060 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_dq[29] -tag
         __orion_system_mem_if_ddr3_fpga_p0
1061 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_dq[30] -tag
         __orion_system_mem_if_ddr3_fpga_p0
1062 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_dq[31] -tag
         __orion_system_mem_if_ddr3_fpga_p0
1063 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_dm[0] -tag
         __orion_system_mem_if_ddr3_fpga_p0
1064 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_dm[1] -tag
         __orion_system_mem_if_ddr3_fpga_p0
1065 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_dm[2] -tag
         __orion_system_mem_if_ddr3_fpga_p0
1066 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_dm[3] -tag
         __orion_system_mem_if_ddr3_fpga_p0
1067 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_dqs_p[0] -tag
         __orion_system_mem_if_ddr3_fpga_p0
1068 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_dqs_p[1] -tag
         __orion_system_mem_if_ddr3_fpga_p0
1069 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_dqs_p[2] -tag
         __orion_system_mem_if_ddr3_fpga_p0
1070 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_dqs_p[3] -tag
         __orion_system_mem_if_ddr3_fpga_p0
1071 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_dqs_n[0] -tag
         __orion_system_mem_if_ddr3_fpga_p0
1072 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_dqs_n[1] -tag
         __orion_system_mem_if_ddr3_fpga_p0
1073 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_dqs_n[2] -tag
         __orion_system_mem_if_ddr3_fpga_p0
```

```
set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_dqs_n[3] -tag
    __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_a[0] -tag
    __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_a[10] -tag
    __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_a[11] -tag
    __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_a[12] -tag
    __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_a[13] -tag
    __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_a[14] -tag
    __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_a[1] -tag
    __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_a[2] -tag
    __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_a[3] -tag
    __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_a[4] -tag
    __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_a[5] -tag
    __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_a[6] -tag
    __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_a[7] -tag
    __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_a[8] -tag
    __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_a[9] -tag
    __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_ba[0] -tag
    __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_ba[1] -tag
    __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_ba[2] -tag
    __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_casn -tag
    __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_cke -tag
    __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_csn -tag
    __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_odt -tag
    __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_clk_p -tag
    __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_fpga_clk_n -tag
    __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name GLOBAL_SIGNAL "DUAL-REGIONAL CLOCK" -to u0|mem_if_ddr3_fpga|
    pll0|pll_avl_clk -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name GLOBAL_SIGNAL "DUAL-REGIONAL CLOCK" -to u0|mem_if_ddr3_fpga|
    pll0|pll_config_clk -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name GLOBAL_SIGNAL OFF -to u0|mem_if_ddr3_fpga|p0|umemphy|ureset|
    phy_reset_mem_stable_n -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name GLOBAL_SIGNAL OFF -to u0|mem_if_ddr3_fpga|p0|umemphy|ureset|
    phy_reset_n -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name GLOBAL_SIGNAL OFF -to u0|mem_if_ddr3_fpga|s0|
    sequencer_rw_mgr_inst|rw_mgr_inst|rw_mgr_core_inst|rw_soft_reset_n -tag
    __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name GLOBAL_SIGNAL OFF -to u0|mem_if_ddr3_fpga|p0|umemphy|
    uio_pads|dq_ddio[0].read_capture_clk_buffer -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name GLOBAL_SIGNAL OFF -to u0|mem_if_ddr3_fpga|p0|umemphy|
    uread_datapath|reset_n_fifo_write_side[0] -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name GLOBAL_SIGNAL OFF -to u0|mem_if_ddr3_fpga|p0|umemphy|
    uread_datapath|reset_n_fifo_wraddress[0] -tag __orion_system_mem_if_ddr3_fpga_p0
```

```
set_instance_assignment -name GLOBAL_SIGNAL OFF -to u0|mem_if_ddr3_fpga|p0|umemphy|
    uio_pads|dq_ddio[1].read_capture_clk_buffer -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name GLOBAL_SIGNAL OFF -to u0|mem_if_ddr3_fpga|p0|umemphy|
    uread_datapath|reset_n_fifo_write_side[1] -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name GLOBAL_SIGNAL OFF -to u0|mem_if_ddr3_fpga|p0|umemphy|
    uread_datapath|reset_n_fifo_wraddress[1] -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name GLOBAL_SIGNAL OFF -to u0|mem_if_ddr3_fpga|p0|umemphy|
    uio_pads|dq_ddio[2].read_capture_clk_buffer -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name GLOBAL_SIGNAL OFF -to u0|mem_if_ddr3_fpga|p0|umemphy|
    uread_datapath|reset_n_fifo_write_side[2] -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name GLOBAL_SIGNAL OFF -to u0|mem_if_ddr3_fpga|p0|umemphy|
    uread_datapath|reset_n_fifo_wraddress[2] -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name GLOBAL_SIGNAL OFF -to u0|mem_if_ddr3_fpga|p0|umemphy|
    uio_pads|dq_ddio[3].read_capture_clk_buffer -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name GLOBAL_SIGNAL OFF -to u0|mem_if_ddr3_fpga|p0|umemphy|
    uread_datapath|reset_n_fifo_write_side[3] -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name GLOBAL_SIGNAL OFF -to u0|mem_if_ddr3_fpga|p0|umemphy|
    uread_datapath|reset_n_fifo_wraddress[3] -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name ENABLE_BENEFICIAL_SKEW_OPTIMIZATION_FOR_NON_GLOBAL_CLOCKS ON
     -to u0|mem_if_ddr3_fpga -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name PLL_COMPENSATION_MODE DIRECT -to u0|mem_if_ddr3_fpga|pll0|
    fbout -tag __orion_system_mem_if_ddr3_fpga_p0
set_instance_assignment -name D5_DELAY 2 -to ddr3_hps_clk_p -tag __hps_sdram_p0
set_instance_assignment -name D5_DELAY 2 -to ddr3_hps_clk_n -tag __hps_sdram_p0
set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dq[0] -tag
    __hps_sdram_p0
set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dq[1] -tag
    __hps_sdram_p0
set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dq[2] -tag
    __hps_sdram_p0
set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dq[3] -tag
    __hps_sdram_p0
set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dq[4] -tag
    __hps_sdram_p0
set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dq[5] -tag
    __hps_sdram_p0
set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dq[6] -tag
    __hps_sdram_p0
set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dq[7] -tag
    __hps_sdram_p0
set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dq[8] -tag
    __hps_sdram_p0
set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dq[9] -tag
    __hps_sdram_p0
set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dq[10] -tag
    __hps_sdram_p0
set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dq[11] -tag
    __hps_sdram_p0
set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dq[12] -tag
    __hps_sdram_p0
set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dq[13] -tag
    __hps_sdram_p0
set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dq[14] -tag
    __hps_sdram_p0
set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dq[15] -tag
    __hps_sdram_p0
set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dq[16] -tag
    __hps_sdram_p0
set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dq[17] -tag
    __hps_sdram_p0
set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dq[18] -tag
    __hps_sdram_p0
set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dq[19] -tag
    __hps_sdram_p0
set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dq[20] -tag
    __hps_sdram_p0
set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dq[21] -tag
    __hps_sdram_p0
```

```
1142 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dq[22] -tag
     __hps_sdram_p0
1143 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dq[23] -tag
     __hps_sdram_p0
1144 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dq[24] -tag
     __hps_sdram_p0
1145 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dq[25] -tag
     __hps_sdram_p0
1146 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dq[26] -tag
     __hps_sdram_p0
1147 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dq[27] -tag
     __hps_sdram_p0
1148 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dq[28] -tag
     __hps_sdram_p0
1149 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dq[29] -tag
     __hps_sdram_p0
1150 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dq[30] -tag
     __hps_sdram_p0
1151 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dq[31] -tag
     __hps_sdram_p0
1152 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dq[32] -tag
     __hps_sdram_p0
1153 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dq[33] -tag
     __hps_sdram_p0
1154 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dq[34] -tag
     __hps_sdram_p0
1155 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dq[35] -tag
     __hps_sdram_p0
1156 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dq[36] -tag
     __hps_sdram_p0
1157 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dq[37] -tag
     __hps_sdram_p0
1158 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dq[38] -tag
     __hps_sdram_p0
1159 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dq[39] -tag
     __hps_sdram_p0
1160 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dm[0] -tag
     __hps_sdram_p0
1161 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dm[1] -tag
     __hps_sdram_p0
1162 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dm[2] -tag
     __hps_sdram_p0
1163 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dm[3] -tag
     __hps_sdram_p0
1164 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dm[4] -tag
     __hps_sdram_p0
1165 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dqs_p[0] -tag
     __hps_sdram_p0
1166 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dqs_p[1] -tag
     __hps_sdram_p0
1167 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dqs_p[2] -tag
     __hps_sdram_p0
1168 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dqs_p[3] -tag
     __hps_sdram_p0
1169 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dqs_p[4] -tag
     __hps_sdram_p0
1170 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dqs_n[0] -tag
     __hps_sdram_p0
1171 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dqs_n[1] -tag
     __hps_sdram_p0
1172 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dqs_n[2] -tag
     __hps_sdram_p0
1173 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dqs_n[3] -tag
     __hps_sdram_p0
1174 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_dqs_n[4] -tag
     __hps_sdram_p0
1175 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_a[0] -tag
     __hps_sdram_p0
```

```
1176 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_a[10] -tag
     __hps_sdram_p0
1177 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_a[11] -tag
     __hps_sdram_p0
1178 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_a[12] -tag
     __hps_sdram_p0
1179 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_a[13] -tag
     __hps_sdram_p0
1180 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_a[14] -tag
     __hps_sdram_p0
1181 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_a[1] -tag
     __hps_sdram_p0
1182 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_a[2] -tag
     __hps_sdram_p0
1183 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_a[3] -tag
     __hps_sdram_p0
1184 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_a[4] -tag
     __hps_sdram_p0
1185 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_a[5] -tag
     __hps_sdram_p0
1186 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_a[6] -tag
     __hps_sdram_p0
1187 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_a[7] -tag
     __hps_sdram_p0
1188 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_a[8] -tag
     __hps_sdram_p0
1189 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_a[9] -tag
     __hps_sdram_p0
1190 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_ba[0] -tag
     __hps_sdram_p0
1191 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_ba[1] -tag
     __hps_sdram_p0
1192 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_ba[2] -tag
     __hps_sdram_p0
1193 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_casn -tag
     __hps_sdram_p0
1194 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_cke -tag
     __hps_sdram_p0
1195 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_csn -tag
     __hps_sdram_p0
1196 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_odt -tag
     __hps_sdram_p0
1197 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_rasn -tag
     __hps_sdram_p0
1198 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_wen -tag
     __hps_sdram_p0
1199 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_resetn -tag
     __hps_sdram_p0
1200 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_clk_p -tag
     __hps_sdram_p0
1201 set_instance_assignment -name PACKAGE_SKEW_COMPENSATION OFF -to ddr3_hps_clk_n -tag
     __hps_sdram_p0
1202 set_instance_assignment -name GLOBAL_SIGNAL OFF -to u0|hps_0|hps_io|border|hps_sdram_inst|
     p0|umemphy|ureset|phy_reset_mem_stable_n -tag __hps_sdram_p0
1203 set_instance_assignment -name GLOBAL_SIGNAL OFF -to u0|hps_0|hps_io|border|hps_sdram_inst|
     p0|umemphy|ureset|phy_reset_n -tag __hps_sdram_p0
1204 set_instance_assignment -name GLOBAL_SIGNAL OFF -to u0|hps_0|hps_io|border|hps_sdram_inst|
     p0|umemphy|uio_pads|dq_ddio[0].read_capture_clk_buffer -tag __hps_sdram_p0
1205 set_instance_assignment -name GLOBAL_SIGNAL OFF -to u0|hps_0|hps_io|border|hps_sdram_inst|
     p0|umemphy|uread_datapath|reset_n_fifo_write_side[0] -tag __hps_sdram_p0
1206 set_instance_assignment -name GLOBAL_SIGNAL OFF -to u0|hps_0|hps_io|border|hps_sdram_inst|
     p0|umemphy|uread_datapath|reset_n_fifo_wraddress[0] -tag __hps_sdram_p0
1207 set_instance_assignment -name GLOBAL_SIGNAL OFF -to u0|hps_0|hps_io|border|hps_sdram_inst|
     p0|umemphy|uio_pads|dq_ddio[1].read_capture_clk_buffer -tag __hps_sdram_p0
1208 set_instance_assignment -name GLOBAL_SIGNAL OFF -to u0|hps_0|hps_io|border|hps_sdram_inst|
     p0|umemphy|uread_datapath|reset_n_fifo_write_side[1] -tag __hps_sdram_p0
1209 set_instance_assignment -name GLOBAL_SIGNAL OFF -to u0|hps_0|hps_io|border|hps_sdram_inst|
     p0|umemphy|uread_datapath|reset_n_fifo_wraddress[1] -tag __hps_sdram_p0
```

```
set_instance_assignment -name GLOBAL_SIGNAL OFF -to u0|hps_0|hps_io|border|hps_sdram_inst|p0|umemphy|uio_pads|dq_ddio[2].read_capture_clk_buffer -tag __hps_sdram_p0
set_instance_assignment -name GLOBAL_SIGNAL OFF -to u0|hps_0|hps_io|border|hps_sdram_inst|p0|umemphy|uread_datapath|reset_n_fifo_write_side[2] -tag __hps_sdram_p0
set_instance_assignment -name GLOBAL_SIGNAL OFF -to u0|hps_0|hps_io|border|hps_sdram_inst|p0|umemphy|uread_datapath|reset_n_fifo_wraddress[2] -tag __hps_sdram_p0
set_instance_assignment -name GLOBAL_SIGNAL OFF -to u0|hps_0|hps_io|border|hps_sdram_inst|p0|umemphy|uio_pads|dq_ddio[3].read_capture_clk_buffer -tag __hps_sdram_p0
set_instance_assignment -name GLOBAL_SIGNAL OFF -to u0|hps_0|hps_io|border|hps_sdram_inst|p0|umemphy|uread_datapath|reset_n_fifo_write_side[3] -tag __hps_sdram_p0
set_instance_assignment -name GLOBAL_SIGNAL OFF -to u0|hps_0|hps_io|border|hps_sdram_inst|p0|umemphy|uread_datapath|reset_n_fifo_wraddress[3] -tag __hps_sdram_p0
set_instance_assignment -name GLOBAL_SIGNAL OFF -to u0|hps_0|hps_io|border|hps_sdram_inst|p0|umemphy|uio_pads|dq_ddio[4].read_capture_clk_buffer -tag __hps_sdram_p0
set_instance_assignment -name GLOBAL_SIGNAL OFF -to u0|hps_0|hps_io|border|hps_sdram_inst|p0|umemphy|uread_datapath|reset_n_fifo_write_side[4] -tag __hps_sdram_p0
set_instance_assignment -name GLOBAL_SIGNAL OFF -to u0|hps_0|hps_io|border|hps_sdram_inst|p0|umemphy|uread_datapath|reset_n_fifo_wraddress[4] -tag __hps_sdram_p0
set_instance_assignment -name PLL_COMPENSATION_MODE DIRECT -to u0|hps_0|hps_io|border|hps_sdram_inst|pll0|fbout -tag __hps_sdram_p0
set_location_assignment PIN_W16 -to ddr3_fpga_dqs_n[0]
set_location_assignment PIN_W17 -to ddr3_fpga_dqs_n[1]
set_location_assignment PIN_AA18 -to ddr3_fpga_dqs_n[2]
set_location_assignment PIN_AD19 -to ddr3_fpga_dqs_n[3]
set_location_assignment PIN_V16 -to ddr3_fpga_dqs_p[0]
set_location_assignment PIN_V17 -to ddr3_fpga_dqs_p[1]
set_location_assignment PIN_Y17 -to ddr3_fpga_dqs_p[2]
set_location_assignment PIN_AC20 -to ddr3_fpga_dqs_p[3]
set_location_assignment PIN_H3 -to hsma_tx_p[3]
set_location_assignment PIN_K3 -to hsma_tx_p[2]
set_location_assignment PIN_M3 -to hsma_tx_p[1]
set_location_assignment PIN_P3 -to hsma_tx_p[0]
set_location_assignment PIN_H4 -to hsma_tx_n[3]
set_location_assignment PIN_K4 -to hsma_tx_n[2]
set_location_assignment PIN_M4 -to hsma_tx_n[1]
set_location_assignment PIN_P4 -to hsma_tx_n[0]
set_global_assignment -name PHYSICAL_SYNTHESIS_COMBO_LOGIC ON
set_global_assignment -name PHYSICAL_SYNTHESIS_REGISTER_RETIMING ON
set_global_assignment -name PHYSICAL_SYNTHESIS_ASYNCHRONOUS_SIGNAL_PIPELINING ON
set_global_assignment -name PHYSICAL_SYNTHESIS_REGISTER_DUPLICATION ON
set_global_assignment -name PHYSICAL_SYNTHESIS_COMBO_LOGIC_FOR_AREA ON
set_global_assignment -name PHYSICAL_SYNTHESIS_EFFORT EXTRA
set_global_assignment -name OPTIMIZE_HOLD_TIMING "ALL PATHS"
set_global_assignment -name OPTIMIZE_POWER_DURING_FITTING OFF
set_global_assignment -name FITTER_EFFORT "STANDARD FIT"
set_global_assignment -name ROUTER_TIMING_OPTIMIZATION_LEVEL MAXIMUM
set_global_assignment -name ROUTER_CLOCKING_TOPOLOGY_ANALYSIS ON
set_global_assignment -name AUTO_PACKED_REGISTERS_STRATIXII NORMAL
set_global_assignment -name ROUTER_LCELL_INSERTION_AND_LOGIC_DUPLICATION ON
set_instance_assignment -name PLL_AUTO_RESET OFF -to "*q_sys_pll_0*|altera_pll:altera_pll_i*|*"
set_instance_assignment -name PLL_BANDWIDTH_PRESET AUTO -to "*q_sys_pll_0*|altera_pll:altera_pll_i*|*"
set_instance_assignment -name PLL_COMPENSATION_MODE NORMAL -to "*q_sys_pll_0*|altera_pll:altera_pll_i*|*"
set_global_assignment -name SEED 100
set_global_assignment -name SDC_FILE timing.base.sdc
set_global_assignment -name VERILOG_FILE orion_top.v
set_global_assignment -name QIP_FILE orion_system/synthesis/orion_system.qip
set_global_assignment -name VERILOG_FILE ip/debounce/debounce.v
set_global_assignment -name QIP_FILE clk_cntrl.qip
set_instance_assignment -name PARTITION_HIERARCHY root_partition -to | -section_id Top
```

Listing 16.6: Quartus II Settings File (orion_system.qsf)

16.3.8 Demo IP Top Verilog File (test_ip_top.v)

```
/*
Christopher Yarp

This file was created based on the Altera custom_master template.
This file supports multiple ports and connects to the DSP logic
directly without requireing an export from qsys and attachment
in Quartus.
*/

module test_ip_top (
  clk,
  reset,

  //==================BEGIN WRITE MASTER====================
  // master inputs and outputs
  master_write_address,
  //master_write_read,
  master_write_write,
  master_write_byteenable,
  //master_write_readdata,
  //master_write_readdatavalid,
  master_write_writedata,
  master_write_burstcount,
  master_write_waitrequest,
  //==================END WRITE MASTER======================

  //==================BEGIN READ MASTER=====================
  // master inputs and outputs
  master_read_address,
  master_read_read,
  //master_read_write,
  master_read_byteenable,
  master_read_readdata,
  master_read_readdatavalid,
  //master_read_writedata,
  master_read_burstcount,
  master_read_waitrequest,
  //==================END READ MASTER=======================

  //==================BEGIN SLAVE===========================
  // signals to connect to an Avalon-MM slave interface
  slave_address,
  slave_read,
  slave_write,
  slave_readdata,
  slave_writedata,
  slave_byteenable,

  // interrupt signals
  slave_irq
  //==================END SLAVE=============================
);

   parameter DATA_WIDTH = 32;
   parameter ADDRESS_WIDTH = 32;

  //==================BEGIN WRITE MASTER PARMS==============
  parameter MASTER_WRITE_MEMORY_BASED_FIFO = 1;            // 0 for LE/ALUT FIFOs, 1 for
      memory FIFOs (highly recommend 1)
  parameter MASTER_WRITE_FIFO_DEPTH = 32;
  parameter MASTER_WRITE_FIFO_DEPTH_LOG2 = 5;
  parameter MASTER_WRITE_BURST_CAPABLE = 0;              // 1 to enable burst, 0 to disable
       it
  parameter MASTER_WRITE_MAXIMUM_BURST_COUNT = 2;
  parameter MASTER_WRITE_BURST_COUNT_WIDTH = 2;
  //==================END WRITE MASTER PARMS================

```

```
66  //==================BEGIN READ MASTER PARMS===============
67  parameter MASTER_READ_MEMORY_BASED_FIFO = 1;              // 0 for LE/ALUT FIFOs, 1 for
        memory FIFOs (highly recommend 1)
68  parameter MASTER_READ_FIFO_DEPTH = 32;
69  parameter MASTER_READ_FIFO_DEPTH_LOG2 = 5;
70  parameter MASTER_READ_BURST_CAPABLE = 0;                // 1 to enable burst, 0 to disable
        it
71  parameter MASTER_READ_MAXIMUM_BURST_COUNT = 2;
72  parameter MASTER_READ_BURST_COUNT_WIDTH = 2;
73  //==================END READ MASTER PARMS=================
74
75  //==================BEGIN SLAVE PARMS=====================
76  parameter ENABLE_SYNC_SIGNALS = 0;  // only used by the component.tcl file, 1 to expose
        user_chipselect/write/read, 0 to stub them
77  parameter IRQ_EN = 0;         // 0 = Enable interrupt, 1 = Disable interrupt
78  //==================END SLAVE PARMS=======================
79
80  input clk;
81  input reset;
82
83  //==================BEGIN WRITE MASTER WIRES==============
84  // control inputs and outputs
85  wire master_write_control_fixed_location;
86  //wire [ADDRESS_WIDTH-1:0] master_write_control_read_base;       // for read master
87  //wire [ADDRESS_WIDTH-1:0] master_write_control_read_length;       // for read master
88  wire [ADDRESS_WIDTH-1:0] master_write_control_write_base;      // for write master
89  wire [ADDRESS_WIDTH-1:0] master_write_control_write_length;      // for write master
90  wire master_write_control_go;
91  /*out*/wire master_write_control_done;
92  ///*out*/wire master_write_control_early_done;              // for read master
93
94  // user logic inputs and outputs
95  //wire master_write_user_read_buffer;                   // for read master
96  wire master_write_user_write_buffer;                  // for write master
97  wire [DATA_WIDTH-1:0] master_write_user_buffer_input_data;      // for write master
98  ///*out*/wire [DATA_WIDTH-1:0] master_write_user_buffer_output_data;  // for read master
99  ///*out*/wire master_write_user_data_available;            // for read master
100 /*out*/wire master_write_user_buffer_full;               // for write master
101
102 // master inputs and outputs
103 output wire [ADDRESS_WIDTH-1:0] master_write_address;
104 //output wire master_write_read;                  // for read master
105 output wire master_write_write;                 // for write master
106 output wire [(DATA_WIDTH/8)-1:0] master_write_byteenable;
107 //input [DATA_WIDTH-1:0] master_write_readdata;          // for read master
108 //input master_write_readdatavalid;                // for read master
109 output wire [DATA_WIDTH-1:0] master_write_writedata;       // for write master
110 output wire [MASTER_WRITE_BURST_COUNT_WIDTH-1:0] master_write_burstcount; // for
        bursting read and write masters
111 input master_write_waitrequest;
112 //==================END WRITE MASTER WIRES================
113
114 //==================BEGIN READ MASTER WIRES===============
115 // control inputs and outputs
116 wire master_read_control_fixed_location;
117 wire [ADDRESS_WIDTH-1:0] master_read_control_read_base;      // for read master
118 wire [ADDRESS_WIDTH-1:0] master_read_control_read_length;      // for read master
119 //wire [ADDRESS_WIDTH-1:0] master_read_control_write_base;       // for write master
120 //wire [ADDRESS_WIDTH-1:0] master_read_control_write_length;       // for write master
121 wire master_read_control_go;
122 /*out*/wire master_read_control_done;
123 /*out*/wire master_read_control_early_done;              // for read master
124
125 // user logic inputs and outputs
126 wire master_read_user_read_buffer;                    // for read master
127 //wire master_read_user_write_buffer;                 // for write master
128 //wire [DATA_WIDTH-1:0] master_read_user_buffer_input_data;      // for write master
129 /*out*/wire [DATA_WIDTH-1:0] master_read_user_buffer_output_data; // for read master
```

```
130   /*out*/wire master_read_user_data_available;              // for read master
131   ///*out*/wire master_read_user_buffer_full;               // for write master
132
133   // master inputs and outputs
134   output wire [ADDRESS_WIDTH-1:0] master_read_address;
135   output wire master_read_read;                  // for read master
136   //output wire master_read_write;                  // for write master
137   output wire [(DATA_WIDTH/8)-1:0] master_read_byteenable;
138   input [DATA_WIDTH-1:0] master_read_readdata;           // for read master
139   input master_read_readdatavalid;                  // for read master
140   //output wire [DATA_WIDTH-1:0] master_read_writedata;      // for write master
141   output wire [MASTER_READ_BURST_COUNT_WIDTH-1:0] master_read_burstcount; // for bursting
          read and write masters
142   input master_read_waitrequest;
143   //==================END READ MASTER WIRES=================
144
145   //==================BEGIN SLAVE WIRES=====================
146   input [8:0] slave_address;
147   input slave_read;
148   input slave_write;
149   output wire [DATA_WIDTH-1:0] slave_readdata;
150   input [DATA_WIDTH-1:0] slave_writedata;
151   input [(DATA_WIDTH/8)-1:0] slave_byteenable;
152   output wire slave_irq;
153
154
155   // user interface
156   /*output*/ wire [DATA_WIDTH-1:0] user_dataout_0;
157   /*output*/ wire [DATA_WIDTH-1:0] user_dataout_1;
158   /*output*/ wire [DATA_WIDTH-1:0] user_dataout_2;
159   /*output*/ wire [DATA_WIDTH-1:0] user_dataout_3;
160   /*output*/ wire [DATA_WIDTH-1:0] user_dataout_4;
161   /*output*/ wire [DATA_WIDTH-1:0] user_dataout_5;
162   wire [DATA_WIDTH-1:0] user_datain_0;
163   wire [DATA_WIDTH-1:0] user_datain_1;
164   wire [DATA_WIDTH-1:0] user_datain_2;
165   wire [DATA_WIDTH-1:0] user_datain_3;
166   wire [DATA_WIDTH-1:0] user_datain_4;
167   wire [DATA_WIDTH-1:0] user_datain_5;
168   /*output*/ wire [15:0] user_chipselect;
169   /*output*/ wire [(DATA_WIDTH/8)-1:0] user_byteenable;
170   /*output*/ wire user_write;
171   /*output*/ wire user_read;
172
173   //==================BEGIN WRITE MASTER MODULE==============
174   custom_master write_master(
175   .clk(clk),
176   .reset(reset),
177
178   // control inputs and outputs
179   .control_fixed_location(master_write_control_fixed_location),
180   .control_read_base(),
181   .control_read_length(),
182   .control_write_base(master_write_control_write_base),
183   .control_write_length(master_write_control_write_length),
184   .control_go(master_write_control_go),
185   .control_done(master_write_control_done),
186   .control_early_done(),
187
188   // user logic inputs and outputs
189   .user_read_buffer(),
190   .user_write_buffer(master_write_user_write_buffer),
191   .user_buffer_input_data(master_write_user_buffer_input_data),
192   .user_buffer_output_data(),
193   .user_data_available(),
194   .user_buffer_full(master_write_user_buffer_full),
195
196   // master inputs and outputs
```

```
197     .master_address(master_write_address),
198     .master_read(),
199     .master_write(master_write_write),
200     .master_byteenable(master_write_byteenable),
201     .master_readdata(),
202     .master_readdatavalid(),
203     .master_writedata(master_write_writedata),
204     .master_burstcount(master_write_burstcount),
205     .master_waitrequest(master_write_waitrequest)
206     );
207
208     defparam write_master.MASTER_DIRECTION = 1;              // 0 for read master, 1 for
            write master
209     defparam write_master.DATA_WIDTH = DATA_WIDTH;
210     defparam write_master.MEMORY_BASED_FIFO = MASTER_WRITE_MEMORY_BASED_FIFO;            // 0
             for LE/ALUT FIFOs, 1 for memory FIFOs (highly recommend 1)
211     defparam write_master.FIFO_DEPTH = MASTER_WRITE_FIFO_DEPTH;
212     defparam write_master.FIFO_DEPTH_LOG2 = MASTER_WRITE_FIFO_DEPTH_LOG2;
213     defparam write_master.ADDRESS_WIDTH = ADDRESS_WIDTH;
214     defparam write_master.BURST_CAPABLE = MASTER_WRITE_BURST_CAPABLE;              // 1 to
            enable burst, 0 to disable it
215     defparam write_master.MAXIMUM_BURST_COUNT = MASTER_WRITE_MAXIMUM_BURST_COUNT;
216     defparam write_master.BURST_COUNT_WIDTH = MASTER_WRITE_BURST_COUNT_WIDTH;
217     //==================END WRITE MASTER MODULE==============
218
219     //==================BEGIN READ MASTER MODULE=============
220     custom_master read_master(
221     .clk(clk),
222     .reset(reset),
223
224     // control inputs and outputs
225     .control_fixed_location(master_read_control_fixed_location),
226     .control_read_base(master_read_control_read_base),
227     .control_read_length(master_read_control_read_length),
228     .control_write_base(),
229     .control_write_length(),
230     .control_go(master_read_control_go),
231     .control_done(master_read_control_done),
232     .control_early_done(master_read_control_early_done),
233
234     // user logic inputs and outputs
235     .user_read_buffer(master_read_user_read_buffer),
236     .user_write_buffer(),
237     .user_buffer_input_data(),
238     .user_buffer_output_data(master_read_user_buffer_output_data),
239     .user_data_available(master_read_user_data_available),
240     .user_buffer_full(),
241
242     // master inputs and outputs
243     .master_address(master_read_address),
244     .master_read(master_read_read),
245     .master_write(),
246     .master_byteenable(master_read_byteenable),
247     .master_readdata(master_read_readdata),
248     .master_readdatavalid(master_read_readdatavalid),
249     .master_writedata(),
250     .master_burstcount(master_read_burstcount),
251     .master_waitrequest(master_read_waitrequest)
252
253     );
254
255     defparam read_master.MASTER_DIRECTION = 0;               // 0 for read master, 1 for
            write master
256     defparam read_master.DATA_WIDTH = DATA_WIDTH;
257     defparam read_master.MEMORY_BASED_FIFO = MASTER_READ_MEMORY_BASED_FIFO;             // 0
            for LE/ALUT FIFOs, 1 for memory FIFOs (highly recommend 1)
258     defparam read_master.FIFO_DEPTH = MASTER_READ_FIFO_DEPTH;
259     defparam read_master.FIFO_DEPTH_LOG2 = MASTER_READ_FIFO_DEPTH_LOG2;
```

```
260   defparam read_master.ADDRESS_WIDTH = ADDRESS_WIDTH;
261   defparam read_master.BURST_CAPABLE = MASTER_READ_BURST_CAPABLE;                 // 1 to
          enable burst, 0 to disable it
262   defparam read_master.MAXIMUM_BURST_COUNT = MASTER_READ_MAXIMUM_BURST_COUNT;
263   defparam read_master.BURST_COUNT_WIDTH = MASTER_READ_BURST_COUNT_WIDTH;
264   //==================END READ MASTER MODULE===============
265
266   //==================BEGIN SLAVE MODULE===================
267   slave_template csr(
268   // signals to connect to an Avalon clock source interface
269   .clk(clk),
270   .reset(reset),
271
272   // signals to connect to an Avalon-MM slave interface
273   .slave_address(slave_address),
274   .slave_read(slave_read),
275   .slave_write(slave_write),
276   .slave_readdata(slave_readdata),
277   .slave_writedata(slave_writedata),
278   .slave_byteenable(slave_byteenable),
279
280   // interrupt signals
281   .slave_irq(slave_irq),
282
283   // signals to connect to custom user logic (up to 16 input and output pairs)
284   .user_dataout_0(user_dataout_0), //address
285   .user_dataout_1(user_dataout_1), //mode
286   .user_dataout_2(user_dataout_2), //data0
287   .user_dataout_3(user_dataout_3), //data1
288   .user_dataout_4(user_dataout_4), //data2
289   .user_dataout_5(user_dataout_5), //data3
290   .user_dataout_6(),
291   .user_dataout_7(),
292   .user_dataout_8(),
293   .user_dataout_9(),
294   .user_dataout_10(),
295   .user_dataout_11(),
296   .user_dataout_12(),
297   .user_dataout_13(),
298   .user_dataout_14(),
299   .user_dataout_15(),
300   .user_datain_0(user_datain_0), //readReady
301   .user_datain_1(user_datain_1), //writeReady
302   .user_datain_2(user_datain_2), //data0
303   .user_datain_3(user_datain_3), //data1
304   .user_datain_4(user_datain_4), //data2
305   .user_datain_5(user_datain_5), //data3
306   .user_datain_6(),
307   .user_datain_7(),
308   .user_datain_8(),
309   .user_datain_9(),
310   .user_datain_10(),
311   .user_datain_11(),
312   .user_datain_12(),
313   .user_datain_13(),
314   .user_datain_14(),
315   .user_datain_15(),
316
317   // optional signals so that your external logic knows what location is being accessed
318   .user_chipselect(user_chipselect),
319   .user_byteenable(user_byteenable),
320   .user_write(user_write),
321   .user_read(user_read)
322 );
323   defparam csr.DATA_WIDTH = DATA_WIDTH;            // word size of each input and output
          register
324   defparam csr.ENABLE_SYNC_SIGNALS = ENABLE_SYNC_SIGNALS;   // only used by the component.
          tcl file, 1 to expose user_chipselect/write/read, 0 to stub them
```

```
325   defparam csr.MODE_0 = 2;                // 0 = Output, 1 = Input, 2 = Ouput and Input, 3
          = Output with loopback, 4 = Disabled
326   defparam csr.MODE_1 = 2;                // 0 = Output, 1 = Input, 2 = Ouput and Input, 3
          = Output with loopback, 4 = Disabled
327   defparam csr.MODE_2 = 2;                // 0 = Output, 1 = Input, 2 = Ouput and Input, 3
          = Output with loopback, 4 = Disabled
328   defparam csr.MODE_3 = 2;                // 0 = Output, 1 = Input, 2 = Ouput and Input, 3
          = Output with loopback, 4 = Disabled
329   defparam csr.MODE_4 = 2;                // 0 = Output, 1 = Input, 2 = Ouput and Input, 3
          = Output with loopback, 4 = Disabled
330   defparam csr.MODE_5 = 2;                // 0 = Output, 1 = Input, 2 = Ouput and Input, 3
          = Output with loopback, 4 = Disabled
331   defparam csr.MODE_6 = 4;                // 0 = Output, 1 = Input, 2 = Ouput and Input, 3
          = Output with loopback, 4 = Disabled
332   defparam csr.MODE_7 = 4;                // 0 = Output, 1 = Input, 2 = Ouput and Input, 3
          = Output with loopback, 4 = Disabled
333   defparam csr.MODE_8 = 4;                // 0 = Output, 1 = Input, 2 = Ouput and Input, 3
          = Output with loopback, 4 = Disabled
334   defparam csr.MODE_9 = 4;                // 0 = Output, 1 = Input, 2 = Ouput and Input, 3
          = Output with loopback, 4 = Disabled
335   defparam csr.MODE_10 = 4;                // 0 = Output, 1 = Input, 2 = Ouput and Input, 3
          = Output with loopback, 4 = Disabled
336   defparam csr.MODE_11 = 4;                // 0 = Output, 1 = Input, 2 = Ouput and Input, 3
          = Output with loopback, 4 = Disabled
337   defparam csr.MODE_12 = 4;                // 0 = Output, 1 = Input, 2 = Ouput and Input, 3
          = Output with loopback, 4 = Disabled
338   defparam csr.MODE_13 = 4;                // 0 = Output, 1 = Input, 2 = Ouput and Input, 3
          = Output with loopback, 4 = Disabled
339   defparam csr.MODE_14 = 4;                // 0 = Output, 1 = Input, 2 = Ouput and Input, 3
          = Output with loopback, 4 = Disabled
340   defparam csr.MODE_15 = 4;                // 0 = Output, 1 = Input, 2 = Ouput and Input, 3
          = Output with loopback, 4 = Disabled
341   defparam csr.IRQ_EN = IRQ_EN;          // 0 = Enable interrupt, 1 = Disable interrupt
342
343 //==================END SLAVE MODULE=====================
344 assign user_datain_1[31:1] = 0;
345 assign user_datain_0[31:1] = 0;
346 //==================BEGIN TEST IP MODULE=================
347   test_ip test_ip1(
348   .clk(clk),
349   .reset(reset),
350
351   //***Base Module Logic I/O***
352
353   //inputs
354   .address(user_dataout_0),
355
356   .writeData0(user_dataout_2),
357   .writeData1(user_dataout_3),
358   .writeData2(user_dataout_4),
359   .writeData3(user_dataout_5),
360
361   .mode(user_dataout_1), //0 for read, 1 for write
362   .modeAccessed(user_chipselect[1]&&user_read),
363
364   //outputs
365   .readData0(user_datain_2),
366   .readData1(user_datain_3),
367   .readData2(user_datain_4),
368   .readData3(user_datain_5),
369
370   .writeReady(user_datain_1[0]),
371   .readReady(user_datain_0[0]),
372
373   //***Memory Logic***
374   //write logic
375   .master_write_control_fixed_location(master_write_control_fixed_location),
376   .master_write_control_write_base(master_write_control_write_base),
```

```
377     .master_write_control_write_length(master_write_control_write_length),
378     .master_write_user_write_buffer(master_write_user_write_buffer),
379     .master_write_user_buffer_input_data(master_write_user_buffer_input_data),
380     .master_write_user_buffer_full(master_write_user_buffer_full),
381     .master_write_user_control_go(master_write_user_control_go),
382     .master_write_user_control_done(master_write_user_control_done),
383
384     //read logic
385     .master_read_control_fixed_location(master_read_control_fixed_location),
386     .master_read_control_read_base(master_read_control_read_base),
387     .master_read_control_read_length(master_read_control_read_length),
388     .master_read_user_read_buffer(master_read_user_read_buffer),
389     .master_read_user_buffer_output_data(master_read_user_buffer_output_data),
390     .master_read_user_data_available(master_read_user_data_available),
391     .master_read_user_control_go(master_read_user_control_go),
392     .master_read_user_control_done(master_read_user_control_done)
393
394 );
395
396 //==================END TEST IP MODULE==================
397 endmodule
```

Listing 16.7: Demo IP Top Verilog File (test_ip_top.v)

16.3.9 Demo IP Top Verilog File (test_ip.v)

```
1  module test_ip(
2    clk,
3    reset,
4
5    //***Base Module Logic I/O***
6
7    //inputs
8    address,
9
10   writeData0,
11   writeData1,
12   writeData2,
13   writeData3,
14
15   mode, //0 for read, 1 for write
16   modeAccessed,
17
18   //outputs
19   readData0,
20   readData1,
21   readData2,
22   readData3,
23
24   writeReady,
25   readReady,
26
27   //***Memory Logic***
28   //write logic
29   master_write_control_fixed_location,
30   master_write_control_write_base,
31   master_write_control_write_length,
32   master_write_user_write_buffer,
33   master_write_user_buffer_input_data,
34   master_write_user_buffer_full,
35   master_write_user_control_go,
36   master_write_user_control_done,
37
38   //read logic
39   master_read_control_fixed_location,
40   master_read_control_read_base,
```

```
41    master_read_control_read_length,
42    master_read_user_read_buffer,
43    master_read_user_buffer_output_data,
44    master_read_user_data_available,
45    master_read_user_control_go,
46    master_read_user_control_done
47
48 );
49
50 input clk;
51 input reset;
52
53 input address;
54
55 input writeData0;
56 input writeData1;
57 input writeData2;
58 input writeData3;
59
60 input mode;
61 input modeAccessed;
62
63 //outputs
64 output reg [31:0] readData0;
65 output reg [31:0] readData1;
66 output reg [31:0] readData2;
67 output reg [31:0] readData3;
68
69 output reg writeReady;
70 output reg readReady;
71
72 //avalon connections
73 output reg master_write_control_fixed_location;
74 output reg [31:0] master_write_control_write_base;
75 output reg [31:0] master_write_control_write_length;
76 output reg master_write_user_write_buffer;
77 output reg [31:0] master_write_user_buffer_input_data;
78 input master_write_user_buffer_full;
79 output reg master_write_user_control_go;
80 input master_write_user_control_done;
81
82 output reg master_read_control_fixed_location;
83 output reg [31:0] master_read_control_read_base;
84 output reg [31:0] master_read_control_read_length;
85 output reg master_read_user_read_buffer;
86 input [31:0] master_read_user_buffer_output_data;
87 input master_read_user_data_available;
88 output reg master_read_user_control_go;
89 input master_read_user_control_done;
90
91 reg [1:0] writeState;
92 // 00 ready
93 // 01 writingToBuffer
94 // 11 writeCommand
95 // 10 waitingForFinish
96 reg [1:0] writeByteNum;
97
98 reg [1:0] readState;
99 // 00 ready/sendCommand
100 // 01 readingFromBuffer
101 // 11 waitingForFinish
102 // 10 UNUSED
103 reg [1:0] readByteNum;
104
105 reg [31:0] writeAddr;
106
107 reg [31:0] writeData0r;
108 reg [31:0] writeData1r;
```

```
109 reg [31:0] writeData2r;
110 reg [31:0] writeData3r;
111
112 initial
113 begin
114 readData0 <= 0;
115 readData1 <= 0;
116 readData2 <= 0;
117 readData3 <= 0;
118
119 writeReady <= 1;
120 readReady <= 1;
121
122 writeState <= 0;
123 writeByteNum <= 0;
124 readState <= 0;
125 readByteNum <= 0 ;
126
127 writeAddr <= 0;
128
129 writeData0r <= 0;
130 writeData1r <= 0;
131 writeData2r <= 0;
132 writeData3r <= 0;
133
134 master_write_user_buffer_input_data <= 0;
135
136 master_write_control_fixed_location <= 0;
137 master_write_control_write_base <= 0;
138 master_write_control_write_length <= 0;
139 master_write_user_write_buffer <= 0;
140 master_write_user_control_go <= 0;
141
142 master_read_control_fixed_location <= 0;
143 master_read_control_read_base <= 0;
144 master_read_control_read_length <= 0;
145 master_read_user_read_buffer <= 0;
146 master_read_user_control_go <= 0;
147 end
148
149 //write function
150 always @ (posedge clk or posedge reset)
151 begin
152   if(reset)
153   begin
154     writeReady <= 1;
155     writeState <= 0;
156     writeByteNum <= 0;
157
158     writeAddr <= 0;
159
160     writeData0r <= 0;
161     writeData1r <= 0;
162     writeData2r <= 0;
163     writeData3r <= 0;
164
165     master_write_user_buffer_input_data <= 0;
166
167     master_write_control_fixed_location <= 0;
168     master_write_control_write_base <= 0;
169     master_write_control_write_length <= 0;
170     master_write_user_write_buffer <= 0;
171     master_write_user_control_go <= 0;
172   end
173   else
174   begin
175     case(writeState)
176       2'b00://ready
```

```
177         begin
178           if(modeAccessed && mode == 1)
179           begin
180             //begin the process of writing
181             writeReady <= 0; //no longer ready to accept new write command
182 
183             writeAddr <= address;
184             writeData0r <= writeData0;
185             writeData1r <= writeData1;
186             writeData2r <= writeData2;
187             writeData3r <= writeData3;
188 
189             writeState <= 2'b01;
190             writeByteNum <= 2'b00;
191           end
192         end
193 
194         2'b01:
195         begin
196           //now writing bytes into fifo buffer
197           case(writeByteNum)
198             2'b00:
199             begin
200               master_write_user_buffer_input_data <= writeData0r;
201               writeByteNum <= 2'b01;
202             end
203             2'b01:
204             begin
205               master_write_user_buffer_input_data <= writeData1r;
206               writeByteNum <= 2'b11;
207             end
208             2'b11:
209             begin
210               master_write_user_buffer_input_data <= writeData2r;
211               writeByteNum <= 2'b10;
212             end
213             2'b10:
214             begin
215               master_write_user_buffer_input_data <= writeData3r;
216               writeByteNum <= 2'b00; //return to 0
217               writeState <= 2'b11;//move to next phase
218             end
219           endcase
220         end
221 
222         2'b11:
223         begin
224           //send command
225           master_write_control_write_base <= writeAddr;
226           master_write_control_write_length <= 32'd16;
227           master_write_user_control_go <= 1;
228           writeState <= 2'b10;
229         end
230 
231         2'b10:
232         begin
233           //send command
234           master_write_user_control_go <= 0;
235           if(master_write_user_control_done == 1)
236           begin
237             writeState <= 2'b00;//done
238             writeReady <= 1;
239           end
240         end
241       endcase
242     end
243 end
244 
```

```
245 //read function
246 always @ (posedge clk)
247 begin
248   if(reset)
249   begin
250     readData0 <= 0;
251     readData1 <= 0;
252     readData2 <= 0;
253     readData3 <= 0;
254
255     readReady <= 1;
256
257     readState <= 0;
258     readByteNum <= 0;
259
260     master_read_control_fixed_location <= 0;
261     master_read_control_read_base <= 0;
262     master_read_control_read_length <= 0;
263     master_read_user_read_buffer <= 0;
264     master_read_user_control_go <= 0;
265   end
266   else
267   begin
268     case(readState)
269       2'b00://ready
270       begin
271         master_read_user_read_buffer <= 0;
272
273         if(modeAccessed && mode == 0)
274         begin
275           //send read command
276           readReady <= 0; //no longer ready to accept new write command
277
278           master_read_control_read_base <= address;
279           master_read_control_read_length <= 32'd16;
280           master_read_user_control_go <= 1;
281
282           readState <= 2'b01;
283           readByteNum <= 2'b00;
284         end
285       end
286
287       2'b01:
288       begin
289         master_read_user_control_go <= 0;
290
291         //now writing bytes into fifo buffer
292         if(master_read_user_data_available == 1)
293         begin
294           case(readByteNum)
295             2'b00:
296             begin
297               readData0 <= master_read_user_buffer_output_data;
298               master_read_user_read_buffer <= 1;
299
300               readByteNum <= 2'b01;
301             end
302             2'b01:
303             begin
304               readData1 <= master_read_user_buffer_output_data;
305               master_read_user_read_buffer <= 1;
306
307               readByteNum <= 2'b11;
308             end
309             2'b11:
310             begin
311               readData2 <= master_read_user_buffer_output_data;
312               master_read_user_read_buffer <= 1;
```

```
313
314               readByteNum <= 2'b10;
315             end
316             2'b10:
317             begin
318               readData3 <= master_read_user_buffer_output_data;
319               master_read_user_read_buffer <= 1;
320
321               readByteNum <= 2'b00;
322               readState <= 2'b10;
323             end
324           endcase
325         end
326         else
327         begin
328           master_read_user_read_buffer <= 0;
329         end
330       end
331
332       2'b10:
333       begin
334         //wait for done
335         master_read_user_control_go <= 0;
336         if(master_read_user_control_done == 1)
337         begin
338           readState <= 2'b00;//done
339           readReady <= 1;
340         end
341       end
342     endcase
343   end
344 end
345
346 endmodule
```

Listing 16.8: Demo IP Verilog File (test_ip.v)

16.3.10 Demo IP _hw.tcl File (IPTest_hw.tcl)

```
1  # TCL File Generated by Component Editor 8.0
2  # Sat May 31 20:41:17 PDT 2008
3  # DO NOT MODIFY
4
5  #Based on the TCL Files for the Avalon Master Template and the Avalon Slave Template by
       Altera Corp.
6
7
8
9  # +-----------------------------------
10 # | module burst_read_master
11 # |
12 set_module_property DESCRIPTION "ARM Mapped IP Test"
13 set_module_property NAME IPTest
14 set_module_property VERSION 1.0
15 set_module_property GROUP "Test IP"
16 set_module_property AUTHOR cyarp
17 set_module_property ICON_PATH SCU_Logo_Seperate_Small.gif
18 set_module_property DISPLAY_NAME "Test IP"
19 set_module_property TOP_LEVEL_HDL_FILE test_ip_top.v
20 set_module_property TOP_LEVEL_HDL_MODULE test_ip_top
21 #set_module_property INSTANTIATE_IN_SYSTEM_MODULE true
22 set_module_property EDITABLE false
23 set_module_property SIMULATION_MODEL_IN_VERILOG false
24 set_module_property SIMULATION_MODEL_IN_VHDL false
25 set_module_property SIMULATION_MODEL_HAS_TULIPS false
26 set_module_property SIMULATION_MODEL_IS_OBFUSCATED false
```

```
27
28 set_module_assignment embeddedsw.dts.compatible "simple-bus"
29
30 # |
31 # +-----------------------------------
32
33
34 set_module_property ELABORATION_CALLBACK     elaborate_me
35 set_module_property VALIDATION_CALLBACK      validate_me
36
37
38 # +-----------------------------------
39 # | files
40 # |
41 add_file test_ip_top.v {SYNTHESIS SIMULATION}
42 add_file test_ip.v {SYNTHESIS SIMULATION}
43 add_file custom_master.v {SYNTHESIS SIMULATION}
44 add_file burst_write_master.v {SYNTHESIS SIMULATION}
45 add_file burst_read_master.v {SYNTHESIS SIMULATION}
46 add_file write_master.v {SYNTHESIS SIMULATION}
47 add_file latency_aware_read_master.v {SYNTHESIS SIMULATION}
48 add_file slave_template.v {SYNTHESIS SIMULATION}
49 add_file interrupt_logic.v {SYNTHESIS SIMULATION}
50 add_file slave_template_macros.h {SYNTHESIS SIMULATION}
51 # |
52 # +-----------------------------------
53
54 # +-----------------------------------
55 # | parameters
56 # |
57
58 # ====================BEGIN GENERAL PARMS======================
59 # Avalon Master Settings
60 add_parameter DATA_WIDTH Integer 32 "Width of the data path"
61 set_parameter_property DATA_WIDTH VISIBLE false
62 set_parameter_property DATA_WIDTH DISPLAY_NAME "Data Width"
63 set_parameter_property DATA_WIDTH GROUP "Avalon-MM Master Properties"
64 set_parameter_property DATA_WIDTH AFFECTS_PORT_WIDTHS true
65 set_parameter_property DATA_WIDTH ALLOWED_RANGES {8 16 32 64 128 256 512 1024}
66
67 add_parameter ADDRESS_WIDTH Integer "32" "Address Width"
68 set_parameter_property ADDRESS_WIDTH VISIBLE false
69 set_parameter_property ADDRESS_WIDTH DISPLAY_NAME "Address Width"
70 set_parameter_property ADDRESS_WIDTH GROUP "Avalon-MM Master Properties"
71 set_parameter_property ADDRESS_WIDTH AFFECTS_PORT_WIDTHS true
72 set_parameter_property ADDRESS_WIDTH ALLOWED_RANGES {32 31 30 29 28 27 26 25 24 23 22 21
       20 19 18 17 16 14 13 12 11 10 9 8 7 6 5 4}
73 # ====================END GNERAL PARMS=========================
74
75 # ====================BEGIN WRITE MASTER PARMS=================
76 # Burst Settings
77 add_parameter MASTER_WRITE_BURST_CAPABLE Integer 0 "Enable bursting"
78 set_parameter_property MASTER_WRITE_BURST_CAPABLE VISIBLE true
79 set_parameter_property MASTER_WRITE_BURST_CAPABLE DISPLAY_NAME "Burst Capable"
80 set_parameter_property MASTER_WRITE_BURST_CAPABLE GROUP "Write Master - Burst Properties"
81 set_parameter_property MASTER_WRITE_BURST_CAPABLE AFFECTS_PORT_WIDTHS true
82 set_parameter_property MASTER_WRITE_BURST_CAPABLE ALLOWED_RANGES {"0:Disabled" "1:Enabled"
       }
83
84 add_parameter MASTER_WRITE_MAXIMUM_BURST_COUNT Integer "2" "Maximum Burst Count"
85 set_parameter_property MASTER_WRITE_MAXIMUM_BURST_COUNT VISIBLE true
86 set_parameter_property MASTER_WRITE_MAXIMUM_BURST_COUNT DISPLAY_NAME "Maximum Burst Count"
87 set_parameter_property MASTER_WRITE_MAXIMUM_BURST_COUNT GROUP "Write Master - Burst
       Properties"
88 set_parameter_property MASTER_WRITE_MAXIMUM_BURST_COUNT AFFECTS_PORT_WIDTHS false
89 set_parameter_property MASTER_WRITE_MAXIMUM_BURST_COUNT ALLOWED_RANGES {1 2 4 8 16 32 64
       128}
90
```

```
91  add_parameter MASTER_WRITE_BURST_COUNT_WIDTH Integer "2" "Enable bursting"
92  set_parameter_property MASTER_WRITE_BURST_COUNT_WIDTH VISIBLE false
93  set_parameter_property MASTER_WRITE_BURST_COUNT_WIDTH DISPLAY_NAME "Burst Count Width"
94  set_parameter_property MASTER_WRITE_BURST_COUNT_WIDTH GROUP "Write Master - Burst
        Properties"
95  set_parameter_property MASTER_WRITE_BURST_COUNT_WIDTH AFFECTS_PORT_WIDTHS true
96  set_parameter_property MASTER_WRITE_BURST_COUNT_WIDTH ALLOWED_RANGES {1:8}
97
98
99  # Other Settings
100 add_parameter MASTER_WRITE_FIFO_DEPTH Integer "32" "FIFO depth"
101 set_parameter_property MASTER_WRITE_FIFO_DEPTH VISIBLE true
102 set_parameter_property MASTER_WRITE_FIFO_DEPTH DISPLAY_NAME "FIFO Depth"
103 set_parameter_property MASTER_WRITE_FIFO_DEPTH GROUP "Write Master - Other Properties"
104 set_parameter_property MASTER_WRITE_FIFO_DEPTH AFFECTS_PORT_WIDTHS false
105 set_parameter_property MASTER_WRITE_FIFO_DEPTH ALLOWED_RANGES {4 8 16 32 64 128 256}
106
107 add_parameter MASTER_WRITE_FIFO_DEPTH_LOG2 Integer "5" "log2(FIFO Depth)"
108 set_parameter_property MASTER_WRITE_FIFO_DEPTH_LOG2 VISIBLE false
109 set_parameter_property MASTER_WRITE_FIFO_DEPTH_LOG2 DISPLAY_NAME "log2(FIFO Depth)"
110 set_parameter_property MASTER_WRITE_FIFO_DEPTH_LOG2 GROUP "Write Master - Other Properties
        "
111 set_parameter_property MASTER_WRITE_FIFO_DEPTH_LOG2 AFFECTS_PORT_WIDTHS false
112 set_parameter_property MASTER_WRITE_FIFO_DEPTH_LOG2 ALLOWED_RANGES {2:8}
113
114 add_parameter MASTER_WRITE_MEMORY_BASED_FIFO Integer 1 "Select false if you want register
        based (0) FIFO instead of memory (1)"
115 set_parameter_property MASTER_WRITE_MEMORY_BASED_FIFO VISIBLE true
116 set_parameter_property MASTER_WRITE_MEMORY_BASED_FIFO DISPLAY_NAME "Memory based FIFO"
117 set_parameter_property MASTER_WRITE_MEMORY_BASED_FIFO GROUP "Write Master - Other
        Properties"
118 set_parameter_property MASTER_WRITE_MEMORY_BASED_FIFO AFFECTS_PORT_WIDTHS false
119 set_parameter_property MASTER_WRITE_MEMORY_BASED_FIFO ALLOWED_RANGES {"1:Memory" "0:Logic"
        }
120 # ==================END WRITE MASTER PARMS====================
121
122 # ==================BEGIN READ MASTER PARMS===================
123 # Burst Settings
124 add_parameter MASTER_READ_BURST_CAPABLE Integer 0 "Enable bursting"
125 set_parameter_property MASTER_READ_BURST_CAPABLE VISIBLE true
126 set_parameter_property MASTER_READ_BURST_CAPABLE DISPLAY_NAME "Burst Capable"
127 set_parameter_property MASTER_READ_BURST_CAPABLE GROUP "Read Master - Burst Properties"
128 set_parameter_property MASTER_READ_BURST_CAPABLE AFFECTS_PORT_WIDTHS true
129 set_parameter_property MASTER_READ_BURST_CAPABLE ALLOWED_RANGES {"0:Disabled" "1:Enabled"}
130
131 add_parameter MASTER_READ_MAXIMUM_BURST_COUNT Integer "2" "Maximum Burst Count"
132 set_parameter_property MASTER_READ_MAXIMUM_BURST_COUNT VISIBLE true
133 set_parameter_property MASTER_READ_MAXIMUM_BURST_COUNT DISPLAY_NAME "Maximum Burst Count"
134 set_parameter_property MASTER_READ_MAXIMUM_BURST_COUNT GROUP "Read Master - Burst
        Properties"
135 set_parameter_property MASTER_READ_MAXIMUM_BURST_COUNT AFFECTS_PORT_WIDTHS false
136 set_parameter_property MASTER_READ_MAXIMUM_BURST_COUNT ALLOWED_RANGES {1 2 4 8 16 32 64
        128}
137
138 add_parameter MASTER_READ_BURST_COUNT_WIDTH Integer "2" "Enable bursting"
139 set_parameter_property MASTER_READ_BURST_COUNT_WIDTH VISIBLE false
140 set_parameter_property MASTER_READ_BURST_COUNT_WIDTH DISPLAY_NAME "Burst Count Width"
141 set_parameter_property MASTER_READ_BURST_COUNT_WIDTH GROUP "Read Master - Burst Properties
        "
142 set_parameter_property MASTER_READ_BURST_COUNT_WIDTH AFFECTS_PORT_WIDTHS true
143 set_parameter_property MASTER_READ_BURST_COUNT_WIDTH ALLOWED_RANGES {1:8}
144
145
146 # Other Settings
147 add_parameter MASTER_READ_FIFO_DEPTH Integer "32" "FIFO depth"
148 set_parameter_property MASTER_READ_FIFO_DEPTH VISIBLE true
149 set_parameter_property MASTER_READ_FIFO_DEPTH DISPLAY_NAME "FIFO Depth"
150 set_parameter_property MASTER_READ_FIFO_DEPTH GROUP "Read Master - Other Properties"
```

```
151 set_parameter_property MASTER_READ_FIFO_DEPTH AFFECTS_PORT_WIDTHS false
152 set_parameter_property MASTER_READ_FIFO_DEPTH ALLOWED_RANGES {4 8 16 32 64 128 256}
153
154 add_parameter MASTER_READ_FIFO_DEPTH_LOG2 Integer "5" "log2(FIFO Depth)"
155 set_parameter_property MASTER_READ_FIFO_DEPTH_LOG2 VISIBLE false
156 set_parameter_property MASTER_READ_FIFO_DEPTH_LOG2 DISPLAY_NAME "log2(FIFO Depth)"
157 set_parameter_property MASTER_READ_FIFO_DEPTH_LOG2 GROUP "Read Master - Other Properties"
158 set_parameter_property MASTER_READ_FIFO_DEPTH_LOG2 AFFECTS_PORT_WIDTHS false
159 set_parameter_property MASTER_READ_FIFO_DEPTH_LOG2 ALLOWED_RANGES {2:8}
160
161 add_parameter MASTER_READ_MEMORY_BASED_FIFO Integer 1 "Select false if you want register
    based (0) FIFO instead of memory (1)"
162 set_parameter_property MASTER_READ_MEMORY_BASED_FIFO VISIBLE true
163 set_parameter_property MASTER_READ_MEMORY_BASED_FIFO DISPLAY_NAME "Memory based FIFO"
164 set_parameter_property MASTER_READ_MEMORY_BASED_FIFO GROUP "Read Master - Other Properties
    "
165 set_parameter_property MASTER_READ_MEMORY_BASED_FIFO AFFECTS_PORT_WIDTHS false
166 set_parameter_property MASTER_READ_MEMORY_BASED_FIFO ALLOWED_RANGES {"1:Memory" "0:Logic"}
167 # ==================END READ MASTER PARMS====================
168
169 # ==================BEGIN SLAVE PARMS========================
170 add_parameter IRQ_EN int 0 "Enable or disable the interrupt capabilities of input ports"
171 set_parameter_property IRQ_EN DISPLAY_NAME "Interrupt capabilities"
172 set_parameter_property IRQ_EN GROUP "CSR - Interrupt"
173 set_parameter_property IRQ_EN AFFECTS_PORT_WIDTHS true
174 set_parameter_property IRQ_EN ALLOWED_RANGES { "0:Disabled" "1:Enabled" }
175 # ==================END SLAVE PARMS==========================
176
177 # |
178 # +-----------------------------------
179
180 # +-----------------------------------
181 # | connection point clk_in
182 # |
183 add_interface clk_in clock end
184 set_interface_property clk_in ptfSchematicName ""
185
186 add_interface_port clk_in clk clk Input 1
187 add_interface_port clk_in reset reset Input 1
188 # |
189 # +-----------------------------------
190
191 # +-----------------------------------
192 # | connection point master_write
193 # |
194 add_interface master_write avalon start
195 set_interface_property master_write linewrapBursts false
196 set_interface_property master_write adaptsTo ""
197 set_interface_property master_write doStreamReads false
198 set_interface_property master_write doStreamWrites false
199 set_interface_property master_write burstOnBurstBoundariesOnly false
200
201 set_interface_property master_write ASSOCIATED_CLOCK clk_in
202
203 add_interface_port master_write master_write_address address Output -1
204 #add_interface_port master_write master_write_read read Output 1
205 add_interface_port master_write master_write_write write Output 1
206 add_interface_port master_write master_write_byteenable byteenable Output -1
207 #add_interface_port master_write master_write_readdata readdata Input -1
208 #add_interface_port master_write master_write_readdatavalid readdatavalid Input 1
209 add_interface_port master_write master_write_writedata writedata Output -1
210 add_interface_port master_write master_write_burstcount burstcount Output -1
211 add_interface_port master_write master_write_waitrequest waitrequest Input 1
212 # |
213 # +-----------------------------------
214
215 # +-----------------------------------
216 # | connection point master_read
```

```
# |
add_interface master_read avalon start
set_interface_property master_read linewrapBursts false
set_interface_property master_read adaptsTo ""
set_interface_property master_read doStreamReads false
set_interface_property master_read doStreamWrites false
set_interface_property master_read burstOnBurstBoundariesOnly false

set_interface_property master_read ASSOCIATED_CLOCK clk_in

add_interface_port master_read master_read_address address Output -1
add_interface_port master_read master_read_read read Output 1
#add_interface_port master_read master_read_write write Output 1
add_interface_port master_read master_read_byteenable byteenable Output -1
add_interface_port master_read master_read_readdata readdata Input -1
add_interface_port master_read master_read_readdatavalid readdatavalid Input 1
#add_interface_port master_read master_read_writedata writedata Output -1
add_interface_port master_read master_read_burstcount burstcount Output -1
add_interface_port master_read master_read_waitrequest waitrequest Input 1
# |
# +-----------------------------------

# +-----------------------------------
# | connection point csr
# |
add_interface csr avalon end
set_interface_property csr holdTime 0
set_interface_property csr linewrapBursts false
set_interface_property csr minimumUninterruptedRunLength 1
set_interface_property csr bridgesToMaster ""
set_interface_property csr isMemoryDevice false
set_interface_property csr burstOnBurstBoundariesOnly false
set_interface_property csr addressSpan 512
set_interface_property csr timingUnits Cycles
set_interface_property csr setupTime 0
set_interface_property csr writeWaitTime 0
set_interface_property csr isNonVolatileStorage false
set_interface_property csr addressAlignment DYNAMIC
set_interface_property csr maximumPendingReadTransactions 0
set_interface_property csr readWaitTime 0
set_interface_property csr readLatency 3
set_interface_property csr printableDevice false

set_interface_property csr ASSOCIATED_CLOCK clk_in

add_interface_port csr slave_address address Input 9
add_interface_port csr slave_read read Input 1
add_interface_port csr slave_write write Input 1
add_interface_port csr slave_readdata readdata Output -1
add_interface_port csr slave_writedata writedata Input -1

# +-----------------------------------
# | connection point csr_irq
# |
add_interface csr_irq interrupt end
set_interface_property csr_irq associatedAddressablePoint csr

set_interface_property csr_irq ASSOCIATED_CLOCK clk_in

add_interface_port csr_irq slave_irq irq Output 1
# |
# |
# +-----------------------------------

proc elaborate_me {}  {

  # set all the new port widths - this is all based
  set the_data_width [get_parameter_value DATA_WIDTH]
```

```
285 set the_address_width [get_parameter_value ADDRESS_WIDTH]
286 set the_byteenable_width [expr {$the_data_width / 8} ]
287
288 # ================BEGIN MASTER WRITE ELABORATE====================
289 set master_write_the_burst_count_width [get_parameter_value
        MASTER_WRITE_BURST_COUNT_WIDTH]
290
291 set_port_property master_write_address WIDTH $the_address_width
292 set_port_property master_write_byteenable WIDTH $the_byteenable_width
293 #set_port_property master_write_readdata WIDTH $the_data_width
294 set_port_property master_write_writedata WIDTH $the_data_width
295 set_port_property master_write_burstcount WIDTH $master_write_the_burst_count_width
296
297
298 # determine the master direction and burst capabilities
299 # write master
300 set master_write_the_burst_capable [get_parameter_value MASTER_WRITE_BURST_CAPABLE]
301
302
303 # switch between read and write master signals (excluding burstcount)
304 #set_port_property master_write_read TERMINATION true
305 set_port_property master_write_write TERMINATION false
306 #set_port_property master_write_readdata TERMINATION true
307 #set_port_property master_write_readdatavalid TERMINATION true
308 set_port_property master_write_writedata TERMINATION false
309
310 # enable/disable the burstcount signal
311 if { $master_write_the_burst_capable == 0 } {
312 set_port_property master_write_burstcount TERMINATION true
313 } else {
314 set_port_property master_write_burstcount TERMINATION false
315 }
316 # ================END MASTER WRITE ELABORATE======================
317
318 # ================BEGIN MASTER READ ELABORATE=====================
319 set master_read_the_burst_count_width [get_parameter_value MASTER_READ_BURST_COUNT_WIDTH
        ]
320
321 set_port_property master_read_address WIDTH $the_address_width
322 set_port_property master_read_byteenable WIDTH $the_byteenable_width
323 set_port_property master_read_readdata WIDTH $the_data_width
324 #set_port_property master_read_writedata WIDTH $the_data_width
325 set_port_property master_read_burstcount WIDTH $master_read_the_burst_count_width
326
327
328 # determine the master direction and burst capabilities
329 # write master
330 set master_read_the_burst_capable [get_parameter_value MASTER_READ_BURST_CAPABLE]
331
332 # switch between read and write master signals (excluding burstcount)
333 set_port_property master_read_read TERMINATION false
334 #set_port_property master_read_write TERMINATION true
335 set_port_property master_read_readdata TERMINATION false
336 set_port_property master_read_readdatavalid TERMINATION false
337 #set_port_property master_read_writedata TERMINATION true
338
339 # enable/disable the burstcount signal
340 if { $master_read_the_burst_capable == 0 } {
341 set_port_property master_read_burstcount TERMINATION true
342 } else {
343 set_port_property master_read_burstcount TERMINATION false
344 }
345 # ================END MASTER READ ELABORATE======================
346
347 # ================BEGIN SLAVE ELABORATE==========================
348 set_port_property slave_readdata WIDTH $the_data_width
349 set_port_property slave_writedata WIDTH $the_data_width
350
```

```
351   set the_irq_en [get_parameter_value IRQ_EN]
352
353     ## adding the slave_byteenable and user_byteenable signals only if the data width is
          greater than 8 bits
354   if { $the_data_width != 8 } {
355     add_interface_port csr slave_byteenable byteenable Input [expr {$the_data_width / 8} ]
356   #if { $the_enable_sync_signals == 1 } {
357   #   add_interface_port user_interface user_byteenable export Output [expr {
          $the_data_width / 8} ]
358   #}
359   }
360
361   ## terminate interrupt signal if it is not turned on
362   expr {($the_irq_en == 1) ? [set_port_property slave_irq TERMINATION false] : [
          set_port_property slave_irq TERMINATION true]};
363   # ================END SLAVE ELABORATE============================
364 }
365
366
367 proc validate_me {}  {
368
369   # ================BEGIN MASTER WRITE VALIDATE====================
370   # read in all the parameter that matter for validation
371   set master_write_the_burst_capable [get_parameter_value MASTER_WRITE_BURST_CAPABLE]
372   set master_write_the_maximum_burst_count [get_parameter_value
          MASTER_WRITE_MAXIMUM_BURST_COUNT]
373   set master_write_the_fifo_depth [get_parameter_value MASTER_WRITE_FIFO_DEPTH]
374
375   # when burst is enabled check to make sure FIFO depth is at least twice as large (also
          enable/disable burst count)
376   if { $master_write_the_burst_capable == 1 } {
377   set_parameter_property MASTER_WRITE_MAXIMUM_BURST_COUNT ENABLED true
378   if { $master_write_the_fifo_depth < [expr {$master_write_the_maximum_burst_count * 2}] }
          {
379     send_message Error "The FIFO Depth must be at least twice as large as Maximum Burst
          Count."
380   }
381   } else {
382   set_parameter_property MASTER_WRITE_MAXIMUM_BURST_COUNT ENABLED false
383   }
384
385
386   set master_write_the_burst_count [get_parameter_value MASTER_WRITE_MAXIMUM_BURST_COUNT]
387   set master_write_the_burst_count_width [expr {(log($master_write_the_burst_count) / log
          (2)) + 1}]
388
389   set master_write_the_fifo_depth [get_parameter_value MASTER_WRITE_FIFO_DEPTH]
390   set master_write_the_fifo_depth_log2 [expr {log($master_write_the_fifo_depth) / log(2)}]
391
392   set_parameter_value MASTER_WRITE_BURST_COUNT_WIDTH $master_write_the_burst_count_width
393   set_parameter_value MASTER_WRITE_FIFO_DEPTH_LOG2 $master_write_the_fifo_depth_log2
394   # ================END MASTER WRITE VALIDATE======================
395
396   # ================BEGIN MASTER READ VALIDATE=====================
397     # read in all the parameter that matter for validation
398   set master_read_the_burst_capable [get_parameter_value MASTER_READ_BURST_CAPABLE]
399   set master_read_the_maximum_burst_count [get_parameter_value
          MASTER_READ_MAXIMUM_BURST_COUNT]
400   set master_read_the_fifo_depth [get_parameter_value MASTER_READ_FIFO_DEPTH]
401
402   # when burst is enabled check to make sure FIFO depth is at least twice as large (also
          enable/disable burst count)
403   if { $master_read_the_burst_capable == 1 } {
404   set_parameter_property MASTER_READ_MAXIMUM_BURST_COUNT ENABLED true
405   if { $master_read_the_fifo_depth < [expr {$master_read_the_maximum_burst_count * 2}] } {
406     send_message Error "The FIFO Depth must be at least twice as large as Maximum Burst
          Count."
407   }
```

```
408   } else {
409   set_parameter_property MASTER_READ_MAXIMUM_BURST_COUNT ENABLED false
410   }
411
412
413   set master_read_the_burst_count [get_parameter_value MASTER_READ_MAXIMUM_BURST_COUNT]
414   set master_read_the_burst_count_width [expr {(log($master_read_the_burst_count) / log(2)
          ) + 1}]
415
416   set master_read_the_fifo_depth [get_parameter_value MASTER_READ_FIFO_DEPTH]
417   set master_read_the_fifo_depth_log2 [expr {log($master_read_the_fifo_depth) / log(2)}]
418
419   set_parameter_value MASTER_READ_BURST_COUNT_WIDTH $master_read_the_burst_count_width
420   set_parameter_value MASTER_READ_FIFO_DEPTH_LOG2 $master_read_the_fifo_depth_log2
421   # ==================END MASTER READ VALIDATE=========================
422 }
```

Listing 16.9: Demo IP _hw.tcl File (IPTest_hw.tcl)

16.3.11 Avalon-MM Custom Master - Altera Template (custom_master.v)

```
1  /*
2  This file is a simple top level that will generate one of four types of
3  Avalon-MM master.  As a result all the ports must be declared and it will be
4  up to the component .tcl file to stub unused signals.
5  */
6
7  // altera message_off 10034
8
9  module custom_master (
10   clk,
11   reset,
12
13   // control inputs and outputs
14   control_fixed_location,
15   control_read_base,
16   control_read_length,
17   control_write_base,
18   control_write_length,
19   control_go,
20   control_done,
21   control_early_done,
22
23   // user logic inputs and outputs
24   user_read_buffer,
25   user_write_buffer,
26   user_buffer_input_data,
27   user_buffer_output_data,
28   user_data_available,
29   user_buffer_full,
30
31   // master inputs and outputs
32   master_address,
33   master_read,
34   master_write,
35   master_byteenable,
36   master_readdata,
37   master_readdatavalid,
38   master_writedata,
39   master_burstcount,
40   master_waitrequest
41 );
42
43   parameter MASTER_DIRECTION = 0;              // 0 for read master, 1 for write master
44   parameter DATA_WIDTH = 32;
```

```
  parameter MEMORY_BASED_FIFO = 1;              // 0 for LE/ALUT FIFOs, 1 for memory FIFOs (highly recommend 1)
  parameter FIFO_DEPTH = 32;
  parameter FIFO_DEPTH_LOG2 = 5;
  parameter ADDRESS_WIDTH = 32;
  parameter BURST_CAPABLE = 0;                  // 1 to enable burst, 0 to disable it
  parameter MAXIMUM_BURST_COUNT = 2;
  parameter BURST_COUNT_WIDTH = 2;


  input clk;
  input reset;

  // control inputs and outputs
  input control_fixed_location;
  input [ADDRESS_WIDTH-1:0] control_read_base;       // for read master
  input [ADDRESS_WIDTH-1:0] control_read_length;      // for read master
  input [ADDRESS_WIDTH-1:0] control_write_base;      // for write master
  input [ADDRESS_WIDTH-1:0] control_write_length;     // for write master
  input control_go;
  output wire control_done;
  output wire control_early_done;               // for read master

  // user logic inputs and outputs
  input user_read_buffer;                   // for read master
  input user_write_buffer;                  // for write master
  input [DATA_WIDTH-1:0] user_buffer_input_data;      // for write master
  output wire [DATA_WIDTH-1:0] user_buffer_output_data; // for read master
  output wire user_data_available;              // for read master
  output wire user_buffer_full;             // for write master

  // master inputs and outputs
  output wire [ADDRESS_WIDTH-1:0] master_address;
  output wire master_read;                  // for read master
  output wire master_write;                 // for write master
  output wire [(DATA_WIDTH/8)-1:0] master_byteenable;
  input [DATA_WIDTH-1:0] master_readdata;           // for read master
  input master_readdatavalid;                   // for read master
  output wire [DATA_WIDTH-1:0] master_writedata;      // for write master
  output wire [BURST_COUNT_WIDTH-1:0] master_burstcount;  // for bursting read and write masters
  input master_waitrequest;


generate  // big generate if statement to select the appropriate master depending on the direction and burst parameters
if(MASTER_DIRECTION == 0)
begin
  if(BURST_CAPABLE == 1)
  begin
    burst_read_master a_burst_read_master(
      .clk (clk),
      .reset (reset),
      .control_fixed_location (control_fixed_location),
      .control_read_base (control_read_base),
      .control_read_length (control_read_length),
      .control_go (control_go),
      .control_done (control_done),
      .control_early_done (control_early_done),
      .user_read_buffer (user_read_buffer),
      .user_buffer_data (user_buffer_output_data),
      .user_data_available (user_data_available),
      .master_address (master_address),
      .master_read (master_read),
      .master_byteenable (master_byteenable),
      .master_readdata (master_readdata),
      .master_readdatavalid (master_readdatavalid),
      .master_burstcount (master_burstcount),
```

```
110         .master_waitrequest (master_waitrequest)
111       );
112       defparam a_burst_read_master.DATAWIDTH = DATA_WIDTH;
113       defparam a_burst_read_master.MAXBURSTCOUNT = MAXIMUM_BURST_COUNT;
114       defparam a_burst_read_master.BURSTCOUNTWIDTH = BURST_COUNT_WIDTH;
115       defparam a_burst_read_master.BYTEENABLEWIDTH = DATA_WIDTH/8;
116       defparam a_burst_read_master.ADDRESSWIDTH = ADDRESS_WIDTH;
117       defparam a_burst_read_master.FIFODEPTH = FIFO_DEPTH;
118       defparam a_burst_read_master.FIFODEPTH_LOG2 = FIFO_DEPTH_LOG2;
119       defparam a_burst_read_master.FIFOUSEMEMORY = MEMORY_BASED_FIFO;
120     end
121     else
122     begin
123       latency_aware_read_master a_latency_aware_read_master(
124         .clk (clk),
125         .reset (reset),
126         .control_fixed_location (control_fixed_location),
127         .control_read_base (control_read_base),
128         .control_read_length (control_read_length),
129         .control_go (control_go),
130         .control_done (control_done),
131         .control_early_done (control_early_done),
132         .user_read_buffer (user_read_buffer),
133         .user_buffer_data (user_buffer_output_data),
134         .user_data_available (user_data_available),
135         .master_address (master_address),
136         .master_read (master_read),
137         .master_byteenable (master_byteenable),
138         .master_readdata (master_readdata),
139         .master_readdatavalid (master_readdatavalid),
140         .master_waitrequest (master_waitrequest)
141       );
142       defparam a_latency_aware_read_master.DATAWIDTH = DATA_WIDTH;
143       defparam a_latency_aware_read_master.BYTEENABLEWIDTH = DATA_WIDTH/8;
144       defparam a_latency_aware_read_master.ADDRESSWIDTH = ADDRESS_WIDTH;
145       defparam a_latency_aware_read_master.FIFODEPTH = FIFO_DEPTH;
146       defparam a_latency_aware_read_master.FIFODEPTH_LOG2 = FIFO_DEPTH_LOG2;
147       defparam a_latency_aware_read_master.FIFOUSEMEMORY = MEMORY_BASED_FIFO;
148     end
149   end
150   else
151   begin
152     if(BURST_CAPABLE == 1)
153     begin
154       burst_write_master a_burst_write_master(
155         .clk (clk),
156         .reset (reset),
157         .control_fixed_location (control_fixed_location),
158         .control_write_base (control_write_base),
159         .control_write_length (control_write_length),
160         .control_go (control_go),
161         .control_done (control_done),
162         .user_write_buffer (user_write_buffer),
163         .user_buffer_data (user_buffer_input_data),
164         .user_buffer_full (user_buffer_full),
165         .master_address (master_address),
166         .master_write (master_write),
167         .master_byteenable (master_byteenable),
168         .master_writedata (master_writedata),
169         .master_burstcount (master_burstcount),
170         .master_waitrequest (master_waitrequest)
171       );
172       defparam a_burst_write_master.DATAWIDTH = DATA_WIDTH;
173       defparam a_burst_write_master.MAXBURSTCOUNT = MAXIMUM_BURST_COUNT;
174       defparam a_burst_write_master.BURSTCOUNTWIDTH = BURST_COUNT_WIDTH;
175       defparam a_burst_write_master.BYTEENABLEWIDTH = DATA_WIDTH/8;
176       defparam a_burst_write_master.ADDRESSWIDTH = ADDRESS_WIDTH;
177       defparam a_burst_write_master.FIFODEPTH = FIFO_DEPTH;
```

```
    defparam a_burst_write_master.FIFODEPTH_LOG2 = FIFO_DEPTH_LOG2;
    defparam a_burst_write_master.FIFOUSEMEMORY = MEMORY_BASED_FIFO;
  end
  else
  begin
    write_master a_write_master(
      .clk (clk),
      .reset (reset),
      .control_fixed_location (control_fixed_location),
      .control_write_base (control_write_base),
      .control_write_length (control_write_length),
      .control_go (control_go),
      .control_done (control_done),
      .user_write_buffer (user_write_buffer),
      .user_buffer_data (user_buffer_input_data),
      .user_buffer_full (user_buffer_full),
      .master_address (master_address),
      .master_write (master_write),
      .master_byteenable (master_byteenable),
      .master_writedata (master_writedata),
      .master_waitrequest (master_waitrequest)
    );
    defparam a_write_master.DATAWIDTH = DATA_WIDTH;
    defparam a_write_master.BYTEENABLEWIDTH = DATA_WIDTH/8;
    defparam a_write_master.ADDRESSWIDTH = ADDRESS_WIDTH;
    defparam a_write_master.FIFODEPTH = FIFO_DEPTH;
    defparam a_write_master.FIFODEPTH_LOG2 = FIFO_DEPTH_LOG2;
    defparam a_write_master.FIFOUSEMEMORY = MEMORY_BASED_FIFO;
  end
end
endgenerate


endmodule
```

Listing 16.10: Avalon-MM Custom Master - Altera Template (custom_master.v)

16.3.12 Avalon-MM Burst Read Master - Altera Template (burst_read_master.v)

```
/*
  Legal Notice: (C)2007 Altera Corporation. All rights reserved.  Your
  use of Altera Corporation's design tools, logic functions and other
  software and tools, and its AMPP partner logic functions, and any
  output files any of the foregoing (including device programming or
  simulation files), and any associated documentation or information are
  expressly subject to the terms and conditions of the Altera Program
  License Subscription Agreement or other applicable license agreement,
  including, without limitation, that your use is for the sole purpose
  of programming logic devices manufactured by Altera and sold by Altera
  or its authorized distributors.  Please refer to the applicable
  agreement for further details.
*/

/*

  Author:  JCJB
  Date:  11/04/2007

  This bursting read master is passed a word aligned address, length in bytes,
  and a 'go' bit.  The master will continue to post full length bursts until
  the length register reaches a value less than a full burst.  A single final
  burst is then posted and when all the reads return the done bit will be asserted.

  To use this master you must simply drive the control signals into this block,
  and also read the data from the exposed read FIFO.  To read from the exposed FIFO
  use the 'user_read_buffer' signal to pop data from the FIFO 'user_buffer_data'.
```

```
28    The signal 'user_data_available' is asserted whenever data is available from the
29    exposed FIFO.
30
31  */
32
33  // altera message_off 10230
34
35
36  module burst_read_master (
37    clk,
38    reset,
39
40    // control inputs and outputs
41    control_fixed_location,
42    control_read_base,
43    control_read_length,
44    control_go,
45    control_done,
46    control_early_done,
47
48    // user logic inputs and outputs
49    user_read_buffer,
50    user_buffer_data,
51    user_data_available,
52
53    // master inputs and outputs
54    master_address,
55    master_read,
56    master_byteenable,
57    master_readdata,
58    master_readdatavalid,
59    master_burstcount,
60    master_waitrequest
61  );
62
63    parameter DATAWIDTH = 32;
64    parameter MAXBURSTCOUNT = 4;
65    parameter BURSTCOUNTWIDTH = 3;
66    parameter BYTEENABLEWIDTH = 4;
67    parameter ADDRESSWIDTH = 32;
68    parameter FIFODEPTH = 32;
69    parameter FIFODEPTH_LOG2 = 5;
70    parameter FIFOUSEMEMORY = 1;  // set to 0 to use LEs instead
71
72    input clk;
73    input reset;
74
75
76    // control inputs and outputs
77    input control_fixed_location;
78    input [ADDRESSWIDTH-1:0] control_read_base;
79    input [ADDRESSWIDTH-1:0] control_read_length;
80    input control_go;
81    output wire control_done;
82    output wire control_early_done;  // don't use this unless you know what you are doing,
          it's going to fire when the last read is posted, not when the last data returns!
83
84    // user logic inputs and outputs
85    input user_read_buffer;
86    output wire [DATAWIDTH-1:0] user_buffer_data;
87    output wire user_data_available;
88
89    // master inputs and outputs
90    input master_waitrequest;
91    input master_readdatavalid;
92      input [DATAWIDTH-1:0] master_readdata;
93    output wire [ADDRESSWIDTH-1:0] master_address;
94    output wire master_read;
```

```
95    output wire [BYTEENABLEWIDTH-1:0] master_byteenable;
96    output wire [BURSTCOUNTWIDTH-1:0] master_burstcount;
97
98    // internal control signals
99    reg control_fixed_location_d1;
100   wire fifo_empty;
101   reg [ADDRESSWIDTH-1:0] address;
102   reg [ADDRESSWIDTH-1:0] length;
103   reg [FIFODEPTH_LOG2-1:0] reads_pending;
104   wire increment_address;
105   wire [BURSTCOUNTWIDTH-1:0] burst_count;
106   wire [BURSTCOUNTWIDTH-1:0] first_short_burst_count;
107   wire first_short_burst_enable;
108   wire [BURSTCOUNTWIDTH-1:0] final_short_burst_count;
109   wire final_short_burst_enable;
110   wire [BURSTCOUNTWIDTH-1:0] burst_boundary_word_address;
111   reg burst_begin;
112   wire too_many_reads_pending;
113   wire [FIFODEPTH_LOG2-1:0] fifo_used;
114
115
116
117   // registering the control_fixed_location bit
118   always @ (posedge clk or posedge reset)
119   begin
120     if (reset == 1)
121     begin
122       control_fixed_location_d1 <= 0;
123     end
124     else
125     begin
126       if (control_go == 1)
127       begin
128         control_fixed_location_d1 <= control_fixed_location;
129       end
130     end
131   end
132
133
134
135   // master address logic
136   always @ (posedge clk or posedge reset)
137   begin
138     if (reset == 1)
139     begin
140       address <= 0;
141     end
142     else
143     begin
144       if(control_go == 1)
145       begin
146         address <= control_read_base;
147       end
148       else if((increment_address == 1) & (control_fixed_location_d1 == 0))
149       begin
150         address <= address + (burst_count * BYTEENABLEWIDTH);  // always performing word
                size accesses, increment by the burst count presented
151       end
152     end
153   end
154
155
156
157   // master length logic
158   always @ (posedge clk or posedge reset)
159   begin
160     if (reset == 1)
161     begin
```

```
162       length <= 0;
163     end
164     else
165     begin
166       if(control_go == 1)
167       begin
168         length <= control_read_length;
169       end
170       else if(increment_address == 1)
171       begin
172         length <= length - (burst_count * BYTEENABLEWIDTH);  // always performing word
                size accesses, decrement by the burst count presented
173       end
174     end
175   end
176
177
178
179   // controlled signals going to the master/control ports
180   assign master_address = address;
181   assign master_byteenable = -1;  // all ones, always performing word size accesses
182   assign master_burstcount = burst_count;
183   assign control_done = (length == 0) & (reads_pending == 0);  // need to make sure that
        the reads have returned before firing the done bit
184   assign control_early_done = (length == 0);  // advanced feature, you should use '
        control_done' if you need all the reads to return first
185   assign master_read = (too_many_reads_pending == 0) & (length != 0);
186   assign burst_boundary_word_address = ((address / BYTEENABLEWIDTH) & (MAXBURSTCOUNT - 1))
        ;
187   assign first_short_burst_enable = (burst_boundary_word_address != 0);
188   assign final_short_burst_enable = (length < (MAXBURSTCOUNT * BYTEENABLEWIDTH));
189
190   assign first_short_burst_count = ((burst_boundary_word_address & 1'b1) == 1'b1)? 1 :  //
        if the burst boundary isn't a multiple of 2 then must post a burst of 1 to get to a
        multiple of 2 for the next burst
191                 (((MAXBURSTCOUNT - burst_boundary_word_address) < (length /
                        BYTEENABLEWIDTH))?
192                 (MAXBURSTCOUNT - burst_boundary_word_address) : (length /
                        BYTEENABLEWIDTH));
193   assign final_short_burst_count = (length / BYTEENABLEWIDTH);
194
195   assign burst_count = (first_short_burst_enable == 1)? first_short_burst_count :  // this
        will get the transfer back on a burst boundary,
196           (final_short_burst_enable == 1)? final_short_burst_count : MAXBURSTCOUNT;
197   assign increment_address = (too_many_reads_pending == 0) & (master_waitrequest == 0) & (
        length != 0);
198   assign too_many_reads_pending = (reads_pending + fifo_used) >= (FIFODEPTH -
        MAXBURSTCOUNT - 4);  // make sure there are fewer reads posted than room in the FIFO
199
200
201   // tracking FIFO
202   always @ (posedge clk or posedge reset)
203   begin
204     if (reset == 1)
205     begin
206       reads_pending <= 0;
207     end
208     else
209     begin
210       if(increment_address == 1)
211       begin
212         if(master_readdatavalid == 0)
213         begin
214           reads_pending <= reads_pending + burst_count;
215         end
216         else
217         begin
```

```
218             reads_pending <= reads_pending + burst_count - 1;  // a burst read was posted,
                    but a word returned
219           end
220         end
221         else
222         begin
223           if(master_readdatavalid == 0)
224           begin
225             reads_pending <= reads_pending;  // burst read was not posted and no read
                    returned
226           end
227           else
228           begin
229             reads_pending <= reads_pending - 1;  // burst read was not posted but a word
                    returned
230           end
231         end
232       end
233     end
234
235
236     // read data feeding user logic
237     assign user_data_available = !fifo_empty;
238     scfifo the_master_to_user_fifo (
239       .aclr (reset),
240       .clock (clk),
241       .data (master_readdata),
242       .empty (fifo_empty),
243       .q (user_buffer_data),
244       .rdreq (user_read_buffer),
245       .usedw (fifo_used),
246       .wrreq (master_readdatavalid)
247     );
248     defparam the_master_to_user_fifo.lpm_width = DATAWIDTH;
249     defparam the_master_to_user_fifo.lpm_numwords = FIFODEPTH;
250     defparam the_master_to_user_fifo.lpm_showahead = "ON";
251     defparam the_master_to_user_fifo.use_eab = (FIFOUSEMEMORY == 1)? "ON" : "OFF";
252     defparam the_master_to_user_fifo.add_ram_output_register = "OFF";
253     defparam the_master_to_user_fifo.underflow_checking = "OFF";
254     defparam the_master_to_user_fifo.overflow_checking = "OFF";
255
256 endmodule
```

Listing 16.11: Avalon-MM Burst Read Master - Altera Template (burst_read_master.v)

16.3.13 Avalon-MM Latency Aware Read Master - Altera Template (latency_aware_read_master.v)

```
1  /*
2    Legal Notice: (C)2007 Altera Corporation. All rights reserved.  Your
3    use of Altera Corporation's design tools, logic functions and other
4    software and tools, and its AMPP partner logic functions, and any
5    output files any of the foregoing (including device programming or
6    simulation files), and any associated documentation or information are
7    expressly subject to the terms and conditions of the Altera Program
8    License Subscription Agreement or other applicable license agreement,
9    including, without limitation, that your use is for the sole purpose
10   of programming logic devices manufactured by Altera and sold by Altera
11   or its authorized distributors.  Please refer to the applicable
12   agreement for further details.
13 */
14
15 /*
16
17   Author:  JCJB
18   Date:  11/04/2007
19
```

```
20   This latency aware read master is passed a word aligned address, length in bytes,
21   and a 'go' bit.  The master will continue to post reads until the length register
22   reaches a value of zero.  When all the reads return the done bit will be asserted.
23
24   To use this master you must simply drive the control signals into this block,
25   and also read the data from the exposed read FIFO.  To read from the exposed FIFO
26   use the 'user_read_buffer' signal to pop data from the FIFO 'user_buffer_data'.
27   The signal 'user_data_available' is asserted whenever data is available from the
28   exposed FIFO.
29
30 */
31
32 // altera message_off 10230
33
34
35 module latency_aware_read_master (
36   clk,
37   reset,
38
39   // control inputs and outputs
40   control_fixed_location,
41   control_read_base,
42   control_read_length,
43   control_go,
44   control_done,
45   control_early_done,
46
47   // user logic inputs and outputs
48   user_read_buffer,
49   user_buffer_data,
50   user_data_available,
51
52   // master inputs and outputs
53   master_address,
54   master_read,
55   master_byteenable,
56   master_readdata,
57   master_readdatavalid,
58   master_waitrequest
59 );
60
61   parameter DATAWIDTH = 32;
62   parameter BYTEENABLEWIDTH = 4;
63   parameter ADDRESSWIDTH = 32;
64   parameter FIFODEPTH = 32;
65   parameter FIFODEPTH_LOG2 = 5;
66   parameter FIFOUSEMEMORY = 1;  // set to 0 to use LEs instead
67
68   input clk;
69   input reset;
70
71
72   // control inputs and outputs
73   input control_fixed_location;
74   input [ADDRESSWIDTH-1:0] control_read_base;
75   input [ADDRESSWIDTH-1:0] control_read_length;
76   input control_go;
77   output wire control_done;
78   output wire control_early_done;  // don't use this unless you know what you are doing!
79
80   // user logic inputs and outputs
81   input user_read_buffer;
82   output wire [DATAWIDTH-1:0] user_buffer_data;
83   output wire user_data_available;
84
85   // master inputs and outputs
86   input master_waitrequest;
87   input master_readdatavalid;
```

```
88      input [DATAWIDTH-1:0] master_readdata;
89    output wire [ADDRESSWIDTH-1:0] master_address;
90    output wire master_read;
91    output wire [BYTEENABLEWIDTH-1:0] master_byteenable;
92
93    // internal control signals
94    reg control_fixed_location_d1;
95    wire fifo_empty;
96    reg [ADDRESSWIDTH-1:0] address;
97    reg [ADDRESSWIDTH-1:0] length;
98    reg [FIFODEPTH_LOG2-1:0] reads_pending;
99    wire increment_address;
100   wire too_many_pending_reads;
101   reg too_many_pending_reads_d1;
102   wire [FIFODEPTH_LOG2-1:0] fifo_used;
103
104
105
106
107   // registering the control_fixed_location bit
108   always @ (posedge clk or posedge reset)
109   begin
110     if (reset == 1)
111     begin
112       control_fixed_location_d1 <= 0;
113     end
114     else
115     begin
116       if (control_go == 1)
117       begin
118         control_fixed_location_d1 <= control_fixed_location;
119       end
120     end
121   end
122
123
124
125   // master address logic
126   assign master_address = address;
127   assign master_byteenable = -1;  // all ones, always performing word size accesses
128   always @ (posedge clk or posedge reset)
129   begin
130     if (reset == 1)
131     begin
132       address <= 0;
133     end
134     else
135     begin
136       if(control_go == 1)
137       begin
138         address <= control_read_base;
139       end
140       else if((increment_address == 1) & (control_fixed_location_d1 == 0))
141       begin
142         address <= address + BYTEENABLEWIDTH;  // always performing word size accesses
143       end
144     end
145   end
146
147
148
149   // master length logic
150   always @ (posedge clk or posedge reset)
151   begin
152     if (reset == 1)
153     begin
154       length <= 0;
155     end
```

```
156     else
157     begin
158       if(control_go == 1)
159       begin
160         length <= control_read_length;
161       end
162       else if(increment_address == 1)
163       begin
164         length <= length - BYTEENABLEWIDTH;  // always performing word size accesses
165       end
166     end
167   end
168
169
170
171   // control logic
172   assign too_many_pending_reads = (fifo_used + reads_pending) >= (FIFODEPTH - 4);
173   assign master_read = (length != 0) & (too_many_pending_reads_d1 == 0);
174   assign increment_address = (length != 0) & (too_many_pending_reads_d1 == 0) & (
          master_waitrequest == 0);
175   assign control_done = (reads_pending == 0) & (length == 0);  // master done posting
          reads and all reads have returned
176   assign control_early_done = (length == 0);  // if you need all the pending reads to
          return then use 'control_done' instead of this signal
177
178
179   always @ (posedge clk)
180   begin
181     if (reset == 1)
182     begin
183       too_many_pending_reads_d1 <= 0;
184     end
185     else
186     begin
187       too_many_pending_reads_d1 <= too_many_pending_reads;
188     end
189   end
190
191
192
193   always @ (posedge clk or posedge reset)
194   begin
195     if (reset == 1)
196     begin
197       reads_pending <= 0;
198     end
199     else
200     begin
201       if(increment_address == 1)
202       begin
203         if(master_readdatavalid == 0)
204         begin
205           reads_pending <= reads_pending + 1;
206         end
207         else
208         begin
209           reads_pending <= reads_pending;  // a read was posted, but another returned
210         end
211       end
212       else
213       begin
214         if(master_readdatavalid == 0)
215         begin
216           reads_pending <= reads_pending;  // read was not posted and no read returned
217         end
218         else
219         begin
220           reads_pending <= reads_pending - 1;  // read was not posted but a read returned
```

```
        end
      end
    end
  end


  // read data feeding user logic
  assign user_data_available = !fifo_empty;
  scfifo the_master_to_user_fifo (
    .aclr (reset),
    .clock (clk),
    .data (master_readdata),
    .empty (fifo_empty),
    .q (user_buffer_data),
    .rdreq (user_read_buffer),
    .usedw (fifo_used),
    .wrreq (master_readdatavalid)
  );
  defparam the_master_to_user_fifo.lpm_width = DATAWIDTH;
  defparam the_master_to_user_fifo.lpm_numwords = FIFODEPTH;
  defparam the_master_to_user_fifo.lpm_showahead = "ON";
  defparam the_master_to_user_fifo.use_eab = (FIFOUSEMEMORY == 1)? "ON" : "OFF";
  defparam the_master_to_user_fifo.add_ram_output_register = "OFF";
  defparam the_master_to_user_fifo.underflow_checking = "OFF";
  defparam the_master_to_user_fifo.overflow_checking = "OFF";

endmodule
```

Listing 16.12: Avalon-MM Latency Aware Read Master - Altera Template (latency_aware_read_master.v)

16.3.14 Avalon-MM Write Master - Altera Template (write_master.v)

```
/*
  Legal Notice: (C)2007 Altera Corporation. All rights reserved.  Your
  use of Altera Corporation's design tools, logic functions and other
  software and tools, and its AMPP partner logic functions, and any
  output files any of the foregoing (including device programming or
  simulation files), and any associated documentation or information are
  expressly subject to the terms and conditions of the Altera Program
  License Subscription Agreement or other applicable license agreement,
  including, without limitation, that your use is for the sole purpose
  of programming logic devices manufactured by Altera and sold by Altera
  or its authorized distributors.  Please refer to the applicable
  agreement for further details.
*/

/*

  Author:  JCJB
  Date:  11/04/2007

  This simple write master is passed a word aligned address, length in bytes,
  and a 'go' bit.  The master will continue to post writes until the length register
  reaches zero.  When the length register reaches zero the 'done' bit is asserted.

  To use this master you must simply drive the control signals into this block,
  and also write the data to the exposed write FIFO.  To read from the exposed FIFO
  use the 'user_write_buffer' signal to push data into the FIFO 'user_buffer_data'.
  The signal 'user_buffer_full' is asserted whenever the exposed buffer is full.
  You should not attempt to write data to the exposed FIFO if it is full.

*/


// altera message_off 10230

```

```
module write_master (
  clk,
  reset,

  // control inputs and outputs
  control_fixed_location,
  control_write_base,
  control_write_length,
  control_go,
  control_done,

  // user logic inputs and outputs
  user_write_buffer,
  user_buffer_data,
  user_buffer_full,

  // master inputs and outputs
  master_address,
  master_write,
  master_byteenable,
  master_writedata,
  master_waitrequest
);


  parameter DATAWIDTH = 32;
  parameter BYTEENABLEWIDTH = 4;
  parameter ADDRESSWIDTH = 32;
  parameter FIFODEPTH = 32;
  parameter FIFODEPTH_LOG2 = 5;
  parameter FIFOUSEMEMORY = 1;  // set to 0 to use LEs instead



  input clk;
  input reset;

  // control inputs and outputs
  input control_fixed_location;  // this only makes sense to enable when MAXBURSTCOUNT = 1
  input [ADDRESSWIDTH-1:0] control_write_base;
  input [ADDRESSWIDTH-1:0] control_write_length;
  input control_go;
  output wire control_done;

  // user logic inputs and outputs
  input user_write_buffer;
  input [DATAWIDTH-1:0] user_buffer_data;
  output wire user_buffer_full;

  // master inputs and outputs
  input master_waitrequest;
  output wire [ADDRESSWIDTH-1:0] master_address;
  output wire master_write;
  output wire [BYTEENABLEWIDTH-1:0] master_byteenable;
  output wire [DATAWIDTH-1:0] master_writedata;


  // internal control signals
  reg control_fixed_location_d1;
  reg [ADDRESSWIDTH-1:0] address;  // this increments for each word
  reg [ADDRESSWIDTH-1:0] length;
  wire increment_address;  // this increments the 'address' register when write is
      asserted and waitrequest is de-asserted
  wire read_fifo;
    wire user_buffer_empty;


```

```
102   // registering the control_fixed_location bit
103   always @ (posedge clk or posedge reset)
104   begin
105     if (reset == 1)
106     begin
107       control_fixed_location_d1 <= 0;
108     end
109     else
110     begin
111       if (control_go == 1)
112       begin
113         control_fixed_location_d1 <= control_fixed_location;
114       end
115     end
116   end
117
118
119
120   // master word increment counter
121   always @ (posedge clk or posedge reset)
122   begin
123     if (reset == 1)
124     begin
125       address <= 0;
126     end
127     else
128     begin
129       if (control_go == 1)
130       begin
131         address <= control_write_base;
132       end
133       else if ((increment_address == 1) & (control_fixed_location_d1 == 0))
134       begin
135         address <= address + BYTEENABLEWIDTH;  // always performing word size accesses
136       end
137     end
138   end
139
140
141   // master length logic
142   always @ (posedge clk or posedge reset)
143   begin
144     if (reset == 1)
145     begin
146       length <= 0;
147     end
148     else
149     begin
150       if (control_go == 1)
151       begin
152         length <= control_write_length;
153       end
154       else if (increment_address == 1)
155       begin
156         length <= length - BYTEENABLEWIDTH;  // always performing word size accesses
157       end
158     end
159   end
160
161
162
163   // controlled signals going to the master/control ports
164   assign master_address = address;
165   assign master_byteenable = -1;  // all ones, always performing word size accesses
166   assign control_done = (length == 0);
167   assign master_write = (user_buffer_empty == 0) & (control_done == 0);
168
```

```
  assign increment_address = (user_buffer_empty == 0) & (master_waitrequest == 0) & (
      control_done == 0);
  assign read_fifo = increment_address;

  // write data feed by user logic
  scfifo the_user_to_master_fifo (
    .aclr (reset),
    .clock (clk),
    .data (user_buffer_data),
    .full (user_buffer_full),
    .empty (user_buffer_empty),
    .q (master_writedata),
    .rdreq (read_fifo),
    .wrreq (user_write_buffer)
  );
  defparam the_user_to_master_fifo.lpm_width = DATAWIDTH;
  defparam the_user_to_master_fifo.lpm_numwords = FIFODEPTH;
  defparam the_user_to_master_fifo.lpm_showahead = "ON";
  defparam the_user_to_master_fifo.use_eab = (FIFOUSEMEMORY == 1)? "ON" : "OFF";
  defparam the_user_to_master_fifo.add_ram_output_register = "OFF";
  defparam the_user_to_master_fifo.underflow_checking = "OFF";
  defparam the_user_to_master_fifo.overflow_checking = "OFF";

endmodule
```

Listing 16.13: Avalon-MM Write Master (write_master.v)

16.3.15 Avalon-MM Burst Write Master (burst_write_master.v)

```
/*
  Legal Notice: (C)2007 Altera Corporation. All rights reserved.  Your
  use of Altera Corporation's design tools, logic functions and other
  software and tools, and its AMPP partner logic functions, and any
  output files any of the foregoing (including device programming or
  simulation files), and any associated documentation or information are
  expressly subject to the terms and conditions of the Altera Program
  License Subscription Agreement or other applicable license agreement,
  including, without limitation, that your use is for the sole purpose
  of programming logic devices manufactured by Altera and sold by Altera
  or its authorized distributors.  Please refer to the applicable
  agreement for further details.
*/

/*

  Author:  JCJB
  Date:  11/04/2007

  This bursting write master is passed a word aligned address, length in bytes,
  and a 'go' bit.  The master will continue to post full length bursts until
  the length register reaches a value less than a full burst.  A single final
  burst is then posted when enough data has been buffered and then the done bit
  will be asserted.

  To use this master you must simply drive the control signals into this block,
  and also write the data to the exposed write FIFO.  To read from the exposed FIFO
  use the 'user_write_buffer' signal to push data into the FIFO 'user_buffer_data'.
  The signal 'user_buffer_full' is asserted whenever the exposed buffer is full.
  You should not attempt to write data to the exposed FIFO if it is full.

*/

// altera message_off 10230


module burst_write_master (
```

```
38    clk,
39    reset,
40
41    // control inputs and outputs
42    control_fixed_location,
43    control_write_base,
44    control_write_length,
45    control_go,
46    control_done,
47
48    // user logic inputs and outputs
49    user_write_buffer,
50    user_buffer_data,
51    user_buffer_full,
52
53    // master inputs and outputs
54    master_address,
55    master_write,
56    master_byteenable,
57    master_writedata,
58    master_burstcount,
59    master_waitrequest
60 );
61
62
63    parameter DATAWIDTH = 32;
64    parameter MAXBURSTCOUNT = 4;
65    parameter BURSTCOUNTWIDTH = 3;
66    parameter BYTEENABLEWIDTH = 4;
67    parameter ADDRESSWIDTH = 32;
68    parameter FIFODEPTH = 32;  // must be at least twice MAXBURSTCOUNT in order to be
          efficient
69    parameter FIFODEPTH_LOG2 = 5;
70    parameter FIFOUSEMEMORY = 1;  // set to 0 to use LEs instead
71
72
73
74    input clk;
75    input reset;
76
77    // control inputs and outputs
78    input control_fixed_location;  // this only makes sense to enable when MAXBURSTCOUNT = 1
79    input [ADDRESSWIDTH-1:0] control_write_base;
80    input [ADDRESSWIDTH-1:0] control_write_length;
81    input control_go;
82    output wire control_done;
83
84    // user logic inputs and outputs
85    input user_write_buffer;
86    input [DATAWIDTH-1:0] user_buffer_data;
87    output wire user_buffer_full;
88
89    // master inputs and outputs
90    input master_waitrequest;
91    output reg [ADDRESSWIDTH-1:0] master_address;
92    output wire master_write;
93    output wire [BYTEENABLEWIDTH-1:0] master_byteenable;
94    output wire [DATAWIDTH-1:0] master_writedata;
95    output reg [BURSTCOUNTWIDTH-1:0] master_burstcount;
96
97    // internal control signals
98    reg control_fixed_location_d1;
99    reg [ADDRESSWIDTH-1:0] length;
100   wire final_short_burst_enable;  // when the length is less than MAXBURSTCOUNT * # of
          bytes per word (BYTEENABLEWIDTH) (i.e. end of the transfer)
101   wire final_short_burst_ready;  // when there is enough data in the FIFO for the final
          burst
```

```
102   wire [BURSTCOUNTWIDTH-1:0] burst_boundary_word_address;  // represents the word offset
          within the burst boundary
103   wire [BURSTCOUNTWIDTH-1:0] first_short_burst_count;
104   wire [BURSTCOUNTWIDTH-1:0] final_short_burst_count;
105   wire first_short_burst_enable;  // when the transfer doesn't start on a burst boundary
106   wire first_short_burst_ready;  // when there is enough data in the FIFO to get the
          master back into burst alignment
107   wire full_burst_ready;  // when there is enough data in the FIFO for a full burst
108   wire increment_address;  // this increments the 'address' register when write is
          asserted and waitrequest is de-asserted
109   wire burst_begin;  // used to register the registers 'burst_address' and 'burst_count_d1
          ' as well as drive the master_address and burst_count muxes
110   wire read_fifo;
111   wire [FIFODEPTH_LOG2-1:0] fifo_used;  // going to combined used with the full bit
112   wire [BURSTCOUNTWIDTH-1:0] burst_count;  // watermark of the FIFO, it has a latency of 2
          cycles
113   reg [BURSTCOUNTWIDTH-1:0] burst_counter;
114   reg first_transfer;  // need to keep track of the first burst so that we don't
          incorrectly increment the address
115
116
117
118   // registering the control_fixed_location bit
119   always @ (posedge clk or posedge reset)
120   begin
121     if (reset == 1)
122     begin
123       control_fixed_location_d1 <= 0;
124     end
125     else
126     begin
127       if (control_go == 1)
128       begin
129         control_fixed_location_d1 <= control_fixed_location;
130       end
131     end
132   end
133
134
135   // set when control_go fires, and reset once the first burst starts
136   always @ (posedge clk or posedge reset)
137   begin
138     if (reset == 1)
139     begin
140       first_transfer <= 0;
141     end
142     else
143     begin
144       if (control_go == 1)
145       begin
146         first_transfer <= 1;
147       end
148       else if (burst_begin == 1)
149       begin
150         first_transfer <= 0;
151       end
152     end
153   end
154
155
156   // master address (held constant during burst)
157   always @ (posedge clk or posedge reset)
158   begin
159     if (reset == 1)
160     begin
161       master_address <= 0;
162     end
163     else
```

```
164     begin
165       if (control_go == 1)
166       begin
167         master_address <= control_write_base;
168       end
169       else if ((first_transfer == 0) & (burst_begin == 1) & (control_fixed_location_d1 ==
              0))
170       begin
171         master_address <= master_address + (master_burstcount * BYTEENABLEWIDTH);  // we
              don't want address + BYTEENABLEWIDTH for the first access
172       end
173     end
174   end
175
176
177   // master length logic
178   always @ (posedge clk or posedge reset)
179   begin
180     if (reset == 1)
181     begin
182       length <= 0;
183     end
184     else
185     begin
186       if (control_go == 1)
187       begin
188         length <= control_write_length;
189       end
190       else if (increment_address == 1)
191       begin
192         length <= length - BYTEENABLEWIDTH;  // always performing word size accesses
193       end
194     end
195   end
196
197
198
199   // register the master burstcount (held constant during burst)
200   always @ (posedge clk or posedge reset)
201   begin
202     if (reset == 1)
203     begin
204       master_burstcount <= 0;
205     end
206     else
207     begin
208       if (burst_begin == 1)
209       begin
210         master_burstcount <= burst_count;
211       end
212     end
213   end
214
215
216
217   // burst counter.  This is set to the burst count being posted then counts down when
          each word
218   // of data goes out.  If it reaches 0 (i.e. not reloaded after 1) then the master stalls
          due to
219   // a lack of data to post a new burst.
220   always @ (posedge clk or posedge reset)
221   begin
222     if (reset == 1)
223     begin
224       burst_counter <= 0;
225     end
226     else
227     begin
```

```
228         if (control_go == 1)
229         begin
230           burst_counter <= 0;
231         end
232         else if (burst_begin == 1)
233         begin
234           burst_counter <= burst_count;
235         end
236         else if (increment_address == 1)  // decrements each write, burst_counter will only
                hit 0 if burst begin doesn't fire on the next cycle
237         begin
238           burst_counter <= burst_counter - 1;
239         end
240       end
241     end
242
243
244
245
246     // burst boundaries are on the master "width * maximum burst count".  The burst boundary
          word address will be used to determine how far off the boundary the transfer starts
          from.
247     assign burst_boundary_word_address = ((master_address / BYTEENABLEWIDTH) & (
          MAXBURSTCOUNT - 1));
248
249     // first short burst enable will only be active on the first transfer (if applicable).
          It will either post the amount of words remaining to reach the end of the burst
250     // boundary or post the remainder of the transfer whichever is shorter.  If the transfer
          is very short and not aligned on a burst boundary then the same logic as the final
          short transfer is used
251     assign first_short_burst_enable = (burst_boundary_word_address != 0) & (first_transfer
          == 1);
252     assign first_short_burst_count = ((burst_boundary_word_address & 1'b1) == 1'b1)? 1 :  //
          if the burst boundary isn't a multiple of 2 then must post a burst of 1 to get to a
          multiple of 2 for the next burst
253                   (((MAXBURSTCOUNT - burst_boundary_word_address) < (length /
                          BYTEENABLEWIDTH))?
254                   (MAXBURSTCOUNT - burst_boundary_word_address) : final_short_burst_count)
                          ;
255     assign first_short_burst_ready = (fifo_used > first_short_burst_count) | ((fifo_used ==
          first_short_burst_count) & (burst_counter == 0));
256
257     // when there isn't enough data for a full burst at the end of the transfer a short
          burst is sent out instead
258     assign final_short_burst_enable = (length < (MAXBURSTCOUNT * BYTEENABLEWIDTH));
259     assign final_short_burst_count = (length/BYTEENABLEWIDTH);
260     assign final_short_burst_ready = (fifo_used > final_short_burst_count) | ((fifo_used ==
          final_short_burst_count) & (burst_counter == 0));  // this will add a one cycle
          stall between bursts, since fifo_used has a cycle of latency, this only affects the
          last burst
261
262     // since the fifo has a latency of 1 we need to make sure we don't under flow
263     assign full_burst_ready = (fifo_used > MAXBURSTCOUNT) | ((fifo_used == MAXBURSTCOUNT) &
          (burst_counter == 0));  // when fifo used watermark equals the burst count the
          statemachine must stall for one cycle, this will make sure that when a burst begins
          there really is enough data present in the FIFO
264
265
266     assign master_byteenable = -1;  // all ones, always performing word size accesses
267     assign control_done = (length == 0);
268     assign master_write = (control_done == 0) & (burst_counter != 0);  // burst_counter = 0
          means the transfer is done, or not enough data in the fifo for a new burst
269
270     // fifo controls and the burst_begin responsible for timing most of the circuit,
          burst_begin starts the writing statemachine
271     assign burst_begin = (((first_short_burst_enable == 1) & (first_short_burst_ready == 1))
272               | ((final_short_burst_enable == 1) & (final_short_burst_ready == 1))
273               | (full_burst_ready == 1))
```

```
            & (control_done == 0)  // since the FIFO can have data before the master
                starts we need to disable this bit from firing when length = 0
            & ((burst_counter == 0) | ((burst_counter == 1) & (master_waitrequest == 0) &
                (length > (MAXBURSTCOUNT * BYTEENABLEWIDTH))));  // need to make a short
                final burst doesn't start right after a full burst completes.

  assign burst_count = (first_short_burst_enable == 1)? first_short_burst_count :  //
      alignment correction gets priority, if the transfer is short and unaligned this will
       cover both
           (final_short_burst_enable == 1)? final_short_burst_count : MAXBURSTCOUNT;

  assign increment_address = (master_write == 1) & (master_waitrequest == 0);  // writing
      is occuring without wait states
  assign read_fifo = increment_address;



  // write data feed by user logic
  scfifo the_user_to_master_fifo (
    .aclr (reset),
    .usedw (fifo_used),
    .clock (clk),
    .data (user_buffer_data),
    .almost_full (user_buffer_full),
    .q (master_writedata),
    .rdreq (read_fifo),
    .wrreq (user_write_buffer)
  );
  defparam the_user_to_master_fifo.lpm_width = DATAWIDTH;
  defparam the_user_to_master_fifo.lpm_numwords = FIFODEPTH;
  defparam the_user_to_master_fifo.lpm_showahead = "ON";
  defparam the_user_to_master_fifo.almost_full_value = (FIFODEPTH - 2);
  defparam the_user_to_master_fifo.use_eab = (FIFOUSEMEMORY == 1)? "ON" : "OFF";
  defparam the_user_to_master_fifo.add_ram_output_register = "OFF";  // makes timing the
      burst begin single simplier
  defparam the_user_to_master_fifo.underflow_checking = "OFF";
  defparam the_user_to_master_fifo.overflow_checking = "OFF";

endmodule
```

Listing 16.14: Avalon-MM Burst Write Master - Altera Template (burst_write_master.v)

16.3.16 Avalon-MM Interrupt Logic - Altera Template (interrupt_logic.v)

```
module interrupt_logic (
  clk,
  reset,
  data_in,
  write,
  write_data,
  address_decode,
  irq_mask_reg_en,
  edge_capture_reg_en,
  read_data,
  irq_out
);

parameter DATA_WIDTH = 32;

input clk;
input reset;
input [DATA_WIDTH-1:0] data_in;
input write;
input [DATA_WIDTH-1:0] write_data;
input address_decode;
input irq_mask_reg_en;
```

```
input edge_capture_reg_en;
output reg [DATA_WIDTH-1:0] read_data;
output wire irq_out;

//internal logic
reg [DATA_WIDTH-1:0] data_in_d1;
reg [DATA_WIDTH-1:0] data_in_d2;
reg [DATA_WIDTH-1:0] data_in_d3;
wire [DATA_WIDTH-1:0] edge_detect;
reg [DATA_WIDTH-1:0] edge_capture;
reg [DATA_WIDTH-1:0] irq_mask;
wire edge_capture_wr_strobe;
wire irq_mask_wr_strobe;
wire [DATA_WIDTH-1:0] readdata_mux;


//interrupt mask register
always @ (posedge clk or posedge reset)
begin
  if (reset == 1)
  begin
    irq_mask <= 0;
  end
  else if (irq_mask_wr_strobe)
  begin
    irq_mask <= write_data;
  end
end

//double registers for asynchronous input and assure metastability
always @ (posedge clk or posedge reset)
begin
  if (reset == 1)
  begin
    data_in_d1 <= 0;
    data_in_d2 <= 0;
  end
  else
  begin
    data_in_d1 <= data_in;
    data_in_d2 <= data_in_d1;
  end
end

//edge detection logic
always @ (posedge clk or posedge reset)
begin
  if (reset == 1)
  begin
    data_in_d3 <= 0;
  end
  else
  begin
    data_in_d3 <= data_in_d2;
  end
end

assign edge_detect = data_in_d2 & ~data_in_d3;

//edge capture registers with separate clear bit
generate
genvar BIT;
for(BIT = 0; BIT < DATA_WIDTH; BIT = BIT + 1)
begin: edge_capture_generation
  always @ (posedge clk or posedge reset)
  begin
    if (reset == 1)
    begin
```

```
91        edge_capture[BIT] <= 0;
92      end
93      else
94    begin
95      if (edge_capture_wr_strobe && write_data[BIT])
96          edge_capture[BIT] <= 0;
97      else if (edge_detect[BIT])
98        edge_capture[BIT] <= 1;
99      end
100   end
101 end
102 endgenerate
103
104 // register the readdata_mux
105 always @ (posedge clk or posedge reset)
106 begin
107   if (reset == 1)
108   begin
109     read_data <= 0;
110   end
111   else
112   begin
113     read_data <= readdata_mux;
114   end
115 end
116
117
118 assign readdata_mux = ({DATA_WIDTH{(irq_mask_reg_en == 1'b1)}} & irq_mask) | ({DATA_WIDTH
      {(edge_capture_reg_en == 1'b1)}} & edge_capture);
119 assign irq_mask_wr_strobe = (write == 1'b1) && (address_decode == 1'b1) && (
      irq_mask_reg_en == 1'b1);
120 assign edge_capture_wr_strobe = (write == 1'b1) && (address_decode == 1'b1) && (
      edge_capture_reg_en == 1'b1);
121 assign irq_out = |(edge_capture & irq_mask);
122
123 endmodule
```

Listing 16.15: Avalon-MM Interrupt Logic - Altera Template (interrupt_logic.v)

16.3.17 Avalon-MM Slave Template - Altera Template (slave_template.v)

```
1  /*
2    Legal Notice: (C)2009 Altera Corporation. All rights reserved.  Your
3    use of Altera Corporation's design tools, logic functions and other
4    software and tools, and its AMPP partner logic functions, and any
5    output files any of the foregoing (including device programming or
6    simulation files), and any associated documentation or information are
7    expressly subject to the terms and conditions of the Altera Program
8    License Subscription Agreement or other applicable license agreement,
9    including, without limitation, that your use is for the sole purpose
10   of programming logic devices manufactured by Altera and sold by Altera
11   or its authorized distributors.  Please refer to the applicable
12   agreement for further details.
13 */
14
15 /*
16
17   This slave component has a parameterizable data width and 16 input/output
18   words.  There are five modes for each addressable word in this component
19   as follows:
20
21   Mode = 0  --> Output only
22   Mode = 1  --> Input only
23   Mode = 2  --> Output and input (independent I/O, default)
24   Mode = 3  --> Output with loopback (software readable output registers)
25   Mode = 4  --> Disabled
```

```
26
27    This component is always available so the waitrequest signal is not used
28    and it has fixed read and write latencies.  The write latency is 0 and the
29    read latency is 3 cycles.  If you attempt to access a location that doesn't
30    support the necessary functionality then you will either write to a
31    non-existent register (write to space) or will readback 0 as the inputs
32    will be grounded if they are disabled.  Inputs or outputs that are removed
33    will be due to the component tcl file stubbing the signals.  Disabled outputs
34    will not be exposed at the top of the system and the Quartus II software
35    will optimize the register away.  Disabled inputs (except in the loopback
36    mode) will not be exposed at the top and internally be wired to ground.
37    The Quartus II software as a result will optimize the input registers to
38    be hardcoded wires set to ground automatically.
39
40    In order for your external logic to know which register is being accessed
41    you will need to enable 'ENABLE_SYNC_SIGNALS' by setting it to 1.  When
42    enabled, the user_chipselect/byteenable/read/write signals will be exposed to your
43    external logic which you can use to determine which register is being accessed
44    and whether it's a read or write access.
45
46    If you use the syncronization signals use the following to qualify them:
47
48    Read:  user_chipselect[x] AND user_read
49    Write:  user_chipselect[x] AND user_write AND user_byteenable
50
51    Note: Reads return the full word regardless of the byteenables presented.
52  */
53
54
55
56
57  module slave_template (
58    // signals to connect to an Avalon clock source interface
59    clk,
60    reset,
61
62    // signals to connect to an Avalon-MM slave interface
63    slave_address,
64    slave_read,
65    slave_write,
66    slave_readdata,
67    slave_writedata,
68    slave_byteenable,
69
70    // interrupt signals
71    slave_irq,
72
73    // signals to connect to custom user logic (up to 16 input and output pairs)
74    user_dataout_0,
75    user_dataout_1,
76    user_dataout_2,
77    user_dataout_3,
78    user_dataout_4,
79    user_dataout_5,
80    user_dataout_6,
81    user_dataout_7,
82    user_dataout_8,
83    user_dataout_9,
84    user_dataout_10,
85    user_dataout_11,
86    user_dataout_12,
87    user_dataout_13,
88    user_dataout_14,
89    user_dataout_15,
90    user_datain_0,
91    user_datain_1,
92    user_datain_2,
93    user_datain_3,
```

```
94    user_datain_4,
95    user_datain_5,
96    user_datain_6,
97    user_datain_7,
98    user_datain_8,
99    user_datain_9,
100   user_datain_10,
101   user_datain_11,
102   user_datain_12,
103   user_datain_13,
104   user_datain_14,
105   user_datain_15,
106
107   // optional signals so that your external logic knows what location is being accessed
108   user_chipselect,
109   user_byteenable,
110   user_write,
111   user_read
112 );
113
114   // most of the set values will only be used by the component.tcl file.  The DATA_WIDTH
          and MODE_X = 3 influence the hardware created.
115   // ENABLE_SYNC_SIGNALS isn't used by this hardware at all but it provided anyway so that
          it can be exposed in the component.tcl file
116   // to control the stubbing of certain signals.
117   parameter DATA_WIDTH = 32;          // word size of each input and output register
118   parameter ENABLE_SYNC_SIGNALS = 0;  // only used by the component.tcl file, 1 to expose
          user_chipselect/write/read, 0 to stub them
119   parameter MODE_0 = 2;               // 0 = Output, 1 = Input, 2 = Ouput and Input, 3 =
          Output with loopback, 4 = Disabled
120   parameter MODE_1 = 2;               // 0 = Output, 1 = Input, 2 = Ouput and Input, 3 =
          Output with loopback, 4 = Disabled
121   parameter MODE_2 = 2;               // 0 = Output, 1 = Input, 2 = Ouput and Input, 3 =
          Output with loopback, 4 = Disabled
122   parameter MODE_3 = 2;               // 0 = Output, 1 = Input, 2 = Ouput and Input, 3 =
          Output with loopback, 4 = Disabled
123   parameter MODE_4 = 2;               // 0 = Output, 1 = Input, 2 = Ouput and Input, 3 =
          Output with loopback, 4 = Disabled
124   parameter MODE_5 = 2;               // 0 = Output, 1 = Input, 2 = Ouput and Input, 3 =
          Output with loopback, 4 = Disabled
125   parameter MODE_6 = 2;               // 0 = Output, 1 = Input, 2 = Ouput and Input, 3 =
          Output with loopback, 4 = Disabled
126   parameter MODE_7 = 2;               // 0 = Output, 1 = Input, 2 = Ouput and Input, 3 =
          Output with loopback, 4 = Disabled
127   parameter MODE_8 = 2;               // 0 = Output, 1 = Input, 2 = Ouput and Input, 3 =
          Output with loopback, 4 = Disabled
128   parameter MODE_9 = 2;               // 0 = Output, 1 = Input, 2 = Ouput and Input, 3 =
          Output with loopback, 4 = Disabled
129   parameter MODE_10 = 2;               // 0 = Output, 1 = Input, 2 = Ouput and Input, 3 =
          Output with loopback, 4 = Disabled
130   parameter MODE_11 = 2;               // 0 = Output, 1 = Input, 2 = Ouput and Input, 3 =
          Output with loopback, 4 = Disabled
131   parameter MODE_12 = 2;               // 0 = Output, 1 = Input, 2 = Ouput and Input, 3 =
          Output with loopback, 4 = Disabled
132   parameter MODE_13 = 2;               // 0 = Output, 1 = Input, 2 = Ouput and Input, 3 =
          Output with loopback, 4 = Disabled
133   parameter MODE_14 = 2;               // 0 = Output, 1 = Input, 2 = Ouput and Input, 3 =
          Output with loopback, 4 = Disabled
134   parameter MODE_15 = 2;               // 0 = Output, 1 = Input, 2 = Ouput and Input, 3 =
          Output with loopback, 4 = Disabled
135   parameter IRQ_EN = 0;         // 0 = Enable interrupt, 1 = Disable interrupt
136
137
138
139   // clock interface
140   input clk;
141   input reset;
142
```

```
143
144   // slave interface
145   input [8:0] slave_address;
146   input slave_read;
147   input slave_write;
148   output wire [DATA_WIDTH-1:0] slave_readdata;
149   input [DATA_WIDTH-1:0] slave_writedata;
150   input [(DATA_WIDTH/8)-1:0] slave_byteenable;
151   output wire slave_irq;
152
153
154   // user interface
155   output wire [DATA_WIDTH-1:0] user_dataout_0;
156   output wire [DATA_WIDTH-1:0] user_dataout_1;
157   output wire [DATA_WIDTH-1:0] user_dataout_2;
158   output wire [DATA_WIDTH-1:0] user_dataout_3;
159   output wire [DATA_WIDTH-1:0] user_dataout_4;
160   output wire [DATA_WIDTH-1:0] user_dataout_5;
161   output wire [DATA_WIDTH-1:0] user_dataout_6;
162   output wire [DATA_WIDTH-1:0] user_dataout_7;
163   output wire [DATA_WIDTH-1:0] user_dataout_8;
164   output wire [DATA_WIDTH-1:0] user_dataout_9;
165   output wire [DATA_WIDTH-1:0] user_dataout_10;
166   output wire [DATA_WIDTH-1:0] user_dataout_11;
167   output wire [DATA_WIDTH-1:0] user_dataout_12;
168   output wire [DATA_WIDTH-1:0] user_dataout_13;
169   output wire [DATA_WIDTH-1:0] user_dataout_14;
170   output wire [DATA_WIDTH-1:0] user_dataout_15;
171   input [DATA_WIDTH-1:0] user_datain_0;
172   input [DATA_WIDTH-1:0] user_datain_1;
173   input [DATA_WIDTH-1:0] user_datain_2;
174   input [DATA_WIDTH-1:0] user_datain_3;
175   input [DATA_WIDTH-1:0] user_datain_4;
176   input [DATA_WIDTH-1:0] user_datain_5;
177   input [DATA_WIDTH-1:0] user_datain_6;
178   input [DATA_WIDTH-1:0] user_datain_7;
179   input [DATA_WIDTH-1:0] user_datain_8;
180   input [DATA_WIDTH-1:0] user_datain_9;
181   input [DATA_WIDTH-1:0] user_datain_10;
182   input [DATA_WIDTH-1:0] user_datain_11;
183   input [DATA_WIDTH-1:0] user_datain_12;
184   input [DATA_WIDTH-1:0] user_datain_13;
185   input [DATA_WIDTH-1:0] user_datain_14;
186   input [DATA_WIDTH-1:0] user_datain_15;
187   output wire [15:0] user_chipselect;
188   output wire [(DATA_WIDTH/8)-1:0] user_byteenable;
189   output wire user_write;
190   output wire user_read;
191
192
193   // internal logic signals
194   wire [(DATA_WIDTH/8)-1:0] internal_byteenable;  // when DATA_WIDTH is 8 bits need to
          hardcode this signal to 1 since the fabric doesn't support 1 bit byte enables
195   reg [(DATA_WIDTH/8)-1:0] internal_byteenable_d1;
196   wire [15:0] address_decode;
197   reg [15:0] address_decode_d1;          // used to select the first stage of mux
          pipelining (a, b, c, or d)
198   wire [3:0] address_bank_decode;        // used to select the second stage of mux
          pipelining (mux a, b, c, or d)
199   reg [3:0] address_bank_decode_d1;      // used to select the second stage of mux
          pipelining (mux a, b, c, or d)
200     reg slave_read_d1;                     // used to qualify the first stage of mux
            pipelining (a, b, c, or d)
201     reg slave_read_d2;                     // used to qualify the second stage of mux
            pipelining (slave_readdata)
202     reg slave_write_d1;                    // used by the option user_write signal
203   reg [DATA_WIDTH-1:0] user_datain_0_d1;
204   reg [DATA_WIDTH-1:0] user_datain_1_d1;
```

```
205   reg [DATA_WIDTH-1:0] user_datain_2_d1;
206   reg [DATA_WIDTH-1:0] user_datain_3_d1;
207   reg [DATA_WIDTH-1:0] user_datain_4_d1;
208   reg [DATA_WIDTH-1:0] user_datain_5_d1;
209   reg [DATA_WIDTH-1:0] user_datain_6_d1;
210   reg [DATA_WIDTH-1:0] user_datain_7_d1;
211   reg [DATA_WIDTH-1:0] user_datain_8_d1;
212   reg [DATA_WIDTH-1:0] user_datain_9_d1;
213   reg [DATA_WIDTH-1:0] user_datain_10_d1;
214   reg [DATA_WIDTH-1:0] user_datain_11_d1;
215   reg [DATA_WIDTH-1:0] user_datain_12_d1;
216   reg [DATA_WIDTH-1:0] user_datain_13_d1;
217   reg [DATA_WIDTH-1:0] user_datain_14_d1;
218   reg [DATA_WIDTH-1:0] user_datain_15_d1;
219   reg [DATA_WIDTH-1:0] mux_first_stage_a;       // muxed inputs 0-3
220   reg [DATA_WIDTH-1:0] mux_first_stage_b;       // muxed inputs 4-7
221   reg [DATA_WIDTH-1:0] mux_first_stage_c;       // muxed inputs 8-11
222   reg [DATA_WIDTH-1:0] mux_first_stage_d;       // muxed inputs 12-15
223   reg [DATA_WIDTH-1:0] int_mux_first_stage_a;       // muxed inputs 0-3
224   reg [DATA_WIDTH-1:0] int_mux_first_stage_b;       // muxed inputs 4-7
225   reg [DATA_WIDTH-1:0] int_mux_first_stage_c;       // muxed inputs 8-11
226   reg [DATA_WIDTH-1:0] int_mux_first_stage_d;       // muxed inputs 12-15
227   reg [DATA_WIDTH-1:0] read_from_user;
228   reg [DATA_WIDTH-1:0] read_from_int_regs;
229   wire [DATA_WIDTH-1:0] readdata_0;
230   wire [DATA_WIDTH-1:0] readdata_1;
231   wire [DATA_WIDTH-1:0] readdata_2;
232   wire [DATA_WIDTH-1:0] readdata_3;
233   wire [DATA_WIDTH-1:0] readdata_4;
234   wire [DATA_WIDTH-1:0] readdata_5;
235   wire [DATA_WIDTH-1:0] readdata_6;
236   wire [DATA_WIDTH-1:0] readdata_7;
237   wire [DATA_WIDTH-1:0] readdata_8;
238   wire [DATA_WIDTH-1:0] readdata_9;
239   wire [DATA_WIDTH-1:0] readdata_10;
240   wire [DATA_WIDTH-1:0] readdata_11;
241   wire [DATA_WIDTH-1:0] readdata_12;
242   wire [DATA_WIDTH-1:0] readdata_13;
243   wire [DATA_WIDTH-1:0] readdata_14;
244   wire [DATA_WIDTH-1:0] readdata_15;
245   wire [15:0] irq_out;
246   wire [DATA_WIDTH-1:0] priority_int_src;
247   wire user_datain_reg_en;
248   wire user_dataout_reg_en;
249   wire irq_mask_reg_en;
250   wire edge_capture_reg_en;
251
252
253   // when the data width is 8 need to hardcode the 1 bit byteenable to high
254   generate
255     if (DATA_WIDTH == 8)
256     begin
257       assign internal_byteenable = 1'b1;
258     end
259     else
260     begin
261       assign internal_byteenable = slave_byteenable;
262     end
263   endgenerate
264
265
266   // sixteen address decodes (using one-hot encoding)  A bank is considered to be a
          grouping of four addresses.
267   assign address_decode[0] = (slave_address[7:4] == 4'b0000) & (slave_write | slave_read);
268   assign address_decode[1] = (slave_address[7:4] == 4'b0001) & (slave_write | slave_read);
269   assign address_decode[2] = (slave_address[7:4] == 4'b0010) & (slave_write | slave_read);
270   assign address_decode[3] = (slave_address[7:4] == 4'b0011) & (slave_write | slave_read);
271   assign address_decode[4] = (slave_address[7:4] == 4'b0100) & (slave_write | slave_read);
```

```
272   assign address_decode[5] = (slave_address[7:4] == 4'b0101) & (slave_write | slave_read);
273   assign address_decode[6] = (slave_address[7:4] == 4'b0110) & (slave_write | slave_read);
274   assign address_decode[7] = (slave_address[7:4] == 4'b0111) & (slave_write | slave_read);
275   assign address_decode[8] = (slave_address[7:4] == 4'b1000) & (slave_write | slave_read);
276   assign address_decode[9] = (slave_address[7:4] == 4'b1001) & (slave_write | slave_read);
277   assign address_decode[10] = (slave_address[7:4] == 4'b1010) & (slave_write | slave_read)
          ;
278   assign address_decode[11] = (slave_address[7:4] == 4'b1011) & (slave_write | slave_read)
          ;
279   assign address_decode[12] = (slave_address[7:4] == 4'b1100) & (slave_write | slave_read)
          ;
280   assign address_decode[13] = (slave_address[7:4] == 4'b1101) & (slave_write | slave_read)
          ;
281   assign address_decode[14] = (slave_address[7:4] == 4'b1110) & (slave_write | slave_read)
          ;
282   assign address_decode[15] = (slave_address[7:4] == 4'b1111) & (slave_write | slave_read)
          ;
283   assign address_bank_decode[0] = (address_decode_d1[3:0] != 0)? 1'b1 : 1'b0;
284   assign address_bank_decode[1] = (address_decode_d1[7:4] != 0)? 1'b1 : 1'b0;
285   assign address_bank_decode[2] = (address_decode_d1[11:8] != 0)? 1'b1 : 1'b0;
286   assign address_bank_decode[3] = (address_decode_d1[15:12] != 0)? 1'b1 : 1'b0;
287   assign user_datain_reg_en  = (slave_address[1:0] == 2'b00);
288   assign user_dataout_reg_en = (slave_address[1:0] == 2'b01);
289   assign irq_mask_reg_en     = (slave_address[1:0] == 2'b10);
290   assign edge_capture_reg_en = (slave_address[1:0] == 2'b11);
291
292
293   // registering various address decoding registers and the input data
294   always @ (posedge clk or posedge reset)
295   begin
296     if (reset == 1)
297     begin
298       slave_read_d1 <= 0;
299       slave_read_d2 <= 0;
300       slave_write_d1 <= 0;
301       address_decode_d1 <= 0;
302       address_bank_decode_d1 <= 0;
303       internal_byteenable_d1 <= 0;
304       user_datain_0_d1 <= 0;
305       user_datain_1_d1 <= 0;
306       user_datain_2_d1 <= 0;
307       user_datain_3_d1 <= 0;
308       user_datain_4_d1 <= 0;
309       user_datain_5_d1 <= 0;
310       user_datain_6_d1 <= 0;
311       user_datain_7_d1 <= 0;
312       user_datain_8_d1 <= 0;
313       user_datain_9_d1 <= 0;
314       user_datain_10_d1 <= 0;
315       user_datain_11_d1 <= 0;
316       user_datain_12_d1 <= 0;
317       user_datain_13_d1 <= 0;
318       user_datain_14_d1 <= 0;
319       user_datain_15_d1 <= 0;
320     end
321     else
322     begin
323       slave_read_d1 <= slave_read;
324       slave_read_d2 <= slave_read_d1;
325       slave_write_d1 <= slave_write;
326       internal_byteenable_d1 <= internal_byteenable;
327       if((slave_read == 1) | (slave_write == 1))
328       begin
329         address_decode_d1 <= address_decode;
330       end
331       if(slave_read_d1 == 1)
332       begin
333         address_bank_decode_d1 <= address_bank_decode;
```

```
334         end
335         if((address_decode[0] == 1) & (slave_read == 1))
336         begin
337           user_datain_0_d1 <= user_datain_0;
338         end
339         if((address_decode[1] == 1) & (slave_read == 1))
340         begin
341           user_datain_1_d1 <= user_datain_1;
342         end
343         if((address_decode[2] == 1) & (slave_read == 1))
344         begin
345           user_datain_2_d1 <= user_datain_2;
346         end
347         if((address_decode[3] == 1) & (slave_read == 1))
348         begin
349           user_datain_3_d1 <= user_datain_3;
350         end
351         if((address_decode[4] == 1) & (slave_read == 1))
352         begin
353           user_datain_4_d1 <= user_datain_4;
354         end
355         if((address_decode[5] == 1) & (slave_read == 1))
356         begin
357           user_datain_5_d1 <= user_datain_5;
358         end
359         if((address_decode[6] == 1) & (slave_read == 1))
360         begin
361           user_datain_6_d1 <= user_datain_6;
362         end
363         if((address_decode[7] == 1) & (slave_read == 1))
364         begin
365           user_datain_7_d1 <= user_datain_7;
366         end
367         if((address_decode[8] == 1) & (slave_read == 1))
368         begin
369           user_datain_8_d1 <= user_datain_8;
370         end
371         if((address_decode[9] == 1) & (slave_read == 1))
372         begin
373           user_datain_9_d1 <= user_datain_9;
374         end
375         if((address_decode[10] == 1) & (slave_read == 1))
376         begin
377           user_datain_10_d1 <= user_datain_10;
378         end
379         if((address_decode[11] == 1) & (slave_read == 1))
380         begin
381           user_datain_11_d1 <= user_datain_11;
382         end
383         if((address_decode[12] == 1) & (slave_read == 1))
384         begin
385           user_datain_12_d1 <= user_datain_12;
386         end
387         if((address_decode[13] == 1) & (slave_read == 1))
388         begin
389           user_datain_13_d1 <= user_datain_13;
390         end
391         if((address_decode[14] == 1) & (slave_read == 1))
392         begin
393           user_datain_14_d1 <= user_datain_14;
394         end
395         if((address_decode[15] == 1) & (slave_read == 1))
396         begin
397           user_datain_15_d1 <= user_datain_15;
398         end
399       end
400     end
401
```

```
402   // Instantiate interrupt logic according to port type chosen and irq_en
403   generate
404     if ((IRQ_EN == 1) && ((MODE_0 == 1) || (MODE_0 == 2)))
405     begin
406       interrupt_logic interrupt_0 (clk, reset, user_datain_0, slave_write, slave_writedata
          , address_decode[0], irq_mask_reg_en, edge_capture_reg_en, readdata_0, irq_out
          [0]);
407       defparam interrupt_0.DATA_WIDTH = DATA_WIDTH;
408     end
409     else
410     begin
411       assign readdata_0 = 0;
412       assign irq_out[0] = 1'b0;
413       end
414
415     if ((IRQ_EN == 1) && ((MODE_1 == 1) || (MODE_1 == 2)))
416     begin
417       interrupt_logic interrupt_1 (clk, reset, user_datain_1, slave_write, slave_writedata
          , address_decode[1],  irq_mask_reg_en, edge_capture_reg_en, readdata_1, irq_out
          [1]);
418       defparam interrupt_1.DATA_WIDTH = DATA_WIDTH;
419     end
420     else
421     begin
422       assign readdata_1 = 0;
423       assign irq_out[1] = 1'b0;
424       end
425
426     if ((IRQ_EN == 1) && ((MODE_2 == 1) || (MODE_2 == 2)))
427     begin
428       interrupt_logic interrupt_2 (clk, reset, user_datain_2, slave_write, slave_writedata
          , address_decode[2], irq_mask_reg_en, edge_capture_reg_en, readdata_2, irq_out
          [2]);
429       defparam interrupt_2.DATA_WIDTH = DATA_WIDTH;
430     end
431     else
432     begin
433       assign readdata_2 = 0;
434       assign irq_out[2] = 1'b0;
435       end
436
437     if ((IRQ_EN == 1) && ((MODE_3 == 1) || (MODE_3 == 2)))
438     begin
439       interrupt_logic interrupt_3 (clk, reset, user_datain_3, slave_write, slave_writedata
          , address_decode[3], irq_mask_reg_en, edge_capture_reg_en, readdata_3, irq_out
          [3]);
440       defparam interrupt_3.DATA_WIDTH = DATA_WIDTH;
441     end
442     else
443     begin
444       assign readdata_3 = 0;
445       assign irq_out[3] = 1'b0;
446       end
447
448     if ((IRQ_EN == 1) && ((MODE_4 == 1) || (MODE_4 == 2)))
449     begin
450       interrupt_logic interrupt_4 (clk, reset, user_datain_4, slave_write, slave_writedata
          , address_decode[4], irq_mask_reg_en, edge_capture_reg_en, readdata_4, irq_out
          [4]);
451       defparam interrupt_4.DATA_WIDTH = DATA_WIDTH;
452     end
453     else
454     begin
455       assign readdata_4 = 0;
456       assign irq_out[4] = 1'b0;
457       end
458
459     if ((IRQ_EN == 1) && ((MODE_5 == 1) || (MODE_5 == 2)))
```

```
460     begin
461       interrupt_logic interrupt_5 (clk, reset, user_datain_5, slave_write, slave_writedata
              , address_decode[5], irq_mask_reg_en, edge_capture_reg_en, readdata_5, irq_out
              [5]);
462       defparam interrupt_5.DATA_WIDTH = DATA_WIDTH;
463     end
464     else
465     begin
466       assign readdata_5 = 0;
467       assign irq_out[5] = 1'b0;
468       end
469
470     if ((IRQ_EN == 1) && ((MODE_6 == 1) || (MODE_6 == 2)))
471     begin
472       interrupt_logic interrupt_6 (clk, reset, user_datain_6, slave_write, slave_writedata
              , address_decode[6], irq_mask_reg_en, edge_capture_reg_en, readdata_6, irq_out
              [6]);
473       defparam interrupt_6.DATA_WIDTH = DATA_WIDTH;
474     end
475     else
476     begin
477       assign readdata_6 = 0;
478       assign irq_out[6] = 1'b0;
479       end
480
481     if ((IRQ_EN == 1) && ((MODE_7 == 1) || (MODE_7 == 2)))
482     begin
483       interrupt_logic interrupt_7 (clk, reset, user_datain_7, slave_write, slave_writedata
              , address_decode[7], irq_mask_reg_en, edge_capture_reg_en, readdata_7, irq_out
              [7]);
484       defparam interrupt_7.DATA_WIDTH = DATA_WIDTH;
485     end
486     else
487     begin
488       assign readdata_7 = 0;
489       assign irq_out[7] = 1'b0;
490       end
491
492     if ((IRQ_EN == 1) && ((MODE_8 == 1) || (MODE_8 == 2)))
493     begin
494       interrupt_logic interrupt_8 (clk, reset, user_datain_8, slave_write, slave_writedata
              , address_decode[8], irq_mask_reg_en, edge_capture_reg_en, readdata_8, irq_out
              [8]);
495       defparam interrupt_8.DATA_WIDTH = DATA_WIDTH;
496     end
497     else
498     begin
499       assign readdata_8 = 0;
500       assign irq_out[8] = 1'b0;
501       end
502
503     if ((IRQ_EN == 1) && ((MODE_9 == 1) || (MODE_9 == 2)))
504     begin
505       interrupt_logic interrupt_9 (clk, reset, user_datain_9, slave_write, slave_writedata
              , address_decode[9], irq_mask_reg_en, edge_capture_reg_en, readdata_9, irq_out
              [9]);
506       defparam interrupt_9.DATA_WIDTH = DATA_WIDTH;
507     end
508     else
509     begin
510       assign readdata_9 = 0;
511       assign irq_out[9] = 1'b0;
512       end
513
514     if ((IRQ_EN == 1) && ((MODE_10 == 1) || (MODE_10 == 2)))
515     begin
516       interrupt_logic interrupt_10 (clk, reset, user_datain_10, slave_write,
              slave_writedata, address_decode[10], irq_mask_reg_en, edge_capture_reg_en,
```

```
        readdata_10, irq_out[10]);
517       defparam interrupt_10.DATA_WIDTH = DATA_WIDTH;
518     end
519     else
520     begin
521       assign readdata_10 = 0;
522       assign irq_out[10] = 1'b0;
523       end
524
525     if ((IRQ_EN == 1) && ((MODE_11 == 1) || (MODE_11 == 2)))
526     begin
527       interrupt_logic interrupt_11 (clk, reset, user_datain_11, slave_write,
            slave_writedata, address_decode[11], irq_mask_reg_en, edge_capture_reg_en,
            readdata_11, irq_out[11]);
528       defparam interrupt_11.DATA_WIDTH = DATA_WIDTH;
529     end
530     else
531     begin
532       assign readdata_11 = 0;
533       assign irq_out[11] = 1'b0;
534       end
535
536     if ((IRQ_EN == 1) && ((MODE_12 == 1) || (MODE_12 == 2)))
537     begin
538       interrupt_logic interrupt_12 (clk, reset, user_datain_12, slave_write,
            slave_writedata, address_decode[12], irq_mask_reg_en, edge_capture_reg_en,
            readdata_12, irq_out[12]);
539       defparam interrupt_12.DATA_WIDTH = DATA_WIDTH;
540     end
541     else
542     begin
543       assign readdata_12 = 0;
544       assign irq_out[12] = 1'b0;
545       end
546
547     if ((IRQ_EN == 1) && ((MODE_13 == 1) || (MODE_13 == 2)))
548     begin
549       interrupt_logic interrupt_13 (clk, reset, user_datain_13, slave_write,
            slave_writedata, address_decode[13], irq_mask_reg_en, edge_capture_reg_en,
            readdata_13, irq_out[13]);
550       defparam interrupt_13.DATA_WIDTH = DATA_WIDTH;
551     end
552     else
553     begin
554       assign readdata_13 = 0;
555       assign irq_out[13] = 1'b0;
556       end
557
558     if ((IRQ_EN == 1) && ((MODE_14 == 1) || (MODE_14 == 2)))
559     begin
560       interrupt_logic interrupt_14 (clk, reset, user_datain_14, slave_write,
            slave_writedata, address_decode[14], irq_mask_reg_en, edge_capture_reg_en,
            readdata_14, irq_out[14]);
561       defparam interrupt_14.DATA_WIDTH = DATA_WIDTH;
562     end
563     else
564     begin
565       assign readdata_14 = 0;
566       assign irq_out[14] = 1'b0;
567       end
568
569     if ((IRQ_EN == 1) && ((MODE_15 == 1) || (MODE_15 == 2)))
570     begin
571       interrupt_logic interrupt_15 (clk, reset, user_datain_15, slave_write,
            slave_writedata, address_decode[15], irq_mask_reg_en, edge_capture_reg_en,
            readdata_15, irq_out[15]);
572       defparam interrupt_15.DATA_WIDTH = DATA_WIDTH;
573     end
```

```
574     else
575     begin
576       assign readdata_15 = 0;
577       assign irq_out[15] = 1'b0;
578       end
579
580   endgenerate
581
582   // sixteen output registers which use byteenables to register each byte.  Disabling the
          write enable for output registers that were not needed.
583   register_with_bytelanes register_0 (clk,reset,slave_writedata,slave_write&address_decode
          [0]&user_dataout_reg_en,internal_byteenable,user_dataout_0);
584     defparam register_0.DATA_WIDTH = DATA_WIDTH;
585   register_with_bytelanes register_1 (clk,reset,slave_writedata,slave_write&address_decode
          [1]&user_dataout_reg_en,internal_byteenable,user_dataout_1);
586     defparam register_1.DATA_WIDTH = DATA_WIDTH;
587   register_with_bytelanes register_2 (clk,reset,slave_writedata,slave_write&address_decode
          [2]&user_dataout_reg_en,internal_byteenable,user_dataout_2);
588     defparam register_2.DATA_WIDTH = DATA_WIDTH;
589   register_with_bytelanes register_3 (clk,reset,slave_writedata,slave_write&address_decode
          [3]&user_dataout_reg_en,internal_byteenable,user_dataout_3);
590     defparam register_3.DATA_WIDTH = DATA_WIDTH;
591   register_with_bytelanes register_4 (clk,reset,slave_writedata,slave_write&address_decode
          [4]&user_dataout_reg_en,internal_byteenable,user_dataout_4);
592     defparam register_4.DATA_WIDTH = DATA_WIDTH;
593   register_with_bytelanes register_5 (clk,reset,slave_writedata,slave_write&address_decode
          [5]&user_dataout_reg_en,internal_byteenable,user_dataout_5);
594     defparam register_5.DATA_WIDTH = DATA_WIDTH;
595   register_with_bytelanes register_6 (clk,reset,slave_writedata,slave_write&address_decode
          [6]&user_dataout_reg_en,internal_byteenable,user_dataout_6);
596     defparam register_6.DATA_WIDTH = DATA_WIDTH;
597   register_with_bytelanes register_7 (clk,reset,slave_writedata,slave_write&address_decode
          [7]&user_dataout_reg_en,internal_byteenable,user_dataout_7);
598     defparam register_7.DATA_WIDTH = DATA_WIDTH;
599   register_with_bytelanes register_8 (clk,reset,slave_writedata,slave_write&address_decode
          [8]&user_dataout_reg_en,internal_byteenable,user_dataout_8);
600     defparam register_8.DATA_WIDTH = DATA_WIDTH;
601   register_with_bytelanes register_9 (clk,reset,slave_writedata,slave_write&address_decode
          [9]&user_dataout_reg_en,internal_byteenable,user_dataout_9);
602     defparam register_9.DATA_WIDTH = DATA_WIDTH;
603   register_with_bytelanes register_10 (clk,reset,slave_writedata,slave_write&
          address_decode[10]&user_dataout_reg_en,internal_byteenable,user_dataout_10);
604     defparam register_10.DATA_WIDTH = DATA_WIDTH;
605   register_with_bytelanes register_11 (clk,reset,slave_writedata,slave_write&
          address_decode[11]&user_dataout_reg_en,internal_byteenable,user_dataout_11);
606     defparam register_11.DATA_WIDTH = DATA_WIDTH;
607   register_with_bytelanes register_12 (clk,reset,slave_writedata,slave_write&
          address_decode[12]&user_dataout_reg_en,internal_byteenable,user_dataout_12);
608     defparam register_12.DATA_WIDTH = DATA_WIDTH;
609   register_with_bytelanes register_13 (clk,reset,slave_writedata,slave_write&
          address_decode[13]&user_dataout_reg_en,internal_byteenable,user_dataout_13);
610     defparam register_13.DATA_WIDTH = DATA_WIDTH;
611   register_with_bytelanes register_14 (clk,reset,slave_writedata,slave_write&
          address_decode[14]&user_dataout_reg_en,internal_byteenable,user_dataout_14);
612     defparam register_14.DATA_WIDTH = DATA_WIDTH;
613   register_with_bytelanes register_15 (clk,reset,slave_writedata,slave_write&
          address_decode[15]&user_dataout_reg_en,internal_byteenable,user_dataout_15);
614     defparam register_15.DATA_WIDTH = DATA_WIDTH;
615
616
617   // registered slave_readdata mux for all the return values, if an input register is
          disabled then a zero will be passed in
618   always @ (posedge clk or posedge reset)
619   begin
620     if (reset == 1)
621     begin
622       read_from_user <= 0;
623     end
```

```
624     else
625     begin
626       if (slave_read_d1 == 1)  // first multiplexer stage
627       begin
628         // when the mode parameters are set to 3 then the outputs are looped back as
                inputs
629         case (address_decode_d1[3:0])
630           4'b0001: mux_first_stage_a <= (MODE_0 == 3)? user_dataout_0 : user_datain_0_d1;
631           4'b0010: mux_first_stage_a <= (MODE_1 == 3)? user_dataout_1 : user_datain_1_d1;
632           4'b0100: mux_first_stage_a <= (MODE_2 == 3)? user_dataout_2 : user_datain_2_d1;
633           4'b1000: mux_first_stage_a <= (MODE_3 == 3)? user_dataout_3 : user_datain_3_d1;
634         endcase
635         case (address_decode_d1[7:4])
636           4'b0001: mux_first_stage_b <= (MODE_4 == 3)? user_dataout_4 : user_datain_4_d1;
637           4'b0010: mux_first_stage_b <= (MODE_5 == 3)? user_dataout_5 : user_datain_5_d1;
638           4'b0100: mux_first_stage_b <= (MODE_6 == 3)? user_dataout_6 : user_datain_6_d1;
639           4'b1000: mux_first_stage_b <= (MODE_7 == 3)? user_dataout_7 : user_datain_7_d1;
640         endcase
641         case (address_decode_d1[11:8])
642           4'b0001: mux_first_stage_c <= (MODE_8 == 3)? user_dataout_8 : user_datain_8_d1;
643           4'b0010: mux_first_stage_c <= (MODE_9 == 3)? user_dataout_9 : user_datain_9_d1;
644           4'b0100: mux_first_stage_c <= (MODE_10 == 3)? user_dataout_10 :
                  user_datain_10_d1;
645           4'b1000: mux_first_stage_c <= (MODE_11 == 3)? user_dataout_11 :
                  user_datain_11_d1;
646         endcase
647         case (address_decode_d1[15:12])
648           4'b0001: mux_first_stage_d <= (MODE_12 == 3)? user_dataout_12 :
                  user_datain_12_d1;
649           4'b0010: mux_first_stage_d <= (MODE_13 == 3)? user_dataout_13 :
                  user_datain_13_d1;
650           4'b0100: mux_first_stage_d <= (MODE_14 == 3)? user_dataout_14 :
                  user_datain_14_d1;
651           4'b1000: mux_first_stage_d <= (MODE_15 == 3)? user_dataout_15 :
                  user_datain_15_d1;
652         endcase
653       end
654
655       if (slave_read_d2 == 1)  // second multiplexer stage
656       begin
657         case (address_bank_decode_d1[3:0])
658           4'b0001: read_from_user <= mux_first_stage_a;
659           4'b0010: read_from_user <= mux_first_stage_b;
660           4'b0100: read_from_user <= mux_first_stage_c;
661           4'b1000: read_from_user <= mux_first_stage_d;
662         endcase
663       end
664     end
665   end
666
667   // multiplex the readdata from all I/O interrupt registers
668   always @ (posedge clk or posedge reset)
669   begin
670     if (reset == 1)
671     begin
672       read_from_int_regs <= 0;
673     end
674     else
675     begin
676       if (slave_read_d1 == 1)  // first multiplexer stage
677       begin
678         // when the mode parameters are set to 3 then the outputs are looped back as
                inputs
679         case (address_decode_d1[3:0])
680           4'b0001: int_mux_first_stage_a <= readdata_0;
681           4'b0010: int_mux_first_stage_a <= readdata_1;
682           4'b0100: int_mux_first_stage_a <= readdata_2;
683           4'b1000: int_mux_first_stage_a <= readdata_3;
```

```
684         endcase
685         case (address_decode_d1[7:4])
686           4'b0001: int_mux_first_stage_b <= readdata_4;
687           4'b0010: int_mux_first_stage_b <= readdata_5;
688           4'b0100: int_mux_first_stage_b <= readdata_6;
689           4'b1000: int_mux_first_stage_b <= readdata_7;
690         endcase
691         case (address_decode_d1[11:8])
692           4'b0001: int_mux_first_stage_c <= readdata_8;
693           4'b0010: int_mux_first_stage_c <= readdata_9;
694           4'b0100: int_mux_first_stage_c <= readdata_10;
695           4'b1000: int_mux_first_stage_c <= readdata_11;
696         endcase
697         case (address_decode_d1[15:12])
698           4'b0001: int_mux_first_stage_d <= readdata_12;
699           4'b0010: int_mux_first_stage_d <= readdata_13;
700           4'b0100: int_mux_first_stage_d <= readdata_14;
701           4'b1000: int_mux_first_stage_d <= readdata_15;
702         endcase
703       end
704
705       if (slave_read_d2 == 1)  // second multiplexer stage
706       begin
707         case (address_bank_decode_d1[3:0])
708           4'b0001: read_from_int_regs <= int_mux_first_stage_a;
709           4'b0010: read_from_int_regs <= int_mux_first_stage_b;
710           4'b0100: read_from_int_regs <= int_mux_first_stage_c;
711           4'b1000: read_from_int_regs <= int_mux_first_stage_d;
712         endcase
713       end
714     end
715   end
716
717     assign priority_int_src = (irq_out[0] == 1) ? 1 :
718               (irq_out[1] == 1) ? 2 :
719               (irq_out[2] == 1) ? 3 :
720               (irq_out[3] == 1) ? 4 :
721               (irq_out[4] == 1) ? 5 :
722               (irq_out[5] == 1) ? 6 :
723               (irq_out[6] == 1) ? 7 :
724               (irq_out[7] == 1) ? 8 :
725               (irq_out[8] == 1) ? 9 :
726               (irq_out[9] == 1) ? 10 :
727               (irq_out[10] == 1) ? 11 :
728               (irq_out[11] == 1) ? 12 :
729               (irq_out[12] == 1) ? 13 :
730               (irq_out[13] == 1) ? 14 :
731               (irq_out[14] == 1) ? 15 :
732               (irq_out[15] == 1) ? 16 : 0;
733
734   assign user_write = slave_write_d1;  // outputs are registed so need a delayed copy of
        the write signal
735   assign user_read = slave_read;
736   assign user_chipselect = (slave_write_d1 == 1)? address_decode_d1 : address_decode;  //
        for write cycles need the delayed copy of the address decode since outputs are
        registered
737   assign user_byteenable = (slave_write_d1 == 1)? internal_byteenable_d1 :
        internal_byteenable;  // for write cycles need the delayed copy of the byteenables,
        don't use the byteenables for reads since the full word is always sent on read
        transfers
738   assign slave_readdata = (slave_address == 9'h100)? priority_int_src : (
        user_datain_reg_en == 1)? read_from_user : read_from_int_regs;
739   assign slave_irq = | (irq_out);
740
741 endmodule
742
743
744
```

```
// helper module to simplify having a register of variable width and containing independent byte lanes
module register_with_bytelanes (
  clk,
  reset,

  data_in,
  write,
  byte_enables,
  data_out
);

  parameter DATA_WIDTH = 32;

  input clk;
  input reset;

  input [DATA_WIDTH-1:0] data_in;
  input write;
  input [(DATA_WIDTH/8)-1:0] byte_enables;
  output reg [DATA_WIDTH-1:0] data_out;

  // generating write logic for each group of 8 bits for 'data_out'
  generate
  genvar LANE;
    for(LANE = 0; LANE < (DATA_WIDTH/8); LANE = LANE+1)
    begin: register_bytelane_generation
      always @ (posedge clk or posedge reset)
      begin
        if(reset == 1)
        begin
          data_out[(LANE*8)+7:(LANE*8)] <= 0;
        end
        else
        begin
          if((byte_enables[LANE] == 1) & (write == 1))
          begin
            data_out[(LANE*8)+7:(LANE*8)] <= data_in[(LANE*8)+7:(LANE*8)];  // write to each byte lane with write = 1 and the lane byteenable = 1
          end
        end
      end
    end
  endgenerate

endmodule
```

Listing 16.16: Avalon-MM Slave Template - Altera Template (slave_template.v)

16.3.18 Avalon-MM Slave Template Macros - Altera Template (slave_template_macros.h)

```
/**************************************************
 * These are all word addresses. Please shift the
 * offset accordingly when using them in IORD or IOWR.
 * E.g.: Write to a 32-bit user_datain_0 port
 *   IOWR_32DIRECT(BASE, DATA_IN_0 * 4, 0XFFFFFFFF)
 * Another example: Read a 16-bit edge capture reg
 *   IORD_16DIRECT(BASE, EDGE_CAPTURE_0 * 2)
 *
**************************************************/

#define DATA_IN_0 0x00
#define DATA_IN_1 0x10
#define DATA_IN_2 0x20
#define DATA_IN_3 0x30
#define DATA_IN_4 0x40
```

```
16 #define DATA_IN_5 0x50
17 #define DATA_IN_6 0x60
18 #define DATA_IN_7 0x70
19 #define DATA_IN_8 0x80
20 #define DATA_IN_9 0x90
21 #define DATA_IN_10 0xA0
22 #define DATA_IN_11 0xB0
23 #define DATA_IN_12 0xC0
24 #define DATA_IN_13 0xD0
25 #define DATA_IN_14 0xE0
26 #define DATA_IN_15 0xF0
27
28 #define DATA_OUT_0 0x01
29 #define DATA_OUT_1 0x11
30 #define DATA_OUT_2 0x21
31 #define DATA_OUT_3 0x31
32 #define DATA_OUT_4 0x41
33 #define DATA_OUT_5 0x51
34 #define DATA_OUT_6 0x61
35 #define DATA_OUT_7 0x71
36 #define DATA_OUT_8 0x81
37 #define DATA_OUT_9 0x91
38 #define DATA_OUT_10 0xA1
39 #define DATA_OUT_11 0xB1
40 #define DATA_OUT_12 0xC1
41 #define DATA_OUT_13 0xD1
42 #define DATA_OUT_14 0xE1
43 #define DATA_OUT_15 0xF1
44
45 #define IRQ_MASK_0 0x02
46 #define IRQ_MASK_1 0x12
47 #define IRQ_MASK_2 0x22
48 #define IRQ_MASK_3 0x32
49 #define IRQ_MASK_4 0x42
50 #define IRQ_MASK_5 0x52
51 #define IRQ_MASK_6 0x62
52 #define IRQ_MASK_7 0x72
53 #define IRQ_MASK_8 0x82
54 #define IRQ_MASK_9 0x92
55 #define IRQ_MASK_10 0xA2
56 #define IRQ_MASK_11 0xB2
57 #define IRQ_MASK_12 0xC2
58 #define IRQ_MASK_13 0xD2
59 #define IRQ_MASK_14 0xE2
60 #define IRQ_MASK_15 0xF2
61
62 #define EDGE_CAPTURE_0 0x03
63 #define EDGE_CAPTURE_1 0x13
64 #define EDGE_CAPTURE_2 0x23
65 #define EDGE_CAPTURE_3 0x33
66 #define EDGE_CAPTURE_4 0x43
67 #define EDGE_CAPTURE_5 0x53
68 #define EDGE_CAPTURE_6 0x63
69 #define EDGE_CAPTURE_7 0x73
70 #define EDGE_CAPTURE_8 0x83
71 #define EDGE_CAPTURE_9 0x93
72 #define EDGE_CAPTURE_10 0xA3
73 #define EDGE_CAPTURE_11 0xB3
74 #define EDGE_CAPTURE_12 0xC3
75 #define EDGE_CAPTURE_13 0xD3
76 #define EDGE_CAPTURE_14 0xE3
77 #define EDGE_CAPTURE_15 0xF3
78
79 #define PRIORITIZED_INTERRUPT_SRC 0x100
```

Listing 16.17: Avalon-MM Slave Template Macros - Altera Template (slave_template_macros.h)

16.4 FPGA Demo - Software Code

16.4.1 Partial Directory Structure

```
yocto-ghrd/
├── bitbake
│   └── <FILES>
├── build
│   ├── conf
│   │   ├── <FILES>
│   │   └── bblayers.conf
├── documentation
│   └── <FILES>
├── meta
│   └── <FILES>
├── meta-altera
│   ├── <FILES>
│   ├── recipes-core
│   │   ├── <FILES>
│   │   │   ├── images
│   │   │   ├── <FILES>
│   │   │   └── altera-gsrd.image.bb
│   ├── recipes-gsrd
│   │   ├── <FILES>
│   │   ├── gsrd-altera
│   │   │   └── <FILES>
│   │   ├── linux
│   │   │   ├── <FILES>
│   │   │   ├── files
│   │   │   │   └── socfpga.dts
├── meta-hob
│   └── <FILES>
├── meta-linaro
│   └── <FILES>
├── meta-oe
│   └── <FILES>
├── meta-skeleton
│   └── <FILES>
├── meta-snr-des
│   ├── conf
│   │   └── layer.conf
│   ├── recipes-orion
│   │   └── smart-cam-iptest
│   │   │   ├── files
│   │   │   │   └── test.c
│   │   │   └── smart-cam-iptest_1.0.bb
├── meta-yocto
│   └── <FILES>
├── meta-yocto-bsp
│   └── <FILES>
├── scripts
│   └── <FILES>
├── altera-init
└── LICENSE
```

```
├── make-sd.sh
├── oe-init-build-env
├── README
└── README.hardware
```

16.4.2 Senior Design Test BitBake Recipe (smart-cam-iptest_1.0.bb)

```
DESCRIPTION = "Software to Test ARM / FPGA Interface and Shared DDR3"
SECTION = "smart-cam-iptest"
LICENSE = "BSD"
LIC_FILES_CHKSUM = "file://${COMMON_LICENSE_DIR}/BSD;md5=3775480a712fc46a69647678acb234cb"
PR = "r0"
FILES_${PN} = "/home/root/snr-des/testIP/*"
FILES_${PN}-dbg = "/usr /home/root/snr-des/testIP/.debug"

SRC_URI = "file://test.c "

S = "${WORKDIR}"

#based on http://www.multitech.net/developer/products/multiconnect-ocg/development/writing-bitbake-recipes/
#and http://www.yoctoproject.org/docs/current/dev-manual/dev-manual.html#new-recipe-writing-a-new-recipe

do_compile() {
  ${CC} ${CFLAGS} ${LDFLAGS} test.c -o test
}

do_install() {
  install -d ${D}/home/root/snr-des/testIP

  install -m 0755 test ${D}/home/root/snr-des/testIP/test
}
```

Listing 16.18: Senior Design Test BitBake Recipe (smart-cam-iptest_1.0.bb)

16.4.3 Senior Design Test Application Source (test.c)

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <errno.h>
#include <stdbool.h>
#include <sys/types.h>
#include <sys/stat.h>
#include <fcntl.h>
#include <sys/mman.h>

#define DDR3_ARM_BASE 0xc0000000
#define DDR3_ARM_END  0xcfffffff
#define DDR3_IP_BASE  0x00000000
#define DDR3_IP_END   0x0fffffff
//the linux image uses pages of width 4 KB or 0x00001000
//as it turns out, this is the mem space of the entire bridge
//used to connect to the CSR.  Since we need to map an entire
//page, we need to take 0xff23000 to 0xff230fff.
//However, the real CSR starts at 0xff23800
#define IP_CSR_OFFSET 0x00000800
#define IP_CSR_BASE   0xff230000
#define IP_CSR_END    0xff230fff

//based on readme_first.txt from the altera template
//each register is the 2nd hexideximal digit (16 place)
//the least significan hex digit represents the access
```

```
//mode such as 0 = read, 1 = write, 2 = interupt mask
//3 = edge capture
#define ADDR_OFFSET        0x01 //address     (write)
#define MODE_OFFSET        0x11 //mode        (write) (0 = read, 1 = write)
#define DATA0W_OFFSET      0x21 //data0       (write)
#define DATA1W_OFFSET      0x31 //data1       (write)
#define DATA2W_OFFSET      0x41 //data2       (write)
#define DATA3W_OFFSET      0x51 //data3       (write)

#define READ_RDY_OFFSET    0x00 //readReady   (read)  (0 when ready)
#define WRITE_RDY_OFFSET   0x10 //writeReady  (read)  (0 when ready)
#define DATA0R_OFFSET      0x20 //data0       (read)
#define DATA1R_OFFSET      0x30 //data1       (read)
#define DATA2R_OFFSET      0x40 //data2       (read)
#define DATA3R_OFFSET      0x50 //data3       (read)

//Constants used by IP CSR
#define READ_MODE 0
#define WRITE_MODE 1
#define IP_READY 0

//info on how to use open and mmap on /dev/mem was based on http://stackoverflow.com/questions/9662193/how-to-access-kernel-space-from-user-spacein-linux
//o_sync is essential to ensure writes to and reads from /dev/mem are not buffered by the OS.  Bitwise oring is used to specify multiple parms


int main(int argc, char** argv)
{
  int memSize = DDR3_ARM_END - DDR3_ARM_BASE + 1;
  int csrSize = IP_CSR_END - IP_CSR_BASE + 1;

  int status;

  bool testPass;

  int test0, test1, test2, test3;

  int d0 = 0;
  int d1 = 0;
  int d2 = 0;
  int d3 = 0;

  int memOffset = 0;

  int exitStatus = EXIT_SUCCESS;

  int* ddr3Map;
  int* csrMap;

  int memFile = open("/dev/mem", O_RDWR | O_SYNC);

  if(memFile == -1){
    int errNum = errno;
    char* errorMsg = strerror(errNum);
    printf("Error opening /dev/mem: %s\n", errorMsg);
    return EXIT_FAILURE;
  }

  ddr3Map = (int *) mmap(NULL, memSize, PROT_READ | PROT_WRITE, MAP_SHARED, memFile, DDR3_ARM_BASE);

  if(ddr3Map == MAP_FAILED){
    int errNum = errno;
    char* errorMsg = strerror(errNum);
    printf("Error mapping DDR3: %s\n", errorMsg);
    return EXIT_FAILURE;
  }
```

```
92
93  csrMap = (int *) mmap(NULL, csrSize, PROT_READ | PROT_WRITE, MAP_SHARED, memFile,
        IP_CSR_BASE);
94
95  if(csrMap == MAP_FAILED){
96    int errNum = errno;
97    char* errorMsg = strerror(errNum);
98    printf("Error mapping IP CSR: %s\n", errorMsg);
99    return EXIT_FAILURE;
100 }
101
102 //********* Begin Test Code ***********
103 //Read Test
104 test0 = 0x0000000F;
105 test1 = 0x000000F0;
106 test2 = 0x00000F00;
107 test3 = 0x0000F000;
108
109 *(ddr3Map + memOffset)   = test0;
110 *(ddr3Map + memOffset+1) = test1;
111 *(ddr3Map + memOffset+2) = test2;
112 *(ddr3Map + memOffset+3) = test3;
113
114 printf("Wrote into DDR3 from ARM\n");
115 printf("0x%x\n", test0);
116 printf("0x%x\n", test1);
117 printf("0x%x\n", test2);
118 printf("0x%x\n", test3);
119
120 printf("\n");
121
122 //read using IP
123 *(csrMap + IP_CSR_OFFSET + ADDR_OFFSET) = DDR3_IP_BASE+memOffset;
124 *(csrMap + IP_CSR_OFFSET + MODE_OFFSET) = READ_MODE;
125
126 while((*(csrMap + IP_CSR_OFFSET + READ_RDY_OFFSET) & 1) != IP_READY){
127   ; //wait for IP to finish read operation
128   //this is a spin lock, I know, but this is a simple test of IP
129   //and not a high speed application
130 }
131
132 d0 = *(csrMap + IP_CSR_OFFSET + DATA0R_OFFSET);
133 d1 = *(csrMap + IP_CSR_OFFSET + DATA1R_OFFSET);
134 d2 = *(csrMap + IP_CSR_OFFSET + DATA2R_OFFSET);
135 d3 = *(csrMap + IP_CSR_OFFSET + DATA3R_OFFSET);
136
137 printf("Read from DDR3 in IP\n");
138 printf("0x%x\n", d0);
139 printf("0x%x\n", d1);
140 printf("0x%x\n", d2);
141 printf("0x%x\n", d3);
142
143 testPass = test0==d0 && test1==d1 && test2==d2 && test3==d3;
144
145 if(testPass){
146   printf("***** Read Test: Success *****");
147 }
148 else{
149   printf("***** Read Test: Fail    *****");
150 }
151
152 //Write test
153 memOffset = 0x00000010;
154
155 test0 = 0x11111111;
156 test1 = 0x22222222;
157 test2 = 0x33333333;
158 test3 = 0x44444444;
```

```
159
160   *(csrMap + IP_CSR_OFFSET + ADDR_OFFSET)    = DDR3_IP_BASE+memOffset;
161   *(csrMap + IP_CSR_OFFSET + DATA0W_OFFSET) = test0;
162   *(csrMap + IP_CSR_OFFSET + DATA1W_OFFSET) = test1;
163   *(csrMap + IP_CSR_OFFSET + DATA2W_OFFSET) = test2;
164   *(csrMap + IP_CSR_OFFSET + DATA3W_OFFSET) = test3;
165   *(csrMap + IP_CSR_OFFSET + MODE_OFFSET)    = WRITE_MODE;
166
167   while((*(csrMap + IP_CSR_OFFSET + WRITE_RDY_OFFSET) & 1) != IP_READY){
168     ; //wait for IP to finish write operation
169     //this is a spin lock, I know, but this is a simple test of IP
170     //and not a high speed application
171   }
172
173   printf("Wrote into DDR3 from IP\n");
174   printf("0x%x\n", test0);
175   printf("0x%x\n", test1);
176   printf("0x%x\n", test2);
177   printf("0x%x\n", test3);
178
179   printf("\n");
180
181   d0 = *(ddr3Map + memOffset);
182   d1 = *(ddr3Map + memOffset + 1);
183   d2 = *(ddr3Map + memOffset + 2);
184   d3 = *(ddr3Map + memOffset + 3);
185
186   printf("Read from DDR3 in ARM\n");
187   printf("0x%x\n", d0);
188   printf("0x%x\n", d1);
189   printf("0x%x\n", d2);
190   printf("0x%x\n", d3);
191
192   testPass = test0==d0 && test1==d1 && test2==d2 && test3==d3;
193
194   if(testPass){
195     printf("***** Write Test: Success *****");
196   }
197   else{
198     printf("***** Write Test: Fail    *****");
199   }
200
201
202   //********* End Test Code   **********
203
204   status = munmap(ddr3Map, memSize);
205
206   if(status == -1){
207     int errNum = errno;
208     char* errorMsg = strerror(errNum);
209     printf("Error un-mapping DDR3: %s\n", errorMsg);
210     exitStatus = EXIT_FAILURE;
211   }
212
213   status = munmap(csrMap, csrSize);
214
215   if(status == -1){
216     int errNum = errno;
217     char* errorMsg = strerror(errNum);
218     printf("Error un-mapping CSR: %s\n", errorMsg);
219     exitStatus = EXIT_FAILURE;
220   }
221
222   status = close(memFile);
223
224   if(status == -1){
225     int errNum = errno;
226     char* errorMsg = strerror(errNum);
```

```
227     printf("Error closeing /dev/mem: %s\n", errorMsg);
228     exitStatus = EXIT_FAILURE;
229   }
230
231   return exitStatus;
232 }
```

Listing 16.19: Senior Design Test Application Source (test.c)

16.4.4 Senior Design Layer Configuration File (layer.conf)

```
1  # Based on OE Layer.conf file
2
3  # We have a conf and classes directory, append to BBPATH
4  BBPATH .= ":${LAYERDIR}"
5
6  # We have a recipes directory, add to BBFILES
7  BBFILES += "${LAYERDIR}/recipes-*/*/*.bb ${LAYERDIR}/recipes-*/*/*.bbappend"
8
9  BBFILE_COLLECTIONS += "snr-des-layer"
10 BBFILE_PATTERN_snr-des-layer := "^${LAYERDIR}/"
11
12 # Define the priority for recipes (.bb files) from this layer,
13 # choosing carefully how this layer interacts with all of the
14 # other layers.
15
16 BBFILE_PRIORITY_openembedded-layer = "1"
```

Listing 16.20: Senior Design Layer Configuration File (layer.conf)

16.4.5 Modified bblayers File (bblayers.conf)

```
1  # LAYER_CONF_VERSION is increased each time build/conf/bblayers.conf
2  # changes incompatibly
3  LCONF_VERSION = "6"
4
5  BBPATH = "${TOPDIR}"
6  BBFILES ?= ""
7  BBLAYERS = " \
8    /home/christopher/yocto-ghrd/meta-snr-des \
9    /home/christopher/yocto-ghrd/meta-oe \
10   /home/christopher/yocto-ghrd/meta \
11   /home/christopher/yocto-ghrd/meta-yocto \
12   /home/christopher/yocto-ghrd/meta-yocto-bsp \
13   /home/christopher/yocto-ghrd/meta-yocto-bsp \
14   /home/christopher/yocto-ghrd/meta-linaro \
15   /home/christopher/yocto-ghrd/meta-altera \
16   "
```

Listing 16.21: Modified bblayers File (bblayers.conf)

16.4.6 Modified Image File (altera-gsrd-image.bb)

```
1
2  DESCRIPTION = "The set of packages for development and testing provided by Altera"
3
4  LICENSE = "MIT"
5
```

```
ALTERA_IMAGE_INSTALL ?= "initramfs-altera kernel-modules netbase busybox base-passwd base-
    files tinylogin sysvinit initscripts e2fsprogs mtd-utils gdb gdbserver bash strace
    openssh openssl elfutils sysfsutils usbutils dtc gawk ethtool grep lighttpd iputils
    make pciutils portmap sed setserial wget autoconf diffutils perl libav live555 live555
    -openrtsp live555-mediaserver minicom valgrind i2c-tools lttng-modules lttng-tools
    iptables oprofile net-tools gator openssh-sftp-server gsrd-altera pio-interrupt-altera
     initscripts-altera util-linux smart-cam-iptest"
IMAGE_INSTALL ?= "${ALTERA_IMAGE_INSTALL}"

inherit core-image

# altera-image.inc must be included after inherit core-image to override functionality

include altera-image.inc
```

Listing 16.22: Modified Image File (altera-gsrd-image.bb)

16.4.7 Modified Device Tree File (socfpga.dts)

```
/*
 * This devicetree is generated by sopc2dts on Sat May 03 22:38:06 PDT 2014
 * Sopc2dts is written by Walter Goossens <waltergoossens@home.nl>
 * in cooperation with the nios2 community <Nios2-dev@sopc.et.ntust.edu.tw>
 */
/dts-v1/;

/ {
  model = "ALTR,socfpga-cyclone5";
  compatible = "ALTR,socfpga-cyclone5";
  #address-cells = < 1 >;
  #size-cells = < 1 >;

  aliases {
    ethernet0 = "/sopc/ethernet@0xff702000";
    serial0 = "/sopc/serial@xffc02000";
    serial1 = "/sopc/serial@xffc03000";
    timer0 = "/sopc/timer@xffc08000";
    timer1 = "/sopc/timer@xffc09000";
    timer2 = "/sopc/timer@0xffd00000";
    timer3 = "/sopc/timer@0xffd01000";
  }; //end aliases

  cpus {
    #address-cells = < 1 >;
    #size-cells = < 0 >;

    hps_0_arm_a9_0: cpu@0x0 {
      device_type = "cpu";
      compatible = "arm,cortex-a9-1.0", "arm,cortex-a9";
      reg = < 0x00000000 >;
    }; //end cpu@0x0 (hps_0_arm_a9_0)

    hps_0_arm_a9_1: cpu@0x1 {
      device_type = "cpu";
      compatible = "arm,cortex-a9-1.0", "arm,cortex-a9";
      reg = < 0x00000001 >;
    }; //end cpu@0x1 (hps_0_arm_a9_1)
  }; //end cpus

  memory@0 {
    device_type = "memory";
    reg = < 0xC0000000 0x10000000
      0xFFFF0000 0x00010000 >;
  }; //end memory@0

  sopc@0 {
```

```
48      device_type = "soc";
49      ranges;
50      #address-cells = < 1 >;
51      #size-cells = < 1 >;
52      compatible = "ALTR,avalon", "simple-bus";
53      bus-frequency = < 50000000 >;
54
55      hps_0_h2f_lw: bridge@0xff200000 {
56        compatible = "altr,h2f_lw_bridge-1.0", "simple-bus";
57        reg = < 0xFF200000 0x00200000 >;
58        #address-cells = < 1 >;
59        #size-cells = < 1 >;
60        ranges = < 0x00010000 0xFF210000 0x00000008
61          0x00010040 0xFF210040 0x00000020
62          0x00010080 0xFF210080 0x00000010
63          0x000100C0 0xFF2100C0 0x00000010
64          0x00020000 0xFF220000 0x00000008
65          0x00030000 0xFF230000 0x00001000 >;
66
67        led_pio: gpio@0x10040 {
68          compatible = "ALTR,pio-13.0.1.99.2", "ALTR,pio-1.0", "altr,pio-1.0";
69          reg = < 0x00010040 0x00000020 >;
70          width = < 4 >;  /* width type NUMBER */
71          resetvalue = < 0 >; /* resetValue type NUMBER */
72          #gpio-cells = < 2 >;
73          gpio-controller;
74        }; //end gpio@0x10040 (led_pio)
75
76        dipsw_pio: gpio@0x10080 {
77          compatible = "ALTR,pio-13.0.1.99.2", "ALTR,pio-1.0", "altr,pio-1.0";
78          reg = < 0x00010080 0x00000010 >;
79          interrupt-parent = < &hps_0_arm_gic_0 >;
80          interrupts = < 0 40 1 >;
81          width = < 4 >;  /* width type NUMBER */
82          resetvalue = < 0 >; /* resetValue type NUMBER */
83          edge_type = < 2 >;  /* embeddedsw.dts.params.edge_type type NUMBER */
84          level_trigger = < 0 >;  /* embeddedsw.dts.params.level_trigger type NUMBER */
85          #gpio-cells = < 2 >;
86          gpio-controller;
87        }; //end gpio@0x10080 (dipsw_pio)
88
89        button_pio: gpio@0x100c0 {
90          compatible = "ALTR,pio-13.0.1.99.2", "ALTR,pio-1.0", "altr,pio-1.0";
91          reg = < 0x000100C0 0x00000010 >;
92          interrupt-parent = < &hps_0_arm_gic_0 >;
93          interrupts = < 0 41 1 >;
94          width = < 2 >;  /* width type NUMBER */
95          resetvalue = < 0 >; /* resetValue type NUMBER */
96          edge_type = < 1 >;  /* embeddedsw.dts.params.edge_type type NUMBER */
97          level_trigger = < 0 >;  /* embeddedsw.dts.params.level_trigger type NUMBER */
98          #gpio-cells = < 2 >;
99          gpio-controller;
100       }; //end gpio@0x100c0 (button_pio)
101
102       jtag_uart: serial@0x20000 {
103         compatible = "ALTR,juart-13.0.1.99.2", "ALTR,juart-1.0";
104         reg = < 0x00020000 0x00000008 >;
105         interrupt-parent = < &hps_0_arm_gic_0 >;
106         interrupts = < 0 42 4 >;
107       }; //end serial@0x20000 (jtag_uart)
108
109       mm_clock_crossing_bridge_1: bridge@0x30000 {
110         compatible = "ALTR,avalon-13.0", "simple-bus";
111         reg = < 0x00030000 0x00001000 >;
112         #address-cells = < 1 >;
113         #size-cells = < 1 >;
114         ranges = < 0x00000800 0x00030800 0x00000800 >;
115
```

```
116         IPTest_0: unknown@0x800 {
117           compatible = "unknown,unknown-1.0", "simple-bus";
118           reg = < 0x00000800 0x00000800 >;
119         }; //end unknown@0x800 (IPTest_0)
120       }; //end bridge@0x30000 (mm_clock_crossing_bridge_1)
121     }; //end bridge@0xff200000 (hps_0_h2f_lw)
122
123     hps_0_arm_gic_0: intc@0xfffed000 {
124       compatible = "arm,cortex-a9-gic-1.0", "arm,cortex-a9-gic";
125       reg = < 0xFFFED000 0x00001000
126         0xFFFEC100 0x00000100 >;
127       interrupt-controller;
128       #interrupt-cells = < 3 >;
129     }; //end intc@0xfffed000 (hps_0_arm_gic_0)
130
131     hps_0_L2: L2-cache@0xfffef000 {
132       compatible = "arm,pl310-cache-1.0", "arm,pl310-cache";
133       reg = < 0xFFFEF000 0x00001000 >;
134       interrupt-parent = < &hps_0_arm_gic_0 >;
135       interrupts = < 0 38 4 >;
136       cache-level = < 2 >;  /* embeddedsw.dts.params.cache-level type NUMBER */
137       cache-unified;  /* appended from boardinfo */
138       arm,tag-latency = <1 1 1>;
139       arm,data-latency = <2 1 1>;
140     }; //end L2-cache@0xfffef000 (hps_0_L2)
141
142     hps_0_dma: dma@0xffe01000 {
143       compatible = "arm,pl330-1.0", "arm,pl330", "arm,primecell";
144       reg = < 0xFFE01000 0x00001000 >;
145       interrupt-parent = < &hps_0_arm_gic_0 >;
146       interrupts = < 0 104 4 >;
147     }; //end dma@0xffe01000 (hps_0_dma)
148
149     hps_0_sysmgr: sysmgr@0xffd08000 {
150       compatible = "altr,sys-mgr-1.0", "altr,sys-mgr";
151       reg = < 0xFFD08000 0x00004000 >;
152     }; //end sysmgr@0xffd08000 (hps_0_sysmgr)
153
154     hps_0_clkmgr: clkmgr@0xffd04000 {
155       compatible = "altr,clk-mgr-1.0", "altr,clk-mgr";
156       reg = < 0xFFD04000 0x00001000 >;
157     }; //end clkmgr@0xffd04000 (hps_0_clkmgr)
158
159     hps_0_rstmgr: rstmgr@0xffd05000 {
160       compatible = "altr,rst-mgr-1.0", "altr,rst-mgr";
161       reg = < 0xFFD05000 0x00001000 >;
162     }; //end rstmgr@0xffd05000 (hps_0_rstmgr)
163
164     hps_0_fpgamgr: fpgamgr@0xff706000 {
165       compatible = "altr,fpga-mgr-1.0", "altr,fpga-mgr";
166       reg = < 0xFF706000 0x00001000
167         0xFFB90000 0x00001000 >;
168       interrupt-parent = < &hps_0_arm_gic_0 >;
169       interrupts = < 0 175 4 >;
170       transport = "mmio"; /* embeddedsw.dts.params.transport type STRING */
171     }; //end fpgamgr@0xff706000 (hps_0_fpgamgr)
172
173     hps_0_uart0: serial@0xffc02000 {
174       compatible = "snps,dw-apb-uart-1.0", "snps,dw-apb-uart";
175       reg = < 0xFFC02000 0x00001000 >;
176       interrupt-parent = < &hps_0_arm_gic_0 >;
177       interrupts = < 0 162 4 >;
178       reg-io-width = < 4 >; /* embeddedsw.dts.params.reg-io-width type NUMBER */
179       reg-shift = < 2 >;  /* embeddedsw.dts.params.reg-shift type NUMBER */
180       clock-frequency = < 100000000 >;  /* appended from boardinfo */
181     }; //end serial@0xffc02000 (hps_0_uart0)
182
183     hps_0_uart1: serial@0xffc03000 {
```

```
        compatible = "snps,dw-apb-uart-1.0", "snps,dw-apb-uart";
        reg = < 0xFFC03000 0x00001000 >;
        interrupt-parent = < &hps_0_arm_gic_0 >;
        interrupts = < 0 163 4 >;
        reg-io-width = < 4 >; /* embeddedsw.dts.params.reg-io-width type NUMBER */
        reg-shift = < 2 >;  /* embeddedsw.dts.params.reg-shift type NUMBER */
        clock-frequency = < 100000000 >;  /* appended from boardinfo */
      }; //end serial@0xffc03000 (hps_0_uart1)

      hps_0_timer0: timer@0xffc08000 {
        compatible = "snps,dw-apb-timer-sp-1.0", "snps,dw-apb-timer-sp";
        reg = < 0xFFC08000 0x00001000 >;
        interrupt-parent = < &hps_0_arm_gic_0 >;
        interrupts = < 0 167 4 >;
        clock-frequency = < 100000000 >;  /* appended from boardinfo */
      }; //end timer@0xffc08000 (hps_0_timer0)

      hps_0_timer1: timer@0xffc09000 {
        compatible = "snps,dw-apb-timer-sp-1.0", "snps,dw-apb-timer-sp";
        reg = < 0xFFC09000 0x00001000 >;
        interrupt-parent = < &hps_0_arm_gic_0 >;
        interrupts = < 0 168 4 >;
        clock-frequency = < 100000000 >;  /* appended from boardinfo */
      }; //end timer@0xffc09000 (hps_0_timer1)

      hps_0_timer2: timer@0xffd00000 {
        compatible = "snps,dw-apb-timer-osc-1.0", "snps,dw-apb-timer-osc";
        reg = < 0xFFD00000 0x00001000 >;
        interrupt-parent = < &hps_0_arm_gic_0 >;
        interrupts = < 0 169 4 >;
        clock-frequency = < 25000000 >; /* appended from boardinfo */
      }; //end timer@0xffd00000 (hps_0_timer2)

      hps_0_timer3: timer@0xffd01000 {
        compatible = "snps,dw-apb-timer-osc-1.0", "snps,dw-apb-timer-osc";
        reg = < 0xFFD01000 0x00001000 >;
        interrupt-parent = < &hps_0_arm_gic_0 >;
        interrupts = < 0 170 4 >;
        clock-frequency = < 25000000 >; /* appended from boardinfo */
      }; //end timer@0xffd01000 (hps_0_timer3)

      hps_0_gpio0: gpio@0xff708000 {
        compatible = "snps,dw-gpio-1.0", "snps,dw-gpio";
        reg = < 0xFF708000 0x00001000 >;
        interrupt-parent = < &hps_0_arm_gic_0 >;
        interrupts = < 0 164 4 >;
        #gpio-cells = < 2 >;
        gpio-controller;
      }; //end gpio@0xff708000 (hps_0_gpio0)

      hps_0_gpio1: gpio@0xff709000 {
        compatible = "snps,dw-gpio-1.0", "snps,dw-gpio";
        reg = < 0xFF709000 0x00001000 >;
        interrupt-parent = < &hps_0_arm_gic_0 >;
        interrupts = < 0 165 4 >;
        #gpio-cells = < 2 >;
        gpio-controller;
      }; //end gpio@0xff709000 (hps_0_gpio1)

      hps_0_gpio2: gpio@0xff70a000 {
        compatible = "snps,dw-gpio-1.0", "snps,dw-gpio";
        reg = < 0xFF70A000 0x00001000 >;
        interrupt-parent = < &hps_0_arm_gic_0 >;
        interrupts = < 0 166 4 >;
        #gpio-cells = < 2 >;
        gpio-controller;
      }; //end gpio@0xff70a000 (hps_0_gpio2)

```

```
252     hps_0_i2c0: i2c@0xffc04000 {
253       compatible = "snps,designware-i2c-1.0", "snps,designware-i2c";
254       reg = < 0xFFC04000 0x00001000 >;
255       interrupt-parent = < &hps_0_arm_gic_0 >;
256       interrupts = < 0 158 4 >;
257       emptyfifo_hold_master = < 1 >;  /* embeddedsw.dts.params.emptyfifo_hold_master type
              NUMBER */
258       #address-cells = < 1 >;
259       #size-cells = < 0 >;
260       speed-mode = < 0 >; /* appended from boardinfo */
261
262       lcd: newhaven,nhd-0216k3z-nsw-bbw@0x28 {
263         compatible = "newhaven,nhd-0216k3z-nsw-bbw";
264         reg = < 0x00000028 >;
265         height = < 2 >; /* appended from boardinfo */
266         width = < 16 >; /* appended from boardinfo */
267       }; //end newhaven,nhd-0216k3z-nsw-bbw@0x28 (lcd)
268
269       eeprom: atmel,24c32@0x51 {
270         compatible = "atmel,24c32";
271         reg = < 0x00000051 >;
272         pagesize = < 32 >;  /* appended from boardinfo */
273       }; //end atmel,24c32@0x51 (eeprom)
274     }; //end i2c@0xffc04000 (hps_0_i2c0)
275
276     hps_0_i2c1: i2c@0xffc05000 {
277       compatible = "snps,designware-i2c-1.0", "snps,designware-i2c";
278       reg = < 0xFFC05000 0x00001000 >;
279       interrupt-parent = < &hps_0_arm_gic_0 >;
280       interrupts = < 0 159 4 >;
281       emptyfifo_hold_master = < 1 >;  /* embeddedsw.dts.params.emptyfifo_hold_master type
              NUMBER */
282     }; //end i2c@0xffc05000 (hps_0_i2c1)
283
284     hps_0_i2c2: i2c@0xffc06000 {
285       compatible = "snps,designware-i2c-1.0", "snps,designware-i2c";
286       reg = < 0xFFC06000 0x00001000 >;
287       interrupt-parent = < &hps_0_arm_gic_0 >;
288       interrupts = < 0 160 4 >;
289       emptyfifo_hold_master = < 1 >;  /* embeddedsw.dts.params.emptyfifo_hold_master type
              NUMBER */
290     }; //end i2c@0xffc06000 (hps_0_i2c2)
291
292     hps_0_i2c3: i2c@0xffc07000 {
293       compatible = "snps,designware-i2c-1.0", "snps,designware-i2c";
294       reg = < 0xFFC07000 0x00001000 >;
295       interrupt-parent = < &hps_0_arm_gic_0 >;
296       interrupts = < 0 161 4 >;
297       emptyfifo_hold_master = < 1 >;  /* embeddedsw.dts.params.emptyfifo_hold_master type
              NUMBER */
298     }; //end i2c@0xffc07000 (hps_0_i2c3)
299
300     hps_0_nand0: flash@0xff900000 {
301       compatible = "denali,nand-1.0", "denali,denali-nand-dt";
302       reg = < 0xFF900000 0x00100000
303         0xFFB80000 0x00010000 >;
304       interrupt-parent = < &hps_0_arm_gic_0 >;
305       interrupts = < 0 144 4 >;
306       #adress-cells = < 1 >;  /* embeddedsw.dts.params.#adress-cells type NUMBER */
307       #size-cells = < 1 >;  /* embeddedsw.dts.params.#size-cells type NUMBER */
308       reg-names = "nand_data", "denali_reg";  /* embeddedsw.dts.params.reg-names type
              STRING */
309       bank-width = < 2 >;
310       device-width = < 1 >;
311     }; //end flash@0xff900000 (hps_0_nand0)
312
313     hps_0_qspi: flash@0xff705000 {
314       compatible = "cadence,qspi-1.0", "cadence,qspi";
```

```
315         reg = < 0xFF705000 0x00001000
316           0xFFA00000 0x00001000 >;
317         interrupt-parent = < &hps_0_arm_gic_0 >;
318         interrupts = < 0 151 4 >;
319         bus-num = < 2 >;  /* embeddedsw.dts.params.bus-num type NUMBER */
320         fifo-depth = < 128 >; /* embeddedsw.dts.params.fifo-depth type NUMBER */
321         num-chipselect = < 4 >; /* embeddedsw.dts.params.num-chipselect type NUMBER */
322         bank-width = < 2 >;
323         device-width = < 1 >;
324         master-ref-clk = < 400000000 >; /* appended from boardinfo */
325         ext-decoder = < 0 >;  /* appended from boardinfo */
326
327         flash0: n25q128@0 {
328           #address-cells = < 1 >; /* appended from boardinfo */
329           #size-cells = < 1 >;  /* appended from boardinfo */
330           compatible = "n25q128"; /* appended from boardinfo */
331           reg = < 0 >;  /* appended from boardinfo */
332           spi-max-frequency = < 100000000 >;  /* appended from boardinfo */
333           page-size = < 256 >;  /* appended from boardinfo */
334           block-size = < 16 >;  /* appended from boardinfo */
335           quad = < 1 >; /* appended from boardinfo */
336           tshsl-ns = < 200 >; /* appended from boardinfo */
337           tsd2d-ns = < 255 >; /* appended from boardinfo */
338           tchsh-ns = < 20 >;  /* appended from boardinfo */
339           tslch-ns = < 20 >;  /* appended from boardinfo */
340
341           part0: partition@0 {
342             label = "Flash 0 Raw Data"; /* appended from boardinfo */
343           }; //end partition@0 (part0)
344
345           part1: partition@800000 {
346             label = "Flash 1 jffs2 Filesystem"; /* appended from boardinfo */
347           }; //end partition@800000 (part1)
348         }; //end n25q128@0 (flash0)
349       }; //end flash@0xff705000 (hps_0_qspi)
350
351       hps_0_sdmmc: flash@0xff704000 {
352         compatible = "snps,mmc-1.0", "snps,dw-mshc";
353         reg = < 0xFF704000 0x00001000 >;
354         interrupt-parent = < &hps_0_arm_gic_0 >;
355         interrupts = < 0 139 4 >;
356         fifo-depth = < 1024 >;  /* embeddedsw.dts.params.fifo-depth type NUMBER */
357         num-slots = < 1 >;  /* embeddedsw.dts.params.num-slots type NUMBER */
358         bank-width = < 2 >;
359         device-width = < 1 >;
360         bus-hz = < 12500000 >;  /* appended from boardinfo */
361         #address-cells = < 1 >; /* appended from boardinfo */
362         #size-cells = < 0 >;  /* appended from boardinfo */
363         supports-highspeed; /* appended from boardinfo */
364         broken-cd;  /* appended from boardinfo */
365
366         slot_0: slot@0 {
367           reg = < 0 >;  /* appended from boardinfo */
368           bus-width = < 4 >;  /* appended from boardinfo */
369         }; //end slot@0 (slot_0)
370       }; //end flash@0xff704000 (hps_0_sdmmc)
371
372       hps_0_usb0: usb@0xffb00000 {
373         compatible = "snps,dwc-otg-1.0", "snps,dwc-otg";
374         reg = < 0xFFB00000 0x00001000 >;
375         interrupt-parent = < &hps_0_arm_gic_0 >;
376         interrupts = < 0 125 4 >;
377         dev-nperio-tx-fifo-size = < 4096 >; /* embeddedsw.dts.params.dev-nperio-tx-fifo-size
              type NUMBER */
378         dev-perio-tx-fifo-size = "<512 512 512 512 512 512 512 512 512 512 512 512 512 512 512
              512>"; /* embeddedsw.dts.params.dev-perio-tx-fifo-size type STRING */
379         dev-tx-fifo-size = "<512 512 512 512 512 512 512 512 512 512 512 512 512 512 512 512>";
              /* embeddedsw.dts.params.dev-tx-fifo-size type STRING */
```

```
380         dev_rx_fifo-size = < 512 >; /* embeddedsw.dts.params.dev_rx_fifo-size type NUMBER */
381         dma-mask = < 268435455 >; /* embeddedsw.dts.params.dma-mask type NUMBER */
382         host_rx_fifo-size = < 512 >;  /* embeddedsw.dts.params.host_rx_fifo-size type NUMBER
                */
383         ulpi-ddr = < 0 >; /* embeddedsw.dts.params.ulpi-ddr type NUMBER */
384         voltage-switch = < 0 >; /* embeddedsw.dts.params.voltage-switch type NUMBER */
385       }; //end usb@0xffb00000 (hps_0_usb0)
386
387       hps_0_usb1: usb@0xffb40000 {
388         compatible = "snps,dwc-otg-1.0", "snps,dwc-otg";
389         reg = < 0xFFB40000 0x00001000 >;
390         interrupt-parent = < &hps_0_arm_gic_0 >;
391         interrupts = < 0 128 4 >;
392         dev-nperio-tx-fifo-size = < 4096 >; /* embeddedsw.dts.params.dev-nperio-tx-fifo-size
                type NUMBER */
393         dev-perio-tx-fifo-size = "<512 512 512 512 512 512 512 512 512 512 512 512 512 512
                512>"; /* embeddedsw.dts.params.dev-perio-tx-fifo-size type STRING */
394         dev-tx-fifo-size = "<512 512 512 512 512 512 512 512 512 512 512 512 512 512 512>";
                /* embeddedsw.dts.params.dev-tx-fifo-size type STRING */
395         dev_rx_fifo-size = < 512 >; /* embeddedsw.dts.params.dev_rx_fifo-size type NUMBER */
396         dma-mask = < 268435455 >; /* embeddedsw.dts.params.dma-mask type NUMBER */
397         host_rx_fifo-size = < 512 >;  /* embeddedsw.dts.params.host_rx_fifo-size type NUMBER
                */
398         ulpi-ddr = < 0 >; /* embeddedsw.dts.params.ulpi-ddr type NUMBER */
399         voltage-switch = < 0 >; /* embeddedsw.dts.params.voltage-switch type NUMBER */
400       }; //end usb@0xffb40000 (hps_0_usb1)
401
402       hps_0_gmac0: ethernet@0xff700000 {
403         compatible = "synopsys,dwmac-1.0", "altr,socfpga-stmmac", "snps,dwmac-3.70a", "snps,
                dwmac";
404         reg = < 0xFF700000 0x00002000 >;
405         interrupt-parent = < &hps_0_arm_gic_0 >;
406         interrupts = < 0 115 4 >;
407         interrupt-names = "macirq"; /* embeddedsw.dts.params.interrupt-names type STRING */
408         mac-address = "[00 00 00 00 00 00]";  /* embeddedsw.dts.params.mac-address type
                STRING */
409         address-bits = < 48 >;
410         max-frame-size = < 1518 >;
411         local-mac-address = [ 00 00 00 00 00 00 ];
412         status = "disabled";  /* appended from boardinfo */
413       }; //end ethernet@0xff700000 (hps_0_gmac0)
414
415       hps_0_gmac1: ethernet@0xff702000 {
416         compatible = "synopsys,dwmac-1.0", "altr,socfpga-stmmac", "snps,dwmac-3.70a", "snps,
                dwmac";
417         reg = < 0xFF702000 0x00002000 >;
418         interrupt-parent = < &hps_0_arm_gic_0 >;
419         interrupts = < 0 120 4 >;
420         interrupt-names = "macirq"; /* embeddedsw.dts.params.interrupt-names type STRING */
421         mac-address = "[00 00 00 00 00 00]";  /* embeddedsw.dts.params.mac-address type
                STRING */
422         address-bits = < 48 >;
423         max-frame-size = < 1518 >;
424         local-mac-address = [ 00 00 00 00 00 00 ];
425         phy-mode = "rgmii"; /* appended from boardinfo */
426         phy-addr = < 0xFFFFFFFF >;  /* appended from boardinfo */
427       }; //end ethernet@0xff702000 (hps_0_gmac1)
428
429       hps_0_timer: timer@0xfffec600 {
430         compatible = "arm,cortex-a9-twd-timer-1.0", "arm,cortex-a9-twd-timer";
431         reg = < 0xFFFEC600 0x00000100 >;
432         interrupt-parent = < &hps_0_arm_gic_0 >;
433         interrupts = < 1 13 3844 >;
434       }; //end timer@0xfffec600 (hps_0_timer)
435     }; //end sopc@0
436
437     leds {
438       compatible = "gpio-leds";
```

```
439     fpga0 {
440       gpios = <&led_pio 0 1>;
441       label = "fpga_led0";
442     };
443     fpga1 {
444       gpios = <&led_pio 1 1>;
445       label = "fpga_led1";
446     };
447     fpga2 {
448       gpios = <&led_pio 2 1>;
449       label = "fpga_led2";
450     };
451     fpga3 {
452       gpios = <&led_pio 3 1>;
453       label = "fpga_led3";
454     };
455     hps0 {
456       gpios = <&hps_0_gpio1 15 1>;
457       label = "hps_led0";
458     };
459     hps1 {
460       gpios = <&hps_0_gpio1 14 1>;
461       label = "hps_led1";
462     };
463     hps2 {
464       gpios = <&hps_0_gpio1 13 1>;
465       label = "hps_led2";
466     };
467     hps3 {
468       gpios = <&hps_0_gpio1 12 1>;
469       label = "hps_led3";
470     };
471   };
472   chosen {
473     bootargs = "console=ttyS0,57600";
474   }; //end chosen
475 }; //end /
```

Listing 16.23: Modified Device Tree File (socfpga.dts)

16.4.8 SD Card Flash Script (make-sd.sh)

```
1  #!/bin/bash
2
3  set -v
4
5  cd ~/yocto-ghrd/build/tmp/deploy/images
6  sudo rm -rf rootfs
7  mkdir rootfs
8  cd rootfs
9  sudo tar xzf ../altera-gsrd-image-socfpga_cyclone5.tar.gz
10 cd ..
11 rm -f sd_image_yocto.bin
12 sudo /opt/altera-linux-ghrd/bin/make_sdimage.sh -k uImage,socfpga.dtb -rp u-boot-spl-
       socfpga_cyclone5.bin -t ~/altera/13.0sp1/embedded/host_tools/altera/mkpimage/mkpimage
       -b u-boot-socfpga_cyclone5.img -r rootfs/ -o sd_image_yocto.bin
```

Listing 16.24: SD Card Flash Script (make-sd.sh)

16.5 Database - SQL Table Creation Code

```
1 IF object_id('Anomaly') is not null
2 BEGIN
```

```
    DROP TABLE Anomaly
END

IF object_id('DetectionEvent') is not null
BEGIN
    DROP TABLE DetectionEvent
END

IF object_id('CameraStats') is not null
BEGIN
    DROP TABLE CameraStats
END

IF object_id('Camera') is not null
BEGIN
    DROP TABLE Camera
END

IF object_id('KnownImgs') is not null
BEGIN
    DROP TABLE KnownImgs
END

IF object_id('Individual') is not null
BEGIN
    DROP TABLE Individual
END

CREATE TABLE Individual
(
   ID Integer IDENTITY(1,1) Not Null PRIMARY KEY,
   Name Varchar(40) Not Null,
   Details Varchar(100) Null
);

CREATE TABLE KnownImgs
(
   KnownImgID Integer IDENTITY(1,1) Not Null PRIMARY KEY,
   IDImg Integer Not Null,
   Img Image Null,
   FOREIGN KEY(IDImg) REFERENCES Individual(ID) on delete cascade
);

CREATE TABLE Camera
(
   CameraID SmallInt Not Null PRIMARY KEY,
   CameraName Varchar(20) Not Null,
   CameraIP Varchar(40) Null,
   FPGAIP Varchar(40) Null,
   CameraLoc Varchar(40) Null,
   CameraDetails Varchar(100) Null,
   imgIndex smallint Null,
   ThumbPath Varchar(10) Null
);

CREATE TABLE CameraStats
(
   TimeID SmallInt Not Null PRIMARY KEY,
   CameraID SmallInt Not Null,
   LoTimeRange Time(2) Null,
   HiTimeRange Time(2) Null,
   Simga Float Null,
   Mean Float Null,
   NValue Integer Default(0),
   FOREIGN KEY(CameraID) REFERENCES Camera(CameraID) on delete cascade
);

CREATE TABLE DetectionEvent
```

```
(
  EventID Integer PRIMARY KEY Not Null,
  EventCamera SmallInt Not Null,
  EventTimeIn Time(2) Not Null,
  EventTimeOut Time(2) Null,
  isChecked Int default(0),
  EntryType SmallInt Null,
  ExitType SmallInt Null,
  SuspectID Integer Not Null,
  Confidence Float Not Null,
  --ImgSize int Null,
  --ImgXPos int Null,
  --ImgYPos int Null,
  FOREIGN KEY(EventCamera) REFERENCES Camera(CameraID) on delete cascade,
  FOREIGN KEY(SuspectID) REFERENCES Individual(ID) on delete cascade
);

CREATE TABLE Anomaly
(
  AnomalyID Integer Identity(1,1) PRIMARY KEY(AnomalyID),
  EventID Integer Not Null,
  Severity Float default(0),
  isChecked Smallint default(0),
  AnomalyType SmallInt Null,
  FOREIGN KEY(EventID) REFERENCES DetectionEvent(EventID) on delete cascade
);
```

Listing 16.25: Code for Creation of the Database's Tables

16.6 Behavioral Analysis Engine - Primary File

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Data.Linq;
using System.Data.Linq.Mapping;
using System.Text;
using System.Threading.Tasks;

namespace SDBahavioralAnalysis
{
    class Program
    {
        //SeniorDesignCamStatsDataContext SDDatabase;
        static void Main(string[] args)
        {
            SeniorDesignCamStatsDataContext SDDatabase = new
                SeniorDesignCamStatsDataContext();
            DateTime CurrentTime = new DateTime();


            while(SDDatabase.DatabaseExists())
            {
                CurrentTime = DateTime.Now;
                var NewInEvents = (from events in SDDatabase.GetTable<DetectionEvent>()
                    where (events.ExitType == null && events.isChecked == 0) select events
                    );
                var NewOutEvents = (from events in SDDatabase.GetTable<DetectionEvent>()
                    where (events.ExitType != null && events.EventTimeOut != null &&
                    events.isChecked == 0) select events);
                var CamStats = (from camtables in SDDatabase.GetTable<CameraStat>() where
                    (camtables.HiTimeRange > CurrentTime.TimeOfDay && CurrentTime.
                    TimeOfDay > camtables.LoTimeRange) select camtables);

                foreach(DetectionEvent Event in NewOutEvents)
```

```
28                  {
29
30                      if (Event.ExitType == 0 || Event.EntryType == 0)
31                          ReportAnomaly(SDDatabase, Event);
32                      foreach(CameraStat Camera in CamStats)
33                      {
34                          if( Camera.CameraID == Event.EventID)
35                          {
36                              double x = (((TimeSpan)Event.EventTimeOut).TotalMinutes -
                                    Event.EventTimeIn.TotalMinutes);
37                              if (Camera.NValue > 36)
38                              {
39                                  if (Math.Abs(x) > (Camera.Mean + (Camera.Simga * 3)))
40                                  {
41                                      ReportAnomaly(SDDatabase, Event);
42                                      continue;
43                                  }
44                                  if ( checkZVal(Camera, x) > 1.96)
45                                      ReportAnomaly(SDDatabase, Event);
46                              }
47                              UpdateStats(SDDatabase, Camera, Event, x);
48                          }
49                      }
50                  }//end of foreach Out
51
52                  foreach(DetectionEvent Event in NewInEvents)
53                  {
54                      foreach( CameraStat Camera in CamStats)
55                      {
56                          double x = (DateTime.Now.TimeOfDay.TotalMinutes - Event.
                                EventTimeIn.TotalMinutes);
57                          if (Camera.NValue > 36 && checkZVal(Camera, x) > 1.96)
58                          {
59                              ReportAnomaly(SDDatabase, Event);
60                              Event.isChecked = 1;
61                              try
62                              {
63                                  SDDatabase.SubmitChanges();
64                              }
65                              catch (Exception e)
66                              {
67                                  Console.WriteLine("Database Connection Failed, Please
                                        Check Connection and Restart Program. Details: " + e);
68                              }
69                          }
70                      }
71                  }
72
73              }//end of while
74
75          }//end of main
76
77          private static double checkZVal(CameraStat CamInfo, double x)
78          {
79              double z = ((x - (double)CamInfo.Mean) / (double)CamInfo.Simga) * Math.Sqrt((
                    double)CamInfo.NValue);
80              return z;
81          }
82
83          private static void UpdateStats(SeniorDesignCamStatsDataContext SDDatabase,
                CameraStat CameraInfo, DetectionEvent Event, double x)
84          {
85              int NewNVal = (int)CameraInfo.NValue + 1;
86              double NewMean = (((double)CameraInfo.Mean * ( NewNVal - 1 )) + x) / NewNVal;
87              double NewSigma = Math.Sqrt( (((NewNVal - 1) * Math.Pow((int)CameraInfo.Simga
                    ,2.0)) + Math.Pow( (x - NewMean), 2.0 )) / NewNVal);
88              CameraInfo.NValue = NewNVal;
89              CameraInfo.Mean = NewMean;
```

```
            CameraInfo.Simga = NewSigma;
            Event.isChecked = 1;
            try
            {
                SDDatabase.SubmitChanges();
                Console.WriteLine( "Camera " + CameraInfo.CameraID + ": Updated with new values!" );
            }
            catch (Exception e)
            {
                Console.WriteLine("Database Connection Failed, Please Check Connection and Restart Program. Details: " + e);
                return;
            }
        }

        private static void ReportAnomaly(SeniorDesignCamStatsDataContext SDDatabase, DetectionEvent Event)
        {
            Anomaly newAnom = new Anomaly
            {
                EventID = Event.EventID,
                Severity = 0.0,
                AnomalyType = null,
                isChecked = 0
            };

            SDDatabase.Anomalies.InsertOnSubmit(newAnom);
            Event.isChecked = 1;

            try
            {
                SDDatabase.SubmitChanges();
                Console.WriteLine("Anomaly Detected!!!!!");
            }
            catch (Exception e)
            {
                Console.WriteLine("Database Connection Failed, Please Check Connection and Restart Program. Details: " + e);
                return;
            }
        }

    }


}
```

Listing 16.26: Primary File for the Behavioral Analysis Engine

16.7 Behavioral Analysis Engine - Database Configuration File

```
#pragma warning disable 1591
//------------------------------------------------------------------------------
// <auto-generated>
//     This code was generated by a tool.
//     Runtime Version:4.0.30319.34014
//
//     Changes to this file may cause incorrect behavior and will be lost if
//     the code is regenerated.
// </auto-generated>
//------------------------------------------------------------------------------

namespace SDBahavioralAnalysis
{
```

```
14  using System.Data.Linq;
15  using System.Data.Linq.Mapping;
16  using System.Data;
17  using System.Collections.Generic;
18  using System.Reflection;
19  using System.Linq;
20  using System.Linq.Expressions;
21  using System.ComponentModel;
22  using System;
23
24
25  [global::System.Data.Linq.Mapping.DatabaseAttribute(Name="SeniorDesignDB2")]
26  public partial class SeniorDesignCamStatsDataContext : System.Data.Linq.DataContext
27  {
28
29    private static System.Data.Linq.Mapping.MappingSource mappingSource = new
          AttributeMappingSource();
30
31    #region Extensibility Method Definitions
32    partial void OnCreated();
33    partial void InsertCamera(Camera instance);
34    partial void UpdateCamera(Camera instance);
35    partial void DeleteCamera(Camera instance);
36    partial void InsertCameraStat(CameraStat instance);
37    partial void UpdateCameraStat(CameraStat instance);
38    partial void DeleteCameraStat(CameraStat instance);
39    partial void InsertDetectionEvent(DetectionEvent instance);
40    partial void UpdateDetectionEvent(DetectionEvent instance);
41    partial void DeleteDetectionEvent(DetectionEvent instance);
42    partial void InsertAnomaly(Anomaly instance);
43    partial void UpdateAnomaly(Anomaly instance);
44    partial void DeleteAnomaly(Anomaly instance);
45    #endregion
46
47    public SeniorDesignCamStatsDataContext() :
48        base(global::SDBahavioralAnalysis.Properties.Settings.Default.
              SeniorDesignDB2ConnectionString, mappingSource)
49    {
50      OnCreated();
51    }
52
53    public SeniorDesignCamStatsDataContext(string connection) :
54        base(connection, mappingSource)
55    {
56      OnCreated();
57    }
58
59    public SeniorDesignCamStatsDataContext(System.Data.IDbConnection connection) :
60        base(connection, mappingSource)
61    {
62      OnCreated();
63    }
64
65    public SeniorDesignCamStatsDataContext(string connection, System.Data.Linq.Mapping.
          MappingSource mappingSource) :
66        base(connection, mappingSource)
67    {
68      OnCreated();
69    }
70
71    public SeniorDesignCamStatsDataContext(System.Data.IDbConnection connection, System.
          Data.Linq.Mapping.MappingSource mappingSource) :
72        base(connection, mappingSource)
73    {
74      OnCreated();
75    }
76
77    public System.Data.Linq.Table<Camera> Cameras
```

```
78      {
79        get
80        {
81          return this.GetTable<Camera>();
82        }
83      }
84
85      public System.Data.Linq.Table<CameraStat> CameraStats
86      {
87        get
88        {
89          return this.GetTable<CameraStat>();
90        }
91      }
92
93      public System.Data.Linq.Table<DetectionEvent> DetectionEvents
94      {
95        get
96        {
97          return this.GetTable<DetectionEvent>();
98        }
99      }
100
101     public System.Data.Linq.Table<Anomaly> Anomalies
102     {
103       get
104       {
105         return this.GetTable<Anomaly>();
106       }
107     }
108   }
109
110   [global::System.Data.Linq.Mapping.TableAttribute(Name="dbo.Camera")]
111   public partial class Camera : INotifyPropertyChanging, INotifyPropertyChanged
112   {
113
114     private static PropertyChangingEventArgs emptyChangingEventArgs = new
          PropertyChangingEventArgs(String.Empty);
115
116     private short _CameraID;
117
118     private string _CameraName;
119
120     private string _CameraIP;
121
122     private string _FPGAIP;
123
124     private string _CameraLoc;
125
126     private string _CameraDetails;
127
128     private System.Nullable<short> _imgIndex;
129
130     private string _ThumbPath;
131
132     private EntitySet<CameraStat> _CameraStats;
133
134     private EntitySet<DetectionEvent> _DetectionEvents;
135
136     #region Extensibility Method Definitions
137     partial void OnLoaded();
138     partial void OnValidate(System.Data.Linq.ChangeAction action);
139     partial void OnCreated();
140     partial void OnCameraIDChanging(short value);
141     partial void OnCameraIDChanged();
142     partial void OnCameraNameChanging(string value);
143     partial void OnCameraNameChanged();
144     partial void OnCameraIPChanging(string value);
```

```
145     partial void OnCameraIPChanged();
146     partial void OnFPGAIPChanging(string value);
147     partial void OnFPGAIPChanged();
148     partial void OnCameraLocChanging(string value);
149     partial void OnCameraLocChanged();
150     partial void OnCameraDetailsChanging(string value);
151     partial void OnCameraDetailsChanged();
152     partial void OnimgIndexChanging(System.Nullable<short> value);
153     partial void OnimgIndexChanged();
154     partial void OnThumbPathChanging(string value);
155     partial void OnThumbPathChanged();
156     #endregion
157
158     public Camera()
159     {
160       this._CameraStats = new EntitySet<CameraStat>(new Action<CameraStat>(this.
              attach_CameraStats), new Action<CameraStat>(this.detach_CameraStats));
161       this._DetectionEvents = new EntitySet<DetectionEvent>(new Action<DetectionEvent>(
              this.attach_DetectionEvents), new Action<DetectionEvent>(this.
              detach_DetectionEvents));
162       OnCreated();
163     }
164
165     [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_CameraID", DbType="
          SmallInt NOT NULL", IsPrimaryKey=true)]
166     public short CameraID
167     {
168       get
169       {
170         return this._CameraID;
171       }
172       set
173       {
174         if ((this._CameraID != value))
175         {
176           this.OnCameraIDChanging(value);
177           this.SendPropertyChanging();
178           this._CameraID = value;
179           this.SendPropertyChanged("CameraID");
180           this.OnCameraIDChanged();
181         }
182       }
183     }
184
185     [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_CameraName", DbType="
          VarChar(20) NOT NULL", CanBeNull=false)]
186     public string CameraName
187     {
188       get
189       {
190         return this._CameraName;
191       }
192       set
193       {
194         if ((this._CameraName != value))
195         {
196           this.OnCameraNameChanging(value);
197           this.SendPropertyChanging();
198           this._CameraName = value;
199           this.SendPropertyChanged("CameraName");
200           this.OnCameraNameChanged();
201         }
202       }
203     }
204
205     [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_CameraIP", DbType="VarChar
          (40)")]
206     public string CameraIP
```

```
207     {
208       get
209       {
210         return this._CameraIP;
211       }
212       set
213       {
214         if ((this._CameraIP != value))
215         {
216           this.OnCameraIPChanging(value);
217           this.SendPropertyChanging();
218           this._CameraIP = value;
219           this.SendPropertyChanged("CameraIP");
220           this.OnCameraIPChanged();
221         }
222       }
223     }
224
225     [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_FPGAIP", DbType="VarChar
          (40)")]
226     public string FPGAIP
227     {
228       get
229       {
230         return this._FPGAIP;
231       }
232       set
233       {
234         if ((this._FPGAIP != value))
235         {
236           this.OnFPGAIPChanging(value);
237           this.SendPropertyChanging();
238           this._FPGAIP = value;
239           this.SendPropertyChanged("FPGAIP");
240           this.OnFPGAIPChanged();
241         }
242       }
243     }
244
245     [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_CameraLoc", DbType="
          VarChar(40)")]
246     public string CameraLoc
247     {
248       get
249       {
250         return this._CameraLoc;
251       }
252       set
253       {
254         if ((this._CameraLoc != value))
255         {
256           this.OnCameraLocChanging(value);
257           this.SendPropertyChanging();
258           this._CameraLoc = value;
259           this.SendPropertyChanged("CameraLoc");
260           this.OnCameraLocChanged();
261         }
262       }
263     }
264
265     [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_CameraDetails", DbType="
          VarChar(100)")]
266     public string CameraDetails
267     {
268       get
269       {
270         return this._CameraDetails;
271       }
```

```
272         set
273         {
274           if ((this._CameraDetails != value))
275           {
276             this.OnCameraDetailsChanging(value);
277             this.SendPropertyChanging();
278             this._CameraDetails = value;
279             this.SendPropertyChanged("CameraDetails");
280             this.OnCameraDetailsChanged();
281           }
282         }
283       }
284
285       [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_imgIndex", DbType="
              SmallInt")]
286       public System.Nullable<short> imgIndex
287       {
288         get
289         {
290           return this._imgIndex;
291         }
292         set
293         {
294           if ((this._imgIndex != value))
295           {
296             this.OnimgIndexChanging(value);
297             this.SendPropertyChanging();
298             this._imgIndex = value;
299             this.SendPropertyChanged("imgIndex");
300             this.OnimgIndexChanged();
301           }
302         }
303       }
304
305       [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_ThumbPath", DbType="
              VarChar(10)")]
306       public string ThumbPath
307       {
308         get
309         {
310           return this._ThumbPath;
311         }
312         set
313         {
314           if ((this._ThumbPath != value))
315           {
316             this.OnThumbPathChanging(value);
317             this.SendPropertyChanging();
318             this._ThumbPath = value;
319             this.SendPropertyChanged("ThumbPath");
320             this.OnThumbPathChanged();
321           }
322         }
323       }
324
325       [global::System.Data.Linq.Mapping.AssociationAttribute(Name="Camera_CameraStat",
              Storage="_CameraStats", ThisKey="CameraID", OtherKey="CameraID")]
326       public EntitySet<CameraStat> CameraStats
327       {
328         get
329         {
330           return this._CameraStats;
331         }
332         set
333         {
334           this._CameraStats.Assign(value);
335         }
336       }
```

```
337
338     [global::System.Data.Linq.Mapping.AssociationAttribute(Name="Camera_DetectionEvent",
            Storage="_DetectionEvents", ThisKey="CameraID", OtherKey="EventCamera")]
339     public EntitySet<DetectionEvent> DetectionEvents
340     {
341       get
342       {
343         return this._DetectionEvents;
344       }
345       set
346       {
347         this._DetectionEvents.Assign(value);
348       }
349     }
350
351     public event PropertyChangingEventHandler PropertyChanging;
352
353     public event PropertyChangedEventHandler PropertyChanged;
354
355     protected virtual void SendPropertyChanging()
356     {
357       if ((this.PropertyChanging != null))
358       {
359         this.PropertyChanging(this, emptyChangingEventArgs);
360       }
361     }
362
363     protected virtual void SendPropertyChanged(String propertyName)
364     {
365       if ((this.PropertyChanged != null))
366       {
367         this.PropertyChanged(this, new PropertyChangedEventArgs(propertyName));
368       }
369     }
370
371     private void attach_CameraStats(CameraStat entity)
372     {
373       this.SendPropertyChanging();
374       entity.Camera = this;
375     }
376
377     private void detach_CameraStats(CameraStat entity)
378     {
379       this.SendPropertyChanging();
380       entity.Camera = null;
381     }
382
383     private void attach_DetectionEvents(DetectionEvent entity)
384     {
385       this.SendPropertyChanging();
386       entity.Camera = this;
387     }
388
389     private void detach_DetectionEvents(DetectionEvent entity)
390     {
391       this.SendPropertyChanging();
392       entity.Camera = null;
393     }
394   }
395
396   [global::System.Data.Linq.Mapping.TableAttribute(Name="dbo.CameraStats")]
397   public partial class CameraStat : INotifyPropertyChanging, INotifyPropertyChanged
398   {
399
400     private static PropertyChangingEventArgs emptyChangingEventArgs = new
            PropertyChangingEventArgs(String.Empty);
401
402     private short _TimeID;
```

```
403
404     private short _CameraID;
405
406     private System.Nullable<System.TimeSpan> _LoTimeRange;
407
408     private System.Nullable<System.TimeSpan> _HiTimeRange;
409
410     private System.Nullable<double> _Simga;
411
412     private System.Nullable<double> _Mean;
413
414     private System.Nullable<int> _NValue;
415
416     private EntityRef<Camera> _Camera;
417
418     #region Extensibility Method Definitions
419     partial void OnLoaded();
420     partial void OnValidate(System.Data.Linq.ChangeAction action);
421     partial void OnCreated();
422     partial void OnTimeIDChanging(short value);
423     partial void OnTimeIDChanged();
424     partial void OnCameraIDChanging(short value);
425     partial void OnCameraIDChanged();
426     partial void OnLoTimeRangeChanging(System.Nullable<System.TimeSpan> value);
427     partial void OnLoTimeRangeChanged();
428     partial void OnHiTimeRangeChanging(System.Nullable<System.TimeSpan> value);
429     partial void OnHiTimeRangeChanged();
430     partial void OnSimgaChanging(System.Nullable<double> value);
431     partial void OnSimgaChanged();
432     partial void OnMeanChanging(System.Nullable<double> value);
433     partial void OnMeanChanged();
434     partial void OnNValueChanging(System.Nullable<int> value);
435     partial void OnNValueChanged();
436     #endregion
437
438     public CameraStat()
439     {
440       this._Camera = default(EntityRef<Camera>);
441       OnCreated();
442     }
443
444     [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_TimeID", DbType="SmallInt
          NOT NULL", IsPrimaryKey=true)]
445     public short TimeID
446     {
447       get
448       {
449         return this._TimeID;
450       }
451       set
452       {
453         if ((this._TimeID != value))
454         {
455           this.OnTimeIDChanging(value);
456           this.SendPropertyChanging();
457           this._TimeID = value;
458           this.SendPropertyChanged("TimeID");
459           this.OnTimeIDChanged();
460         }
461       }
462     }
463
464     [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_CameraID", DbType="
          SmallInt NOT NULL")]
465     public short CameraID
466     {
467       get
468       {
```

```
469             return this._CameraID;
470         }
471         set
472         {
473             if ((this._CameraID != value))
474             {
475                 if (this._Camera.HasLoadedOrAssignedValue)
476                 {
477                     throw new System.Data.Linq.ForeignKeyReferenceAlreadyHasValueException();
478                 }
479                 this.OnCameraIDChanging(value);
480                 this.SendPropertyChanging();
481                 this._CameraID = value;
482                 this.SendPropertyChanged("CameraID");
483                 this.OnCameraIDChanged();
484             }
485         }
486     }
487
488     [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_LoTimeRange", DbType="Time
        ")]
489     public System.Nullable<System.TimeSpan> LoTimeRange
490     {
491         get
492         {
493             return this._LoTimeRange;
494         }
495         set
496         {
497             if ((this._LoTimeRange != value))
498             {
499                 this.OnLoTimeRangeChanging(value);
500                 this.SendPropertyChanging();
501                 this._LoTimeRange = value;
502                 this.SendPropertyChanged("LoTimeRange");
503                 this.OnLoTimeRangeChanged();
504             }
505         }
506     }
507
508     [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_HiTimeRange", DbType="Time
        ")]
509     public System.Nullable<System.TimeSpan> HiTimeRange
510     {
511         get
512         {
513             return this._HiTimeRange;
514         }
515         set
516         {
517             if ((this._HiTimeRange != value))
518             {
519                 this.OnHiTimeRangeChanging(value);
520                 this.SendPropertyChanging();
521                 this._HiTimeRange = value;
522                 this.SendPropertyChanged("HiTimeRange");
523                 this.OnHiTimeRangeChanged();
524             }
525         }
526     }
527
528     [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_Simga", DbType="Float")]
529     public System.Nullable<double> Simga
530     {
531         get
532         {
533             return this._Simga;
534         }
```

```
535         set
536         {
537           if ((this._Simga != value))
538           {
539             this.OnSimgaChanging(value);
540             this.SendPropertyChanging();
541             this._Simga = value;
542             this.SendPropertyChanged("Simga");
543             this.OnSimgaChanged();
544           }
545         }
546       }
547
548       [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_Mean", DbType="Float")]
549       public System.Nullable<double> Mean
550       {
551         get
552         {
553           return this._Mean;
554         }
555         set
556         {
557           if ((this._Mean != value))
558           {
559             this.OnMeanChanging(value);
560             this.SendPropertyChanging();
561             this._Mean = value;
562             this.SendPropertyChanged("Mean");
563             this.OnMeanChanged();
564           }
565         }
566       }
567
568       [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_NValue", DbType="Int")]
569       public System.Nullable<int> NValue
570       {
571         get
572         {
573           return this._NValue;
574         }
575         set
576         {
577           if ((this._NValue != value))
578           {
579             this.OnNValueChanging(value);
580             this.SendPropertyChanging();
581             this._NValue = value;
582             this.SendPropertyChanged("NValue");
583             this.OnNValueChanged();
584           }
585         }
586       }
587
588       [global::System.Data.Linq.Mapping.AssociationAttribute(Name="Camera_CameraStat",
              Storage="_Camera", ThisKey="CameraID", OtherKey="CameraID", IsForeignKey=true,
              DeleteOnNull=true, DeleteRule="CASCADE")]
589       public Camera Camera
590       {
591         get
592         {
593           return this._Camera.Entity;
594         }
595         set
596         {
597           Camera previousValue = this._Camera.Entity;
598           if (((previousValue != value)
599                 || (this._Camera.HasLoadedOrAssignedValue == false)))
600           {
```

```
601             this.SendPropertyChanging();
602             if ((previousValue != null))
603             {
604               this._Camera.Entity = null;
605               previousValue.CameraStats.Remove(this);
606             }
607             this._Camera.Entity = value;
608             if ((value != null))
609             {
610               value.CameraStats.Add(this);
611               this._CameraID = value.CameraID;
612             }
613             else
614             {
615               this._CameraID = default(short);
616             }
617             this.SendPropertyChanged("Camera");
618           }
619         }
620       }
621
622       public event PropertyChangingEventHandler PropertyChanging;
623
624       public event PropertyChangedEventHandler PropertyChanged;
625
626       protected virtual void SendPropertyChanging()
627       {
628         if ((this.PropertyChanging != null))
629         {
630           this.PropertyChanging(this, emptyChangingEventArgs);
631         }
632       }
633
634       protected virtual void SendPropertyChanged(String propertyName)
635       {
636         if ((this.PropertyChanged != null))
637         {
638           this.PropertyChanged(this, new PropertyChangedEventArgs(propertyName));
639         }
640       }
641     }
642
643     [global::System.Data.Linq.Mapping.TableAttribute(Name="dbo.DetectionEvent")]
644     public partial class DetectionEvent : INotifyPropertyChanging, INotifyPropertyChanged
645     {
646
647       private static PropertyChangingEventArgs emptyChangingEventArgs = new
              PropertyChangingEventArgs(String.Empty);
648
649       private int _EventID;
650
651       private short _EventCamera;
652
653       private System.TimeSpan _EventTimeIn;
654
655       private System.Nullable<System.TimeSpan> _EventTimeOut;
656
657       private System.Nullable<int> _isChecked;
658
659       private System.Nullable<short> _EntryType;
660
661       private System.Nullable<short> _ExitType;
662
663       private int _SuspectID;
664
665       private double _Confidence;
666
667       private EntitySet<Anomaly> _Anomalies;
```

```
668
669     private EntityRef<Camera> _Camera;
670
671     #region Extensibility Method Definitions
672     partial void OnLoaded();
673     partial void OnValidate(System.Data.Linq.ChangeAction action);
674     partial void OnCreated();
675     partial void OnEventIDChanging(int value);
676     partial void OnEventIDChanged();
677     partial void OnEventCameraChanging(short value);
678     partial void OnEventCameraChanged();
679     partial void OnEventTimeInChanging(System.TimeSpan value);
680     partial void OnEventTimeInChanged();
681     partial void OnEventTimeOutChanging(System.Nullable<System.TimeSpan> value);
682     partial void OnEventTimeOutChanged();
683     partial void OnisCheckedChanging(System.Nullable<int> value);
684     partial void OnisCheckedChanged();
685     partial void OnEntryTypeChanging(System.Nullable<short> value);
686     partial void OnEntryTypeChanged();
687     partial void OnExitTypeChanging(System.Nullable<short> value);
688     partial void OnExitTypeChanged();
689     partial void OnSuspectIDChanging(int value);
690     partial void OnSuspectIDChanged();
691     partial void OnConfidenceChanging(double value);
692     partial void OnConfidenceChanged();
693     #endregion
694
695     public DetectionEvent()
696     {
697       this._Anomalies = new EntitySet<Anomaly>(new Action<Anomaly>(this.attach_Anomalies),
            new Action<Anomaly>(this.detach_Anomalies));
698       this._Camera = default(EntityRef<Camera>);
699       OnCreated();
700     }
701
702     [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_EventID", DbType="Int NOT
        NULL", IsPrimaryKey=true)]
703     public int EventID
704     {
705       get
706       {
707         return this._EventID;
708       }
709       set
710       {
711         if ((this._EventID != value))
712         {
713           this.OnEventIDChanging(value);
714           this.SendPropertyChanging();
715           this._EventID = value;
716           this.SendPropertyChanged("EventID");
717           this.OnEventIDChanged();
718         }
719       }
720     }
721
722     [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_EventCamera", DbType="
        SmallInt NOT NULL")]
723     public short EventCamera
724     {
725       get
726       {
727         return this._EventCamera;
728       }
729       set
730       {
731         if ((this._EventCamera != value))
732         {
```

```
733             if (this._Camera.HasLoadedOrAssignedValue)
734             {
735               throw new System.Data.Linq.ForeignKeyReferenceAlreadyHasValueException();
736             }
737             this.OnEventCameraChanging(value);
738             this.SendPropertyChanging();
739             this._EventCamera = value;
740             this.SendPropertyChanged("EventCamera");
741             this.OnEventCameraChanged();
742           }
743         }
744       }
745
746       [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_EventTimeIn", DbType="Time
            NOT NULL")]
747       public System.TimeSpan EventTimeIn
748       {
749         get
750         {
751           return this._EventTimeIn;
752         }
753         set
754         {
755           if ((this._EventTimeIn != value))
756           {
757             this.OnEventTimeInChanging(value);
758             this.SendPropertyChanging();
759             this._EventTimeIn = value;
760             this.SendPropertyChanged("EventTimeIn");
761             this.OnEventTimeInChanged();
762           }
763         }
764       }
765
766       [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_EventTimeOut", DbType="
            Time")]
767       public System.Nullable<System.TimeSpan> EventTimeOut
768       {
769         get
770         {
771           return this._EventTimeOut;
772         }
773         set
774         {
775           if ((this._EventTimeOut != value))
776           {
777             this.OnEventTimeOutChanging(value);
778             this.SendPropertyChanging();
779             this._EventTimeOut = value;
780             this.SendPropertyChanged("EventTimeOut");
781             this.OnEventTimeOutChanged();
782           }
783         }
784       }
785
786       [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_isChecked", DbType="Int")]
787       public System.Nullable<int> isChecked
788       {
789         get
790         {
791           return this._isChecked;
792         }
793         set
794         {
795           if ((this._isChecked != value))
796           {
797             this.OnisCheckedChanging(value);
798             this.SendPropertyChanging();
```

```
            this._isChecked = value;
            this.SendPropertyChanged("isChecked");
            this.OnisCheckedChanged();
        }
      }
    }

    [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_EntryType", DbType="
        SmallInt")]
    public System.Nullable<short> EntryType
    {
      get
      {
        return this._EntryType;
      }
      set
      {
        if ((this._EntryType != value))
        {
          this.OnEntryTypeChanging(value);
          this.SendPropertyChanging();
          this._EntryType = value;
          this.SendPropertyChanged("EntryType");
          this.OnEntryTypeChanged();
        }
      }
    }

    [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_ExitType", DbType="
        SmallInt")]
    public System.Nullable<short> ExitType
    {
      get
      {
        return this._ExitType;
      }
      set
      {
        if ((this._ExitType != value))
        {
          this.OnExitTypeChanging(value);
          this.SendPropertyChanging();
          this._ExitType = value;
          this.SendPropertyChanged("ExitType");
          this.OnExitTypeChanged();
        }
      }
    }

    [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_SuspectID", DbType="Int
        NOT NULL")]
    public int SuspectID
    {
      get
      {
        return this._SuspectID;
      }
      set
      {
        if ((this._SuspectID != value))
        {
          this.OnSuspectIDChanging(value);
          this.SendPropertyChanging();
          this._SuspectID = value;
          this.SendPropertyChanged("SuspectID");
          this.OnSuspectIDChanged();
        }
      }
```

```
864     }
865
866     [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_Confidence", DbType="Float
            NOT NULL")]
867     public double Confidence
868     {
869       get
870       {
871         return this._Confidence;
872       }
873       set
874       {
875         if ((this._Confidence != value))
876         {
877           this.OnConfidenceChanging(value);
878           this.SendPropertyChanging();
879           this._Confidence = value;
880           this.SendPropertyChanged("Confidence");
881           this.OnConfidenceChanged();
882         }
883       }
884     }
885
886     [global::System.Data.Linq.Mapping.AssociationAttribute(Name="DetectionEvent_Anomaly",
            Storage="_Anomalies", ThisKey="EventID", OtherKey="EventID")]
887     public EntitySet<Anomaly> Anomalies
888     {
889       get
890       {
891         return this._Anomalies;
892       }
893       set
894       {
895         this._Anomalies.Assign(value);
896       }
897     }
898
899     [global::System.Data.Linq.Mapping.AssociationAttribute(Name="Camera_DetectionEvent",
            Storage="_Camera", ThisKey="EventCamera", OtherKey="CameraID", IsForeignKey=true,
            DeleteOnNull=true, DeleteRule="CASCADE")]
900     public Camera Camera
901     {
902       get
903       {
904         return this._Camera.Entity;
905       }
906       set
907       {
908         Camera previousValue = this._Camera.Entity;
909         if (((previousValue != value)
910               || (this._Camera.HasLoadedOrAssignedValue == false)))
911         {
912           this.SendPropertyChanging();
913           if ((previousValue != null))
914           {
915             this._Camera.Entity = null;
916             previousValue.DetectionEvents.Remove(this);
917           }
918           this._Camera.Entity = value;
919           if ((value != null))
920           {
921             value.DetectionEvents.Add(this);
922             this._EventCamera = value.CameraID;
923           }
924           else
925           {
926             this._EventCamera = default(short);
927           }
```

```
928             this.SendPropertyChanged("Camera");
929           }
930         }
931       }
932
933       public event PropertyChangingEventHandler PropertyChanging;
934
935       public event PropertyChangedEventHandler PropertyChanged;
936
937       protected virtual void SendPropertyChanging()
938       {
939         if ((this.PropertyChanging != null))
940         {
941           this.PropertyChanging(this, emptyChangingEventArgs);
942         }
943       }
944
945       protected virtual void SendPropertyChanged(String propertyName)
946       {
947         if ((this.PropertyChanged != null))
948         {
949           this.PropertyChanged(this, new PropertyChangedEventArgs(propertyName));
950         }
951       }
952
953       private void attach_Anomalies(Anomaly entity)
954       {
955         this.SendPropertyChanging();
956         entity.DetectionEvent = this;
957       }
958
959       private void detach_Anomalies(Anomaly entity)
960       {
961         this.SendPropertyChanging();
962         entity.DetectionEvent = null;
963       }
964     }
965
966     [global::System.Data.Linq.Mapping.TableAttribute(Name="dbo.Anomaly")]
967     public partial class Anomaly : INotifyPropertyChanging, INotifyPropertyChanged
968     {
969
970       private static PropertyChangingEventArgs emptyChangingEventArgs = new
              PropertyChangingEventArgs(String.Empty);
971
972       private int _AnomalyID;
973
974       private int _EventID;
975
976       private System.Nullable<double> _Severity;
977
978       private System.Nullable<short> _isChecked;
979
980       private System.Nullable<short> _AnomalyType;
981
982       private EntityRef<DetectionEvent> _DetectionEvent;
983
984       #region Extensibility Method Definitions
985       partial void OnLoaded();
986       partial void OnValidate(System.Data.Linq.ChangeAction action);
987       partial void OnCreated();
988       partial void OnAnomalyIDChanging(int value);
989       partial void OnAnomalyIDChanged();
990       partial void OnEventIDChanging(int value);
991       partial void OnEventIDChanged();
992       partial void OnSeverityChanging(System.Nullable<double> value);
993       partial void OnSeverityChanged();
994       partial void OnisCheckedChanging(System.Nullable<short> value);
```

```
995     partial void OnisCheckedChanged();
996     partial void OnAnomalyTypeChanging(System.Nullable<short> value);
997     partial void OnAnomalyTypeChanged();
998     #endregion
999
1000    public Anomaly()
1001    {
1002      this._DetectionEvent = default(EntityRef<DetectionEvent>);
1003      OnCreated();
1004    }
1005
1006    [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_AnomalyID", AutoSync=
          AutoSync.OnInsert, DbType="Int NOT NULL IDENTITY", IsPrimaryKey=true,
          IsDbGenerated=true)]
1007    public int AnomalyID
1008    {
1009      get
1010      {
1011        return this._AnomalyID;
1012      }
1013      set
1014      {
1015        if ((this._AnomalyID != value))
1016        {
1017          this.OnAnomalyIDChanging(value);
1018          this.SendPropertyChanging();
1019          this._AnomalyID = value;
1020          this.SendPropertyChanged("AnomalyID");
1021          this.OnAnomalyIDChanged();
1022        }
1023      }
1024    }
1025
1026    [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_EventID", DbType="Int NOT
          NULL")]
1027    public int EventID
1028    {
1029      get
1030      {
1031        return this._EventID;
1032      }
1033      set
1034      {
1035        if ((this._EventID != value))
1036        {
1037          if (this._DetectionEvent.HasLoadedOrAssignedValue)
1038          {
1039            throw new System.Data.Linq.ForeignKeyReferenceAlreadyHasValueException();
1040          }
1041          this.OnEventIDChanging(value);
1042          this.SendPropertyChanging();
1043          this._EventID = value;
1044          this.SendPropertyChanged("EventID");
1045          this.OnEventIDChanged();
1046        }
1047      }
1048    }
1049
1050    [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_Severity", DbType="Float")
          ]
1051    public System.Nullable<double> Severity
1052    {
1053      get
1054      {
1055        return this._Severity;
1056      }
1057      set
1058      {
```

```
1059            if ((this._Severity != value))
1060            {
1061              this.OnSeverityChanging(value);
1062              this.SendPropertyChanging();
1063              this._Severity = value;
1064              this.SendPropertyChanged("Severity");
1065              this.OnSeverityChanged();
1066            }
1067          }
1068        }
1069
1070        [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_isChecked", DbType="
              SmallInt")]
1071        public System.Nullable<short> isChecked
1072        {
1073          get
1074          {
1075            return this._isChecked;
1076          }
1077          set
1078          {
1079            if ((this._isChecked != value))
1080            {
1081              this.OnisCheckedChanging(value);
1082              this.SendPropertyChanging();
1083              this._isChecked = value;
1084              this.SendPropertyChanged("isChecked");
1085              this.OnisCheckedChanged();
1086            }
1087          }
1088        }
1089
1090        [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_AnomalyType", DbType="
              SmallInt")]
1091        public System.Nullable<short> AnomalyType
1092        {
1093          get
1094          {
1095            return this._AnomalyType;
1096          }
1097          set
1098          {
1099            if ((this._AnomalyType != value))
1100            {
1101              this.OnAnomalyTypeChanging(value);
1102              this.SendPropertyChanging();
1103              this._AnomalyType = value;
1104              this.SendPropertyChanged("AnomalyType");
1105              this.OnAnomalyTypeChanged();
1106            }
1107          }
1108        }
1109
1110        [global::System.Data.Linq.Mapping.AssociationAttribute(Name="DetectionEvent_Anomaly",
              Storage="_DetectionEvent", ThisKey="EventID", OtherKey="EventID", IsForeignKey=
              true, DeleteOnNull=true, DeleteRule="CASCADE")]
1111        public DetectionEvent DetectionEvent
1112        {
1113          get
1114          {
1115            return this._DetectionEvent.Entity;
1116          }
1117          set
1118          {
1119            DetectionEvent previousValue = this._DetectionEvent.Entity;
1120            if (((previousValue != value)
1121                  || (this._DetectionEvent.HasLoadedOrAssignedValue == false)))
1122            {
```

```
            this.SendPropertyChanging();
            if ((previousValue != null))
            {
              this._DetectionEvent.Entity = null;
              previousValue.Anomalies.Remove(this);
            }
            this._DetectionEvent.Entity = value;
            if ((value != null))
            {
              value.Anomalies.Add(this);
              this._EventID = value.EventID;
            }
            else
            {
              this._EventID = default(int);
            }
            this.SendPropertyChanged("DetectionEvent");
          }
        }
    }

    public event PropertyChangingEventHandler PropertyChanging;

    public event PropertyChangedEventHandler PropertyChanged;

    protected virtual void SendPropertyChanging()
    {
      if ((this.PropertyChanging != null))
      {
        this.PropertyChanging(this, emptyChangingEventArgs);
      }
    }

    protected virtual void SendPropertyChanged(String propertyName)
    {
      if ((this.PropertyChanged != null))
      {
        this.PropertyChanged(this, new PropertyChangedEventArgs(propertyName));
      }
    }
  }
}
#pragma warning restore 1591
```

Listing 16.27: Database Configuration File

16.8 User Interface - Primary Form

```
using System;
using System.Collections.Generic;
using System.Drawing;
using System.Windows.Forms;
using System.Linq;
using System.Data.Linq;
using System.Data.Linq.Mapping;
using System.Net;
using System.IO;
using System.Threading;
using System.ComponentModel;


namespace Senior_Design_UI
{
    public partial class Form1 : Form
    {
```

```
18          Form StreamAddForm;
19          delegate void SetTextCallback(string text);
20          delegate void SetUICallback();
21          System.Threading.Thread DBThread;
22
23
24          /******************************** LOADING FUNCTIONS
                ************************************/
25          public Form1()
26          {
27              InitializeComponent();
28          }
29
30          private void Form1_Load(object sender, EventArgs e)
31          {
32              /** Basic Options and links being established **/
33              StreamListView.SmallImageList = Globals.StreamThumbsList;
34              ObjectListView.SmallImageList = Globals.EventThumbsList;
35              Globals.StreamThumbsList.ImageSize = new Size(95,60);
36              Globals.EventThumbsList.ImageSize = new Size(95, 60);
37              VideoStream.CtlVisible = false;
38              VideoStream.Update();
39              DBThread = new System.Threading.Thread(new System.Threading.ThreadStart(
                    StartDBCalls));
40              DBThread.Start();
41
42              /** Setting up resizing options and different screen options **/
43              StreamListMenu.Size = new Size(StreamListMenu.Size.Width, (RightMenu.Size.
                    Height / 2) - 12);
44              ObjectList.Size = new Size(ObjectList.Size.Width, (RightMenu.Size.Height / 2)
                    - 12);
45              NoteLog.Location = new Point(NoteLog.Location.X,(RightMenu.Size.Height +
                    menuStrip1.Size.Height - NoteLog.Size.Height) - 5);
46              VideoStream.Size = new Size( RightMenu.Location.X - 24, NoteLog.Location.Y -
                    12 - menuStrip1.Size.Height );
47              Identity.Location = new Point(NoteLog.Size.Width + NoteLog.Location.X + 12,
                    NoteLog.Location.Y);
48              if(RightMenu.Location.X > 800)
49              {
50                  NoteLog.Size = new Size((RightMenu.Location.X - Identity.Size.Width),
                        NoteLog.Size.Height);
51                  Identity.Size = new Size((RightMenu.Location.X - NoteLog.Size.Width) - 36
                        , Identity.Size.Height);
52                  Identity.Location = new Point(NoteLog.Size.Width + NoteLog.Location.X +
                        12, Identity.Location.Y);
53                  Identity.Show();
54              }
55              else
56              {
57                  NoteLog.Size = new Size(RightMenu.Location.X - 12, NoteLog.Size.Height);
58                  Identity.Hide();
59              }
60              LoadDBCams();
61          }
62
63          private void Form1_FormClosing(object sender, FormClosingEventArgs e)
64          {
65              DBThread.Abort();
66          }
67
68          private void LoadDBCams()
69          {
70              SDDBDataContext DB = new SDDBDataContext();
71
72              var Cameras = (from cams in DB.GetTable<Camera>() select cams);
73
74              foreach (Camera DBCam in Cameras)
75              {
```

```
76                  try
77                  {
78                      Image NewThumb = Image.FromFile(DBCam.ThumbPath);
79                      Globals.StreamThumbsList.Images.Add(NewThumb);
80                  }
81                  catch
82                  {
83                      Image FailedImg = Image.FromFile("C:/Users/nfox/Source/Repos/
                            SmartCameras/Senior Design UI/Senior Design UI/noImageAvailable.
                            jpg");
84                      Globals.StreamThumbsList.Images.Add(FailedImg);
85                  }
86
87                  RSTPStream CamFromDB = new RSTPStream(DBCam.CameraIP, DBCam.CameraDetails,
                        DBCam.CameraLoc, Globals.StreamList.Count + 1);
88                  Globals.StreamList.Add(CamFromDB);
89              }
90              UpdateStreamListBox();
91          }
92
93
94          /*****************************************Event Functions
                ****************************************/
95          private void addStreamToolStripMenuItem_Click(object sender, EventArgs e)
96          {
97              StreamAddForm = new AddStream();
98              StreamAddForm.ShowDialog();
99              if (StreamAddForm.DialogResult == DialogResult.OK)
100             {
101                 label1.Text = "Start" + Globals.StreamList.Count + " : ";
102                 StreamThumbRefresh(Globals.StreamList[Globals.StreamList.Count - 1]);
103             }
104         }
105
106         private void StreamThumbRefresh(RSTPStream thisStream)
107         {
108
109             //Image Newthumb = Image.FromFile();
110             //Globals.SteamThumbsList.Images[thisStream.get_imgIndex()] = NewThumb;
111
112             /* Not Working Yet
113             Uri ThumbUrl = new Uri(thisStream.get_url() + "/dsm?");
114             WebRequest ThumbNailRequest = WebRequest.Create(ThumbUrl);
115             WebResponse ThumbNailReponse = ThumbNailRequest.GetResponse();
116
117             if( ThumbNailReponse.ResponseUri.AbsoluteUri != null)
118             {
119                 Image NewThumb = Image.FromStream(ThumbNailReponse.GetResponseStream());
120                 Globals.SteamThumbsList.Images[thisStream.get_imgIndex()] = NewThumb;
121             }*/
122
123             UpdateStreamListBox();
124         }
125
126         private void UpdateStreamListBox()
127         {
128             for (int i = 0; i < Globals.StreamList.Count; i++ )
129             {
130                 ListViewItem thisColumn = new ListViewItem("", i);
131                 thisColumn.SubItems.Add(Globals.StreamList[i].get_loc());
132                 thisColumn.SubItems.Add(Globals.StreamList[i].get_details());
133                 try
134                 {
135                     StreamListView.Items[i] = thisColumn;
136                 }
137                 catch (ArgumentOutOfRangeException e)
138                 {
139                     StreamListView.Items.Add(thisColumn);
```

```
140                 }
141             }
142         }
143 
144         private void Form1_ResizeBegin(object sender, EventArgs e)
145         {
146             StreamListMenu.Size = new Size(StreamListMenu.Size.Width, (RightMenu.Size.
                    Height / 2) - 12);
147             ObjectList.Size = new Size(ObjectList.Size.Width, (RightMenu.Size.Height / 2)
                    - 12);
148             NoteLog.Location = new Point(NoteLog.Location.X, (RightMenu.Size.Height +
                    menuStrip1.Size.Height - NoteLog.Size.Height) - 5);
149             if (RightMenu.Location.X > 800)
150             {
151                 NoteLog.Size = new Size((RightMenu.Location.X - Identity.Size.Width) -36,
                        NoteLog.Size.Height);
152                 Identity.Size = new Size(300, Identity.Size.Height);
153                 Identity.Location = new Point(NoteLog.Size.Width + NoteLog.Location.X +
                        12, Identity.Location.Y);
154                 Identity.Show();
155             }
156             else
157             {
158                 NoteLog.Size = new Size(RightMenu.Location.X - 12, NoteLog.Size.Height);
159                 Identity.Hide();
160             }
161         }
162 
163         private void refreshIconsToolStripMenuItem_Click(object sender, EventArgs e)
164         {
165             UpdateStreamListBox();
166         }
167 
168 
169         private void StreamListView_SelectedIndexChanged(object sender, EventArgs e)
170         {
171             ListView.SelectedListViewItemCollection indexes = this.StreamListView.
                    SelectedItems;
172             int index = 0;
173 
174             foreach (ListViewItem item in indexes)
175             {
176                 if (item.Selected)
177                 {
178                     index = item.Index;
179                     break;
180                 }
181             }
182 
183             String VideoUrl = "rtsp://" + Globals.StreamList[index].get_url() + "/live1.
                    sdp";
184             VideoStream.playlist.add(VideoUrl);
185             VideoStream.playlist.next();
186             VideoStream.playlist.play();
187         }
188 
189 
190         /************************************ DATABASE FUNCTIONS
                ************************************/
191         //Database Continous Calls checking for anomalies
192         private void StartDBCalls()
193         {
194             SDDBDataContext SDDatabase = new SDDBDataContext();
195             DateTime CurrTime = DateTime.Now;
196 
197             while (SDDatabase.DatabaseExists())
198             {
199                 if(CurrTime <= DateTime.Now)
```

```
200                 {
201                     //SafeSetText(DateTime.Now.TimeOfDay + "Queries started");
202                     UpdateAnoms(SDDatabase);
203                     UpdateEvents(SDDatabase);
204                     CurrTime = CurrTime.AddMilliseconds(1500);
205                     //SafeSetText( DateTime.Now.TimeOfDay + "Queries completed");
206                 }
207                 else
208                 {
209                     System.Threading.Thread.Sleep(500);
210                 }
211             }
212         }
213 
214         private void UpdateAnoms(SDDBDataContext SDDB)
215         {
216             var newAnomalies = (from anom in SDDB.GetTable<Anomaly>() where (anom.
                    isChecked == 0) select anom);
217             String LogOutput = "";
218 
219             foreach (Anomaly anom in newAnomalies)
220             {
221                 try
222                 {
223                     anom.isChecked = 1;
224                     SDDB.SubmitChanges();
225                     var eventID = SDDB.DetectionEvents.Single(x => x.EventID == anom.
                            EventID);
226                     LogOutput = DateTime.Now.TimeOfDay + ": Anomaly Detected in Camera: "
                            + SDDB.Cameras.Single( x => x.CameraID == eventID.Camera.CameraID)
                            .CameraLoc + " with individual: " + SDDB.Individuals.Single(x => x
                            .ID == eventID.SuspectID).Name + "\n";
227                 }
228                 catch (Exception e)
229                 {
230                     LogOutput = DateTime.Now.TimeOfDay + ": Database Connection Failed,
                            Please Check Connection. Details: " + e + "\n";
231                 }
232                 SafeSetText(LogOutput);
233             }
234         }
235 
236         private void UpdateEvents(SDDBDataContext SDDB)
237         {
238             var OngoingEvents = (from events in SDDB.GetTable<DetectionEvent>() where (
                    events.EventTimeIn != null && events.EventTimeOut == null) select events);
239             bool UpdateFlag = false;
240 
241             foreach(DetectionEvent events in OngoingEvents)
242             {
243                 string IDName = SDDB.Individuals.Single(v => v.ID == events.SuspectID).
                        Name;
244                 if (Globals.EventList.Exists(x => x.EventID == events.EventID))
245                     continue;
246                 UpdateFlag = true;
247                 EventObject newEvent = new EventObject(IDName, (Int32)events.EventCamera,
                        Globals.EventList.Count, (Int32)events.EventID, 0, 0);
248                 Globals.EventList.Add(newEvent);
249                 Image FailedImg = Image.FromFile("C:/Users/nfox/Source/Repos/SmartCameras/
                        Senior Design UI/Senior Design UI/noImageAvailable.jpg");
250                 Globals.EventThumbsList.Images.Add(FailedImg);
251             }
252             if(UpdateFlag)
253                 UpdateUISafe();
254         }
255 
256         private void UpdateUISafe()
257         {
```

```
258             if(this.ObjectListView.InvokeRequired)
259             {
260                 SetUICallback d = new SetUICallback(UpdateUISafe);
261                 this.Invoke(d, new object[] {});
262             }
263             else
264             {
265                 for (int i = 0; i < Globals.EventList.Count; i++ )
266                 {
267                     ListViewItem thisColumn = new ListViewItem("", i);
268                     thisColumn.SubItems.Add(Globals.EventList[i].Individ);
269                     thisColumn.SubItems.Add(Globals.StreamList[(Globals.EventList[i].
                            CameraID - 1)].get_loc());
270                     thisColumn.SubItems.Add(Globals.EventList[i].X.ToString());
271                     thisColumn.SubItems.Add(Globals.EventList[i].Y.ToString());
272                     thisColumn.SubItems.Add("0");
273                     try
274                     {
275                         ObjectListView.Items[i] = thisColumn;
276                     }
277                     catch (ArgumentOutOfRangeException e)
278                     {
279                         ObjectListView.Items.Add(thisColumn);
280                     }
281                 }
282             }
283 
284         }
285 
286         private void SafeSetText(String SetText)
287         {
288             if (this.NotificationLog.InvokeRequired)
289             {
290                 SetTextCallback d = new SetTextCallback(SafeSetText);
291                 this.Invoke(d, new object[] { SetText });
292             }
293             else
294                 this.NotificationLog.AppendText(SetText + "\n");
295         }
296 
297         private void textBox1_Enter(object sender, EventArgs e)
298         {
299 
300             NotificationLog.AppendText( DateTime.Now.TimeOfDay + ": " + textEnterLine.Text
                    + "\n");
301             textEnterLine.Text = "";
302         }
303 
304         private void NotificationLog_TextChanged(object sender, EventArgs e)
305         {
306             SDDBDataContext SDDB = new SDDBDataContext();
307             var Array = NotificationLog.Text.Split('\n');
308             if (SDDB.Log == null)
309                 return;
310             try
311             {
312                 SDDB.Log.NewLine = Array[Array.Length - 1];
313             }
314             catch
315             {
316                 SDDB.Log.NewLine = "Bug in TextLog, revisit later";
317             }
318         }
319 
320 
321     }
322 
323     public class Globals
```

```
    {
        internal static List<RSTPStream> StreamList = new  List<RSTPStream>();
        public static ImageList StreamThumbsList = new ImageList();
        internal static List<EventObject> EventList = new List<EventObject>();
        public static ImageList EventThumbsList = new ImageList();
    }

}
```

Listing 16.28: Code for the Primary Form of the User Interface

16.9 User Interface - Primary Form Designer File

```
namespace Senior_Design_UI
{
    partial class Form1
    {
        /// <summary>
        /// Required designer variable.
        /// </summary>
        private System.ComponentModel.IContainer components = null;

        /// <summary>
        /// Clean up any resources being used.
        /// </summary>
        /// <param name="disposing">true if managed resources should be disposed;
            otherwise, false.</param>
        protected override void Dispose(bool disposing)
        {
            if (disposing && (components != null))
            {
                components.Dispose();
            }
            base.Dispose(disposing);
        }

        #region Windows Form Designer generated code

        /// <summary>
        /// Required method for Designer support - do not modify
        /// the contents of this method with the code editor.
        /// </summary>
        private void InitializeComponent()
        {
            this.components = new System.ComponentModel.Container();
            System.ComponentModel.ComponentResourceManager resources = new System.
                ComponentModel.ComponentResourceManager(typeof(Form1));
            this.NoteLog = new System.Windows.Forms.GroupBox();
            this.NotificationLog = new System.Windows.Forms.RichTextBox();
            this.textEnterLine = new System.Windows.Forms.TextBox();
            this.label1 = new System.Windows.Forms.Label();
            this.VideoStream = new AxAXVLC.AxVLCPlugin2();
            this.menuStrip1 = new System.Windows.Forms.MenuStrip();
            this.fileToolStripMenuItem = new System.Windows.Forms.ToolStripMenuItem();
            this.streamsToolStripMenuItem = new System.Windows.Forms.ToolStripMenuItem();
            this.addStreamToolStripMenuItem = new System.Windows.Forms.ToolStripMenuItem()
                ;
            this.refreshIconsToolStripMenuItem = new System.Windows.Forms.
                ToolStripMenuItem();
            this.imageList1 = new System.Windows.Forms.ImageList(this.components);
            this.StreamListView = new System.Windows.Forms.ListView();
            this.columnHeader6 = ((System.Windows.Forms.ColumnHeader)(new System.Windows.
                Forms.ColumnHeader())));
            this.columnHeader7 = ((System.Windows.Forms.ColumnHeader)(new System.Windows.
                Forms.ColumnHeader())));
```

```
47              this.columnHeader8 = ((System.Windows.Forms.ColumnHeader)(new System.Windows.
                    Forms.ColumnHeader()));
48              this.StreamListMenu = new System.Windows.Forms.GroupBox();
49              this.ObjectList = new System.Windows.Forms.GroupBox();
50              this.ObjectListView = new System.Windows.Forms.ListView();
51              this.columnhead1 = ((System.Windows.Forms.ColumnHeader)(new System.Windows.
                    Forms.ColumnHeader()));
52              this.columnHeader2 = ((System.Windows.Forms.ColumnHeader)(new System.Windows.
                    Forms.ColumnHeader()));
53              this.columnHeader1 = ((System.Windows.Forms.ColumnHeader)(new System.Windows.
                    Forms.ColumnHeader()));
54              this.columnHeader3 = ((System.Windows.Forms.ColumnHeader)(new System.Windows.
                    Forms.ColumnHeader()));
55              this.columnHeader4 = ((System.Windows.Forms.ColumnHeader)(new System.Windows.
                    Forms.ColumnHeader()));
56              this.columnHeader5 = ((System.Windows.Forms.ColumnHeader)(new System.Windows.
                    Forms.ColumnHeader()));
57              this.RightMenu = new System.Windows.Forms.GroupBox();
58              this.pictureBox1 = new System.Windows.Forms.PictureBox();
59              this.Identity = new System.Windows.Forms.GroupBox();
60              this.NoteLog.SuspendLayout();
61              ((System.ComponentModel.ISupportInitialize)(this.VideoStream)).BeginInit();
62              this.menuStrip1.SuspendLayout();
63              this.StreamListMenu.SuspendLayout();
64              this.ObjectList.SuspendLayout();
65              this.RightMenu.SuspendLayout();
66              ((System.ComponentModel.ISupportInitialize)(this.pictureBox1)).BeginInit();
67              this.Identity.SuspendLayout();
68              this.SuspendLayout();
69              //
70              // NoteLog
71              //
72              this.NoteLog.Anchor = ((System.Windows.Forms.AnchorStyles)(((System.Windows.
                    Forms.AnchorStyles.Bottom | System.Windows.Forms.AnchorStyles.Left)
73              | System.Windows.Forms.AnchorStyles.Right)));
74              this.NoteLog.AutoSize = true;
75              this.NoteLog.Controls.Add(this.NotificationLog);
76              this.NoteLog.Controls.Add(this.textEnterLine);
77              this.NoteLog.Location = new System.Drawing.Point(9, 596);
78              this.NoteLog.Margin = new System.Windows.Forms.Padding(2);
79              this.NoteLog.Name = "NoteLog";
80              this.NoteLog.Padding = new System.Windows.Forms.Padding(2);
81              this.NoteLog.Size = new System.Drawing.Size(721, 193);
82              this.NoteLog.TabIndex = 2;
83              this.NoteLog.TabStop = false;
84              this.NoteLog.Text = "Notification Log";
85              //
86              // NotificationLog
87              //
88              this.NotificationLog.Dock = System.Windows.Forms.DockStyle.Top;
89              this.NotificationLog.Location = new System.Drawing.Point(2, 15);
90              this.NotificationLog.Margin = new System.Windows.Forms.Padding(2);
91              this.NotificationLog.MaxLength = 50000;
92              this.NotificationLog.Name = "NotificationLog";
93              this.NotificationLog.ReadOnly = true;
94              this.NotificationLog.ScrollBars = System.Windows.Forms.RichTextBoxScrollBars.
                    ForcedVertical;
95              this.NotificationLog.Size = new System.Drawing.Size(717, 154);
96              this.NotificationLog.TabIndex = 1;
97              this.NotificationLog.Text = "";
98              this.NotificationLog.TextChanged += new System.EventHandler(this.
                    NotificationLog_TextChanged);
99              //
100             // textEnterLine
101             //
102             this.textEnterLine.Dock = System.Windows.Forms.DockStyle.Bottom;
103             this.textEnterLine.Location = new System.Drawing.Point(2, 171);
104             this.textEnterLine.Margin = new System.Windows.Forms.Padding(2);
```

```
105             this.textEnterLine.Name = "textEnterLine";
106             this.textEnterLine.Size = new System.Drawing.Size(717, 20);
107             this.textEnterLine.TabIndex = 0;
108             this.textEnterLine.Leave += new System.EventHandler(this.textBox1_Enter);
109             //
110             // label1
111             //
112             this.label1.AutoSize = true;
113             this.label1.Location = new System.Drawing.Point(18, 25);
114             this.label1.Margin = new System.Windows.Forms.Padding(2, 0, 2, 0);
115             this.label1.Name = "label1";
116             this.label1.Size = new System.Drawing.Size(56, 13);
117             this.label1.TabIndex = 5;
118             this.label1.Text = "Testlabel1";
119             this.label1.Visible = false;
120             //
121             // VideoStream
122             //
123             this.VideoStream.Anchor = ((System.Windows.Forms.AnchorStyles)((((System.
                    Windows.Forms.AnchorStyles.Top | System.Windows.Forms.AnchorStyles.Bottom)
124             | System.Windows.Forms.AnchorStyles.Left)
125             | System.Windows.Forms.AnchorStyles.Right)));
126             this.VideoStream.Enabled = true;
127             this.VideoStream.Location = new System.Drawing.Point(11, 27);
128             this.VideoStream.Margin = new System.Windows.Forms.Padding(2);
129             this.VideoStream.Name = "VideoStream";
130             this.VideoStream.OcxState = ((System.Windows.Forms.AxHost.State)(resources.
                    GetObject("VideoStream.OcxState")));
131             this.VideoStream.Size = new System.Drawing.Size(1441, 700);
132             this.VideoStream.TabIndex = 8;
133             //
134             // menuStrip1
135             //
136             this.menuStrip1.Items.AddRange(new System.Windows.Forms.ToolStripItem[] {
137             this.fileToolStripMenuItem,
138             this.streamsToolStripMenuItem});
139             this.menuStrip1.Location = new System.Drawing.Point(0, 0);
140             this.menuStrip1.Name = "menuStrip1";
141             this.menuStrip1.Padding = new System.Windows.Forms.Padding(4, 2, 0, 2);
142             this.menuStrip1.Size = new System.Drawing.Size(1426, 24);
143             this.menuStrip1.TabIndex = 10;
144             this.menuStrip1.Text = "menuStrip1";
145             //
146             // fileToolStripMenuItem
147             //
148             this.fileToolStripMenuItem.Name = "fileToolStripMenuItem";
149             this.fileToolStripMenuItem.Size = new System.Drawing.Size(37, 20);
150             this.fileToolStripMenuItem.Text = "File";
151             //
152             // streamsToolStripMenuItem
153             //
154             this.streamsToolStripMenuItem.DropDownItems.AddRange(new System.Windows.Forms.
                    ToolStripItem[] {
155             this.addStreamToolStripMenuItem,
156             this.refreshIconsToolStripMenuItem});
157             this.streamsToolStripMenuItem.Name = "streamsToolStripMenuItem";
158             this.streamsToolStripMenuItem.Size = new System.Drawing.Size(61, 20);
159             this.streamsToolStripMenuItem.Text = "Streams";
160             //
161             // addStreamToolStripMenuItem
162             //
163             this.addStreamToolStripMenuItem.Name = "addStreamToolStripMenuItem";
164             this.addStreamToolStripMenuItem.Size = new System.Drawing.Size(144, 22);
165             this.addStreamToolStripMenuItem.Text = "Add Stream";
166             this.addStreamToolStripMenuItem.Click += new System.EventHandler(this.
                    addStreamToolStripMenuItem_Click);
167             //
168             // refreshIconsToolStripMenuItem
```

```
169             //
170             this.refreshIconsToolStripMenuItem.Name = "refreshIconsToolStripMenuItem";
171             this.refreshIconsToolStripMenuItem.Size = new System.Drawing.Size(144, 22);
172             this.refreshIconsToolStripMenuItem.Text = "Refresh Icons";
173             this.refreshIconsToolStripMenuItem.Click += new System.EventHandler(this.
                    refreshIconsToolStripMenuItem_Click);
174             //
175             // imageList1
176             //
177             this.imageList1.ColorDepth = System.Windows.Forms.ColorDepth.Depth8Bit;
178             this.imageList1.ImageSize = new System.Drawing.Size(80, 80);
179             this.imageList1.TransparentColor = System.Drawing.Color.Transparent;
180             //
181             // StreamListView
182             //
183             this.StreamListView.Columns.AddRange(new System.Windows.Forms.ColumnHeader[] {
184             this.columnHeader6,
185             this.columnHeader7,
186             this.columnHeader8});
187             this.StreamListView.Dock = System.Windows.Forms.DockStyle.Fill;
188             this.StreamListView.FullRowSelect = true;
189             this.StreamListView.GridLines = true;
190             this.StreamListView.Location = new System.Drawing.Point(2, 15);
191             this.StreamListView.Margin = new System.Windows.Forms.Padding(2);
192             this.StreamListView.MultiSelect = false;
193             this.StreamListView.Name = "StreamListView";
194             this.StreamListView.Size = new System.Drawing.Size(420, 347);
195             this.StreamListView.SmallImageList = this.imageList1;
196             this.StreamListView.TabIndex = 11;
197             this.StreamListView.TileSize = new System.Drawing.Size(400, 65);
198             this.StreamListView.UseCompatibleStateImageBehavior = false;
199             this.StreamListView.View = System.Windows.Forms.View.Details;
200             this.StreamListView.ItemActivate += new System.EventHandler(this.
                    StreamListView_SelectedIndexChanged);
201             this.StreamListView.SelectedIndexChanged += new System.EventHandler(this.
                    StreamListView_SelectedIndexChanged);
202             this.StreamListView.Click += new System.EventHandler(this.
                    StreamListView_SelectedIndexChanged);
203             //
204             // columnHeader6
205             //
206             this.columnHeader6.Text = "Thumb Nail";
207             this.columnHeader6.Width = 100;
208             //
209             // columnHeader7
210             //
211             this.columnHeader7.Text = "Stream Name";
212             this.columnHeader7.Width = 100;
213             //
214             // columnHeader8
215             //
216             this.columnHeader8.Text = "Details";
217             this.columnHeader8.Width = 216;
218             //
219             // StreamListMenu
220             //
221             this.StreamListMenu.Controls.Add(this.StreamListView);
222             this.StreamListMenu.Dock = System.Windows.Forms.DockStyle.Top;
223             this.StreamListMenu.Location = new System.Drawing.Point(2, 15);
224             this.StreamListMenu.Margin = new System.Windows.Forms.Padding(2);
225             this.StreamListMenu.Name = "StreamListMenu";
226             this.StreamListMenu.Padding = new System.Windows.Forms.Padding(2);
227             this.StreamListMenu.Size = new System.Drawing.Size(424, 364);
228             this.StreamListMenu.TabIndex = 12;
229             this.StreamListMenu.TabStop = false;
230             this.StreamListMenu.Text = "Stream List";
231             //
232             // ObjectList
```

```
            //
            this.ObjectList.Controls.Add(this.ObjectListView);
            this.ObjectList.Dock = System.Windows.Forms.DockStyle.Bottom;
            this.ObjectList.Location = new System.Drawing.Point(2, 382);
            this.ObjectList.Margin = new System.Windows.Forms.Padding(2);
            this.ObjectList.Name = "ObjectList";
            this.ObjectList.Padding = new System.Windows.Forms.Padding(2);
            this.ObjectList.Size = new System.Drawing.Size(424, 387);
            this.ObjectList.TabIndex = 13;
            this.ObjectList.TabStop = false;
            this.ObjectList.Text = "Object List";
            //
            // ObjectListView
            //
            this.ObjectListView.Columns.AddRange(new System.Windows.Forms.ColumnHeader[] {
            this.columnhead1,
            this.columnHeader2,
            this.columnHeader1,
            this.columnHeader3,
            this.columnHeader4,
            this.columnHeader5});
            this.ObjectListView.Dock = System.Windows.Forms.DockStyle.Fill;
            this.ObjectListView.FullRowSelect = true;
            this.ObjectListView.GridLines = true;
            this.ObjectListView.Location = new System.Drawing.Point(2, 15);
            this.ObjectListView.Margin = new System.Windows.Forms.Padding(2);
            this.ObjectListView.Name = "ObjectListView";
            this.ObjectListView.Size = new System.Drawing.Size(420, 370);
            this.ObjectListView.SmallImageList = this.imageList1;
            this.ObjectListView.TabIndex = 11;
            this.ObjectListView.TileSize = new System.Drawing.Size(400, 100);
            this.ObjectListView.UseCompatibleStateImageBehavior = false;
            this.ObjectListView.View = System.Windows.Forms.View.Details;
            //
            // columnhead1
            //
            this.columnhead1.Text = "Thumbnail";
            this.columnhead1.Width = 100;
            //
            // columnHeader2
            //
            this.columnHeader2.Text = "Name";
            this.columnHeader2.Width = 100;
            //
            // columnHeader1
            //
            this.columnHeader1.Text = "Stream";
            this.columnHeader1.Width = 90;
            //
            // columnHeader3
            //
            this.columnHeader3.Text = "X";
            this.columnHeader3.Width = 20;
            //
            // columnHeader4
            //
            this.columnHeader4.Text = "Y";
            this.columnHeader4.Width = 20;
            //
            // columnHeader5
            //
            this.columnHeader5.Text = "Last Seen";
            this.columnHeader5.Width = 80;
            //
            // RightMenu
            //
            this.RightMenu.Controls.Add(this.StreamListMenu);
            this.RightMenu.Controls.Add(this.ObjectList);
```

```
301             this.RightMenu.Dock = System.Windows.Forms.DockStyle.Right;
302             this.RightMenu.Location = new System.Drawing.Point(998, 24);
303             this.RightMenu.Margin = new System.Windows.Forms.Padding(2);
304             this.RightMenu.Name = "RightMenu";
305             this.RightMenu.Padding = new System.Windows.Forms.Padding(2);
306             this.RightMenu.Size = new System.Drawing.Size(428, 771);
307             this.RightMenu.TabIndex = 14;
308             this.RightMenu.TabStop = false;
309             this.RightMenu.Text = "Side Menu";
310             //
311             // pictureBox1
312             //
313             this.pictureBox1.BackgroundImage = global::Senior_Design_UI.Properties.
                    Resources.scuseal;
314             this.pictureBox1.BackgroundImageLayout = System.Windows.Forms.ImageLayout.
                    Stretch;
315             this.pictureBox1.Dock = System.Windows.Forms.DockStyle.Fill;
316             this.pictureBox1.Location = new System.Drawing.Point(2, 15);
317             this.pictureBox1.Margin = new System.Windows.Forms.Padding(2);
318             this.pictureBox1.Name = "pictureBox1";
319             this.pictureBox1.Size = new System.Drawing.Size(351, 178);
320             this.pictureBox1.TabIndex = 15;
321             this.pictureBox1.TabStop = false;
322             //
323             // Identity
324             //
325             this.Identity.Anchor = ((System.Windows.Forms.AnchorStyles)(((System.Windows.
                    Forms.AnchorStyles.Bottom | System.Windows.Forms.AnchorStyles.Left)
326             | System.Windows.Forms.AnchorStyles.Right)));
327             this.Identity.Controls.Add(this.label1);
328             this.Identity.Controls.Add(this.pictureBox1);
329             this.Identity.Location = new System.Drawing.Point(734, 596);
330             this.Identity.Margin = new System.Windows.Forms.Padding(2);
331             this.Identity.Name = "Identity";
332             this.Identity.Padding = new System.Windows.Forms.Padding(2);
333             this.Identity.Size = new System.Drawing.Size(355, 195);
334             this.Identity.TabIndex = 16;
335             this.Identity.TabStop = false;
336             this.Identity.Text = "Placeholder";
337             //
338             // Form1
339             //
340             this.AutoScaleDimensions = new System.Drawing.SizeF(6F, 13F);
341             this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;
342             this.BackColor = System.Drawing.Color.MintCream;
343             this.ClientSize = new System.Drawing.Size(1426, 795);
344             this.Controls.Add(this.Identity);
345             this.Controls.Add(this.RightMenu);
346             this.Controls.Add(this.VideoStream);
347             this.Controls.Add(this.NoteLog);
348             this.Controls.Add(this.menuStrip1);
349             this.MainMenuStrip = this.menuStrip1;
350             this.Margin = new System.Windows.Forms.Padding(2);
351             this.MaximizeBox = false;
352             this.MinimizeBox = false;
353             this.Name = "Form1";
354             this.Text = "Senior Deisgn UI";
355             this.WindowState = System.Windows.Forms.FormWindowState.Maximized;
356             this.FormClosing += new System.Windows.Forms.FormClosingEventHandler(this.
                    Form1_FormClosing);
357             this.Load += new System.EventHandler(this.Form1_Load);
358             this.ResizeBegin += new System.EventHandler(this.Form1_ResizeBegin);
359             this.ResizeEnd += new System.EventHandler(this.Form1_ResizeBegin);
360             this.Resize += new System.EventHandler(this.Form1_ResizeBegin);
361             this.NoteLog.ResumeLayout(false);
362             this.NoteLog.PerformLayout();
363             ((System.ComponentModel.ISupportInitialize)(this.VideoStream)).EndInit();
364             this.menuStrip1.ResumeLayout(false);
```

```
            this.menuStrip1.PerformLayout();
            this.StreamListMenu.ResumeLayout(false);
            this.ObjectList.ResumeLayout(false);
            this.RightMenu.ResumeLayout(false);
            ((System.ComponentModel.ISupportInitialize)(this.pictureBox1)).EndInit();
            this.Identity.ResumeLayout(false);
            this.Identity.PerformLayout();
            this.ResumeLayout(false);
            this.PerformLayout();

        }

        #endregion

        private System.Windows.Forms.GroupBox NoteLog;
        private System.Windows.Forms.RichTextBox NotificationLog;
        private System.Windows.Forms.TextBox textEnterLine;
        private System.Windows.Forms.Label label1;
        private AxAXVLC.AxVLCPlugin2 VideoStream;
        private System.Windows.Forms.MenuStrip menuStrip1;
        private System.Windows.Forms.ToolStripMenuItem fileToolStripMenuItem;
        private System.Windows.Forms.ToolStripMenuItem streamsToolStripMenuItem;
        private System.Windows.Forms.ToolStripMenuItem addStreamToolStripMenuItem;
        private System.Windows.Forms.ImageList imageList1;
        private System.Windows.Forms.ListView StreamListView;
        private System.Windows.Forms.GroupBox StreamListMenu;
        private System.Windows.Forms.GroupBox ObjectList;
        private System.Windows.Forms.ListView ObjectListView;
        private System.Windows.Forms.ColumnHeader columnhead1;
        private System.Windows.Forms.ColumnHeader columnHeader2;
        private System.Windows.Forms.ColumnHeader columnHeader1;
        private System.Windows.Forms.ColumnHeader columnHeader3;
        private System.Windows.Forms.ColumnHeader columnHeader4;
        private System.Windows.Forms.ColumnHeader columnHeader5;
        private System.Windows.Forms.ColumnHeader columnHeader6;
        private System.Windows.Forms.ColumnHeader columnHeader7;
        private System.Windows.Forms.ColumnHeader columnHeader8;
        private System.Windows.Forms.GroupBox RightMenu;
        private System.Windows.Forms.PictureBox pictureBox1;
        private System.Windows.Forms.GroupBox Identity;
        private System.Windows.Forms.ToolStripMenuItem refreshIconsToolStripMenuItem;
    }
}
```

Listing 16.29: Code for the Designer of Primary Form of the User Interface

16.10 User Interface - Add Stream Interface

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Net;

namespace Senior_Design_UI
{
    public partial class AddStream : Form
    {

        public AddStream()
```

```
18          {
19              InitializeComponent();
20          }
21
22          private void StreamAddCancel(object sender, EventArgs e)
23          {
24              this.Close();
25          }
26
27          private void StreamAddAccept(object sender, EventArgs e)
28          {
29
30
31              if (ThumbnailText.Text != null || ThumbnailText.Text != "")
32              {
33                  Image NewThumb = Image.FromFile(ThumbnailText.Text);
34                  Globals.StreamThumbsList.Images.Add(NewThumb);
35              }
36              //else
37                  //Globals.StreamThumbsList.Images.Add();
38
39              RSTPStream NewStream = new RSTPStream(urlTextBox.Text, detTextBox.Text,
                    locTextBox.Text, Globals.StreamList.Count + 1);
40              Globals.StreamList.Add(NewStream);
41
42
43              /* Thumbnail WEB Request Code ***NOT WORKING***
44              Uri ThumbUrl = new Uri(NewStream.get_url() + "/dsm?");
45              WebRequest ThumbNailRequest = WebRequest.Create("http://192.168.11.32/dsm?");
46              WebResponse ThumbNailReponse = ThumbNailRequest.GetResponse();
47
48              if (ThumbNailReponse.ResponseUri.AbsoluteUri != null)
49              {
50                  Image NewThumb = Image.FromStream(ThumbNailReponse.GetResponseStream());
51                  Globals.SteamThumbsList.Images.Add(NewThumb);
52              }*/
53
54              this.Close();
55          }
56
57          private void button1_Click(object sender, EventArgs e)
58          {
59              //ez creata invis player
60              //if encountered error
61              //then change text
62
63
64              labelTestResults.Text = "Test Reuslts : Not Implemented";
65
66          }
67
68          private void button3_Click(object sender, EventArgs e)
69          {
70              OpenFileDialog ImgLoc = new OpenFileDialog();
71              if (ImgLoc.ShowDialog() == DialogResult.OK)
72              {
73                  ThumbnailText.Text = ImgLoc.FileName;
74              }
75          }
76      }
77  }
```

Listing 16.30: Code for the Add Stream Interface

16.11 User Interface - Add Stream Interface Designer File

```
namespace Senior_Design_UI
{
    partial class AddStream
    {
        /// <summary>
        /// Required designer variable.
        /// </summary>
        private System.ComponentModel.IContainer components = null;

        /// <summary>
        /// Clean up any resources being used.
        /// </summary>
        /// <param name="disposing">true if managed resources should be disposed;
            otherwise, false.</param>
        protected override void Dispose(bool disposing)
        {
            if (disposing && (components != null))
            {
                components.Dispose();
            }
            base.Dispose(disposing);
        }

        #region Windows Form Designer generated code

        /// <summary>
        /// Required method for Designer support - do not modify
        /// the contents of this method with the code editor.
        /// </summary>
        private void InitializeComponent()
        {
            this.button1 = new System.Windows.Forms.Button();
            this.label1 = new System.Windows.Forms.Label();
            this.label2 = new System.Windows.Forms.Label();
            this.urlTextBox = new System.Windows.Forms.TextBox();
            this.locTextBox = new System.Windows.Forms.TextBox();
            this.detTextBox = new System.Windows.Forms.TextBox();
            this.labelTestResults = new System.Windows.Forms.Label();
            this.label4 = new System.Windows.Forms.Label();
            this.button2 = new System.Windows.Forms.Button();
            this.cancelButton = new System.Windows.Forms.Button();
            this.label5 = new System.Windows.Forms.Label();
            this.nameTextBox = new System.Windows.Forms.TextBox();
            this.button3 = new System.Windows.Forms.Button();
            this.ThumbnailText = new System.Windows.Forms.TextBox();
            this.SuspendLayout();
            //
            // button1
            //
            this.button1.Location = new System.Drawing.Point(11, 139);
            this.button1.Margin = new System.Windows.Forms.Padding(2);
            this.button1.Name = "button1";
            this.button1.Size = new System.Drawing.Size(74, 19);
            this.button1.TabIndex = 0;
            this.button1.Text = "Test Stream";
            this.button1.UseVisualStyleBackColor = true;
            this.button1.Visible = false;
            this.button1.Click += new System.EventHandler(this.button1_Click);
            //
            // label1
            //
            this.label1.AutoSize = true;
            this.label1.Location = new System.Drawing.Point(8, 37);
            this.label1.Margin = new System.Windows.Forms.Padding(2, 0, 2, 0);
            this.label1.Name = "label1";
            this.label1.Size = new System.Drawing.Size(65, 13);
            this.label1.TabIndex = 1;
```

```
            this.label1.Text = "Camera IP : ";
            //
            // label2
            //
            this.label2.AutoSize = true;
            this.label2.Location = new System.Drawing.Point(8, 63);
            this.label2.Margin = new System.Windows.Forms.Padding(2, 0, 2, 0);
            this.label2.Name = "label2";
            this.label2.Size = new System.Drawing.Size(57, 13);
            this.label2.TabIndex = 2;
            this.label2.Text = "Location : ";
            //
            // urlTextBox
            //
            this.urlTextBox.Location = new System.Drawing.Point(77, 34);
            this.urlTextBox.Margin = new System.Windows.Forms.Padding(2);
            this.urlTextBox.Name = "urlTextBox";
            this.urlTextBox.Size = new System.Drawing.Size(162, 20);
            this.urlTextBox.TabIndex = 3;
            this.urlTextBox.Text = "192.168.11.32";
            //
            // locTextBox
            //
            this.locTextBox.Location = new System.Drawing.Point(69, 60);
            this.locTextBox.Margin = new System.Windows.Forms.Padding(2);
            this.locTextBox.Name = "locTextBox";
            this.locTextBox.Size = new System.Drawing.Size(170, 20);
            this.locTextBox.TabIndex = 4;
            this.locTextBox.Text = "608C";
            //
            // detTextBox
            //
            this.detTextBox.Location = new System.Drawing.Point(60, 85);
            this.detTextBox.Margin = new System.Windows.Forms.Padding(2);
            this.detTextBox.Multiline = true;
            this.detTextBox.Name = "detTextBox";
            this.detTextBox.Size = new System.Drawing.Size(179, 49);
            this.detTextBox.TabIndex = 5;
            this.detTextBox.Text = "Bannon Engineering Outside facing benson quad ";
            //
            // labelTestResults
            //
            this.labelTestResults.AutoSize = true;
            this.labelTestResults.Location = new System.Drawing.Point(91, 142);
            this.labelTestResults.Margin = new System.Windows.Forms.Padding(2, 0, 2, 0);
            this.labelTestResults.Name = "labelTestResults";
            this.labelTestResults.Size = new System.Drawing.Size(118, 13);
            this.labelTestResults.TabIndex = 6;
            this.labelTestResults.Text = "Test Results : Untested";
            this.labelTestResults.Visible = false;
            //
            // label4
            //
            this.label4.AutoSize = true;
            this.label4.Location = new System.Drawing.Point(8, 88);
            this.label4.Margin = new System.Windows.Forms.Padding(2, 0, 2, 0);
            this.label4.Name = "label4";
            this.label4.Size = new System.Drawing.Size(48, 13);
            this.label4.TabIndex = 7;
            this.label4.Text = "Details : ";
            //
            // button2
            //
            this.button2.DialogResult = System.Windows.Forms.DialogResult.OK;
            this.button2.Location = new System.Drawing.Point(34, 211);
            this.button2.Margin = new System.Windows.Forms.Padding(2);
            this.button2.Name = "button2";
            this.button2.Size = new System.Drawing.Size(74, 19);
```

```
135             this.button2.TabIndex = 8;
136             this.button2.Text = "Add Stream";
137             this.button2.UseVisualStyleBackColor = true;
138             this.button2.Click += new System.EventHandler(this.StreamAddAccept);
139             //
140             // cancelButton
141             //
142             this.cancelButton.DialogResult = System.Windows.Forms.DialogResult.Cancel;
143             this.cancelButton.Location = new System.Drawing.Point(136, 211);
144             this.cancelButton.Margin = new System.Windows.Forms.Padding(2);
145             this.cancelButton.Name = "cancelButton";
146             this.cancelButton.Size = new System.Drawing.Size(65, 19);
147             this.cancelButton.TabIndex = 9;
148             this.cancelButton.Text = "Cancel";
149             this.cancelButton.UseVisualStyleBackColor = true;
150             this.cancelButton.Click += new System.EventHandler(this.StreamAddCancel);
151             //
152             // label5
153             //
154             this.label5.AutoSize = true;
155             this.label5.Location = new System.Drawing.Point(8, 12);
156             this.label5.Margin = new System.Windows.Forms.Padding(2, 0, 2, 0);
157             this.label5.Name = "label5";
158             this.label5.Size = new System.Drawing.Size(80, 13);
159             this.label5.TabIndex = 10;
160             this.label5.Text = "Stream Name : ";
161             //
162             // nameTextBox
163             //
164             this.nameTextBox.Location = new System.Drawing.Point(94, 9);
165             this.nameTextBox.Margin = new System.Windows.Forms.Padding(2);
166             this.nameTextBox.Name = "nameTextBox";
167             this.nameTextBox.Size = new System.Drawing.Size(145, 20);
168             this.nameTextBox.TabIndex = 11;
169             this.nameTextBox.Text = "Home";
170             //
171             // button3
172             //
173             this.button3.Location = new System.Drawing.Point(11, 163);
174             this.button3.Name = "button3";
175             this.button3.Size = new System.Drawing.Size(228, 19);
176             this.button3.TabIndex = 12;
177             this.button3.Text = "Choose Thumbnail";
178             this.button3.UseVisualStyleBackColor = true;
179             this.button3.Click += new System.EventHandler(this.button3_Click);
180             //
181             // ThumbnailText
182             //
183             this.ThumbnailText.Location = new System.Drawing.Point(11, 187);
184             this.ThumbnailText.Name = "ThumbnailText";
185             this.ThumbnailText.ReadOnly = true;
186             this.ThumbnailText.Size = new System.Drawing.Size(228, 20);
187             this.ThumbnailText.TabIndex = 13;
188             //
189             // AddStream
190             //
191             this.AutoScaleDimensions = new System.Drawing.SizeF(6F, 13F);
192             this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;
193             this.CancelButton = this.cancelButton;
194             this.ClientSize = new System.Drawing.Size(247, 234);
195             this.Controls.Add(this.ThumbnailText);
196             this.Controls.Add(this.button3);
197             this.Controls.Add(this.nameTextBox);
198             this.Controls.Add(this.label5);
199             this.Controls.Add(this.cancelButton);
200             this.Controls.Add(this.button2);
201             this.Controls.Add(this.label4);
202             this.Controls.Add(this.labelTestResults);
```

```
203             this.Controls.Add(this.detTextBox);
204             this.Controls.Add(this.locTextBox);
205             this.Controls.Add(this.urlTextBox);
206             this.Controls.Add(this.label2);
207             this.Controls.Add(this.label1);
208             this.Controls.Add(this.button1);
209             this.Margin = new System.Windows.Forms.Padding(2);
210             this.MaximizeBox = false;
211             this.MinimizeBox = false;
212             this.Name = "AddStream";
213             this.StartPosition = System.Windows.Forms.FormStartPosition.CenterScreen;
214             this.Text = "AddStream";
215             this.ResumeLayout(false);
216             this.PerformLayout();
217
218         }
219
220         #endregion
221
222         private System.Windows.Forms.Button button1;
223         private System.Windows.Forms.Label label1;
224         private System.Windows.Forms.Label label2;
225         private System.Windows.Forms.TextBox urlTextBox;
226         private System.Windows.Forms.TextBox locTextBox;
227         private System.Windows.Forms.TextBox detTextBox;
228         private System.Windows.Forms.Label labelTestResults;
229         private System.Windows.Forms.Label label4;
230         private System.Windows.Forms.Button button2;
231         private System.Windows.Forms.Button cancelButton;
232         private System.Windows.Forms.Label label5;
233         private System.Windows.Forms.TextBox nameTextBox;
234         private System.Windows.Forms.Button button3;
235         private System.Windows.Forms.TextBox ThumbnailText;
236     }
237 }
```

Listing 16.31: Code for the Designer of Add Stream Interface

16.12 User Interface - Event Object Class

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Text;
5  using System.Threading.Tasks;
6
7  namespace Senior_Design_UI
8  {
9      class EventObject
10     {
11         public EventObject()
12         {
13             individual = "";
14             imgindex = 0;
15             camera = 0;
16             Xpos = 0;
17             Ypos = 0;
18         }
19
20         public EventObject( String individ, Int32 cameraid, Int32 img, Int32 eventID,
               Int32 x, Int32 y )
21         {
22             individual = individ;
23             camera = cameraid;
24             imgindex = img;
```

```
            eventid = eventID;
            Xpos = x;
            Ypos = y;
        }

        public String Individ
        {
            get { return individual; }
            set { individual = value;}
        }

        public Int32 ImgIndex
        {
            get { return imgindex; }
            set { imgindex = value; }
        }

        public Int32 CameraID
        {
            get { return camera; }
            set { camera = value; }
        }

        public Int32 EventID
        {
            get { return eventid; }
            set { eventid = value; }
        }

        public Int32 X
        {
            get { return Xpos; }
            set { Xpos = value; }
        }

        public Int32 Y
        {
            get { return Ypos; }
            set { Ypos = value; }
        }

        // Variables
        String individual;
        Int32 imgindex, eventid, camera, Xpos, Ypos;
    }
}
```

Listing 16.32: Code for the Event Object Class

16.13 User Interface - RSTPStream Class

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Senior_Design_UI
{
    class RSTPStream
    {
        public RSTPStream()
        {
            url = "";
            details = "";
```

```
            loc = "";
            imgIndex = 0;
            //options = ""; //VLCplugin issues with working
        }

        public RSTPStream( String urls, String dis, String locs, Int32 name )
        {
            url = urls;
            details = dis;
            loc = locs;
            imgIndex = name;
            //options = ":showdisplay=false"; //Depricated
        }

        public string get_url()
        {
            return url;
        }

        public string get_details()
        {
            return details;
        }

        public string get_loc()
        {
            return loc;
        }

        public Int32 get_imgIndex()
        {
            return imgIndex;
        }

        public void set_url(String urls)
        {
            url = urls;
        }

        public void set_details(String dets)
        {
            details = dets;
        }

        public void set_loc(String locs)
        {
            loc = locs;
        }

        public void set_imgIndex(Int32 img)
        {
            imgIndex = img;
        }

        // Variables
        String url, name, details, loc;
        Int32 imgIndex;
    }
}
```

Listing 16.33: Code for the RSTPStream Class

16.14 User Interface - Database Configuration File

```
#pragma warning disable 1591
```

```
//------------------------------------------------------------------------------
// <auto-generated>
//     This code was generated by a tool.
//     Runtime Version:4.0.30319.34014
//
//     Changes to this file may cause incorrect behavior and will be lost if
//     the code is regenerated.
// </auto-generated>
//------------------------------------------------------------------------------

namespace Senior_Design_UI
{
  using System.Data.Linq;
  using System.Data.Linq.Mapping;
  using System.Data;
  using System.Collections.Generic;
  using System.Reflection;
  using System.Linq;
  using System.Linq.Expressions;
  using System.ComponentModel;
  using System;


  [global::System.Data.Linq.Mapping.DatabaseAttribute(Name="SeniorDesignDB2")]
  public partial class SDDBDataContext : System.Data.Linq.DataContext
  {

    private static System.Data.Linq.Mapping.MappingSource mappingSource = new
        AttributeMappingSource();

    #region Extensibility Method Definitions
    partial void OnCreated();
    partial void InsertCamera(Camera instance);
    partial void UpdateCamera(Camera instance);
    partial void DeleteCamera(Camera instance);
    partial void InsertCameraStat(CameraStat instance);
    partial void UpdateCameraStat(CameraStat instance);
    partial void DeleteCameraStat(CameraStat instance);
    partial void InsertDetectionEvent(DetectionEvent instance);
    partial void UpdateDetectionEvent(DetectionEvent instance);
    partial void DeleteDetectionEvent(DetectionEvent instance);
    partial void InsertIndividual(Individual instance);
    partial void UpdateIndividual(Individual instance);
    partial void DeleteIndividual(Individual instance);
    partial void InsertKnownImg(KnownImg instance);
    partial void UpdateKnownImg(KnownImg instance);
    partial void DeleteKnownImg(KnownImg instance);
    partial void InsertAnomaly(Anomaly instance);
    partial void UpdateAnomaly(Anomaly instance);
    partial void DeleteAnomaly(Anomaly instance);
    #endregion

    public SDDBDataContext() :
        base(global::Senior_Design_UI.Properties.Settings.Default.
            SeniorDesignDB2ConnectionString, mappingSource)
    {
      OnCreated();
    }

    public SDDBDataContext(string connection) :
        base(connection, mappingSource)
    {
      OnCreated();
    }

    public SDDBDataContext(System.Data.IDbConnection connection) :
        base(connection, mappingSource)
    {
```

```
68          OnCreated();
69      }
70
71      public SDDBDataContext(string connection, System.Data.Linq.Mapping.MappingSource
            mappingSource) :
72          base(connection, mappingSource)
73      {
74          OnCreated();
75      }
76
77      public SDDBDataContext(System.Data.IDbConnection connection, System.Data.Linq.Mapping.
            MappingSource mappingSource) :
78          base(connection, mappingSource)
79      {
80          OnCreated();
81      }
82
83      public System.Data.Linq.Table<Camera> Cameras
84      {
85          get
86          {
87              return this.GetTable<Camera>();
88          }
89      }
90
91      public System.Data.Linq.Table<CameraStat> CameraStats
92      {
93          get
94          {
95              return this.GetTable<CameraStat>();
96          }
97      }
98
99      public System.Data.Linq.Table<DetectionEvent> DetectionEvents
100     {
101         get
102         {
103             return this.GetTable<DetectionEvent>();
104         }
105     }
106
107     public System.Data.Linq.Table<Individual> Individuals
108     {
109         get
110         {
111             return this.GetTable<Individual>();
112         }
113     }
114
115     public System.Data.Linq.Table<KnownImg> KnownImgs
116     {
117         get
118         {
119             return this.GetTable<KnownImg>();
120         }
121     }
122
123     public System.Data.Linq.Table<Anomaly> Anomalies
124     {
125         get
126         {
127             return this.GetTable<Anomaly>();
128         }
129     }
130 }
131
132 [global::System.Data.Linq.Mapping.TableAttribute(Name="dbo.Camera")]
133 public partial class Camera : INotifyPropertyChanging, INotifyPropertyChanged
```

```
134 {
135
136     private static PropertyChangingEventArgs emptyChangingEventArgs = new
            PropertyChangingEventArgs(String.Empty);
137
138     private short _CameraID;
139
140     private string _CameraName;
141
142     private string _CameraIP;
143
144     private string _FPGAIP;
145
146     private string _CameraLoc;
147
148     private string _CameraDetails;
149
150     private System.Nullable<short> _imgIndex;
151
152     private string _ThumbPath;
153
154     private EntitySet<CameraStat> _CameraStats;
155
156     private EntitySet<DetectionEvent> _DetectionEvents;
157
158     #region Extensibility Method Definitions
159     partial void OnLoaded();
160     partial void OnValidate(System.Data.Linq.ChangeAction action);
161     partial void OnCreated();
162     partial void OnCameraIDChanging(short value);
163     partial void OnCameraIDChanged();
164     partial void OnCameraNameChanging(string value);
165     partial void OnCameraNameChanged();
166     partial void OnCameraIPChanging(string value);
167     partial void OnCameraIPChanged();
168     partial void OnFPGAIPChanging(string value);
169     partial void OnFPGAIPChanged();
170     partial void OnCameraLocChanging(string value);
171     partial void OnCameraLocChanged();
172     partial void OnCameraDetailsChanging(string value);
173     partial void OnCameraDetailsChanged();
174     partial void OnimgIndexChanging(System.Nullable<short> value);
175     partial void OnimgIndexChanged();
176     partial void OnThumbPathChanging(string value);
177     partial void OnThumbPathChanged();
178     #endregion
179
180     public Camera()
181     {
182       this._CameraStats = new EntitySet<CameraStat>(new Action<CameraStat>(this.
              attach_CameraStats), new Action<CameraStat>(this.detach_CameraStats));
183       this._DetectionEvents = new EntitySet<DetectionEvent>(new Action<DetectionEvent>(
              this.attach_DetectionEvents), new Action<DetectionEvent>(this.
              detach_DetectionEvents));
184       OnCreated();
185     }
186
187     [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_CameraID", DbType="
            SmallInt NOT NULL", IsPrimaryKey=true)]
188     public short CameraID
189     {
190       get
191       {
192         return this._CameraID;
193       }
194       set
195       {
196         if ((this._CameraID != value))
```

```
197             {
198               this.OnCameraIDChanging(value);
199               this.SendPropertyChanging();
200               this._CameraID = value;
201               this.SendPropertyChanged("CameraID");
202               this.OnCameraIDChanged();
203             }
204           }
205         }
206
207         [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_CameraName", DbType="
              VarChar(20) NOT NULL", CanBeNull=false)]
208         public string CameraName
209         {
210           get
211           {
212             return this._CameraName;
213           }
214           set
215           {
216             if ((this._CameraName != value))
217             {
218               this.OnCameraNameChanging(value);
219               this.SendPropertyChanging();
220               this._CameraName = value;
221               this.SendPropertyChanged("CameraName");
222               this.OnCameraNameChanged();
223             }
224           }
225         }
226
227         [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_CameraIP", DbType="VarChar
              (40)")]
228         public string CameraIP
229         {
230           get
231           {
232             return this._CameraIP;
233           }
234           set
235           {
236             if ((this._CameraIP != value))
237             {
238               this.OnCameraIPChanging(value);
239               this.SendPropertyChanging();
240               this._CameraIP = value;
241               this.SendPropertyChanged("CameraIP");
242               this.OnCameraIPChanged();
243             }
244           }
245         }
246
247         [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_FPGAIP", DbType="VarChar
              (40)")]
248         public string FPGAIP
249         {
250           get
251           {
252             return this._FPGAIP;
253           }
254           set
255           {
256             if ((this._FPGAIP != value))
257             {
258               this.OnFPGAIPChanging(value);
259               this.SendPropertyChanging();
260               this._FPGAIP = value;
261               this.SendPropertyChanged("FPGAIP");
```

```
262             this.OnFPGAIPChanged();
263         }
264       }
265     }
266
267     [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_CameraLoc", DbType="
            VarChar(40)")]
268     public string CameraLoc
269     {
270       get
271       {
272         return this._CameraLoc;
273       }
274       set
275       {
276         if ((this._CameraLoc != value))
277         {
278           this.OnCameraLocChanging(value);
279           this.SendPropertyChanging();
280           this._CameraLoc = value;
281           this.SendPropertyChanged("CameraLoc");
282           this.OnCameraLocChanged();
283         }
284       }
285     }
286
287     [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_CameraDetails", DbType="
            VarChar(100)")]
288     public string CameraDetails
289     {
290       get
291       {
292         return this._CameraDetails;
293       }
294       set
295       {
296         if ((this._CameraDetails != value))
297         {
298           this.OnCameraDetailsChanging(value);
299           this.SendPropertyChanging();
300           this._CameraDetails = value;
301           this.SendPropertyChanged("CameraDetails");
302           this.OnCameraDetailsChanged();
303         }
304       }
305     }
306
307     [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_imgIndex", DbType="
            SmallInt")]
308     public System.Nullable<short> imgIndex
309     {
310       get
311       {
312         return this._imgIndex;
313       }
314       set
315       {
316         if ((this._imgIndex != value))
317         {
318           this.OnimgIndexChanging(value);
319           this.SendPropertyChanging();
320           this._imgIndex = value;
321           this.SendPropertyChanged("imgIndex");
322           this.OnimgIndexChanged();
323         }
324       }
325     }
326
```

```
327     [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_ThumbPath", DbType="
            VarChar(10)")]
328     public string ThumbPath
329     {
330       get
331       {
332         return this._ThumbPath;
333       }
334       set
335       {
336         if ((this._ThumbPath != value))
337         {
338           this.OnThumbPathChanging(value);
339           this.SendPropertyChanging();
340           this._ThumbPath = value;
341           this.SendPropertyChanged("ThumbPath");
342           this.OnThumbPathChanged();
343         }
344       }
345     }
346
347     [global::System.Data.Linq.Mapping.AssociationAttribute(Name="Camera_CameraStat",
            Storage="_CameraStats", ThisKey="CameraID", OtherKey="CameraID")]
348     public EntitySet<CameraStat> CameraStats
349     {
350       get
351       {
352         return this._CameraStats;
353       }
354       set
355       {
356         this._CameraStats.Assign(value);
357       }
358     }
359
360     [global::System.Data.Linq.Mapping.AssociationAttribute(Name="Camera_DetectionEvent",
            Storage="_DetectionEvents", ThisKey="CameraID", OtherKey="EventCamera")]
361     public EntitySet<DetectionEvent> DetectionEvents
362     {
363       get
364       {
365         return this._DetectionEvents;
366       }
367       set
368       {
369         this._DetectionEvents.Assign(value);
370       }
371     }
372
373     public event PropertyChangingEventHandler PropertyChanging;
374
375     public event PropertyChangedEventHandler PropertyChanged;
376
377     protected virtual void SendPropertyChanging()
378     {
379       if ((this.PropertyChanging != null))
380       {
381         this.PropertyChanging(this, emptyChangingEventArgs);
382       }
383     }
384
385     protected virtual void SendPropertyChanged(String propertyName)
386     {
387       if ((this.PropertyChanged != null))
388       {
389         this.PropertyChanged(this, new PropertyChangedEventArgs(propertyName));
390       }
391     }
```

```
392
393         private void attach_CameraStats(CameraStat entity)
394         {
395             this.SendPropertyChanging();
396             entity.Camera = this;
397         }
398
399         private void detach_CameraStats(CameraStat entity)
400         {
401             this.SendPropertyChanging();
402             entity.Camera = null;
403         }
404
405         private void attach_DetectionEvents(DetectionEvent entity)
406         {
407             this.SendPropertyChanging();
408             entity.Camera = this;
409         }
410
411         private void detach_DetectionEvents(DetectionEvent entity)
412         {
413             this.SendPropertyChanging();
414             entity.Camera = null;
415         }
416     }
417
418     [global::System.Data.Linq.Mapping.TableAttribute(Name="dbo.CameraStats")]
419     public partial class CameraStat : INotifyPropertyChanging, INotifyPropertyChanged
420     {
421
422         private static PropertyChangingEventArgs emptyChangingEventArgs = new
                PropertyChangingEventArgs(String.Empty);
423
424         private short _TimeID;
425
426         private short _CameraID;
427
428         private System.Nullable<System.TimeSpan> _LoTimeRange;
429
430         private System.Nullable<System.TimeSpan> _HiTimeRange;
431
432         private System.Nullable<double> _Simga;
433
434         private System.Nullable<double> _Mean;
435
436         private System.Nullable<int> _NValue;
437
438         private EntityRef<Camera> _Camera;
439
440         #region Extensibility Method Definitions
441         partial void OnLoaded();
442         partial void OnValidate(System.Data.Linq.ChangeAction action);
443         partial void OnCreated();
444         partial void OnTimeIDChanging(short value);
445         partial void OnTimeIDChanged();
446         partial void OnCameraIDChanging(short value);
447         partial void OnCameraIDChanged();
448         partial void OnLoTimeRangeChanging(System.Nullable<System.TimeSpan> value);
449         partial void OnLoTimeRangeChanged();
450         partial void OnHiTimeRangeChanging(System.Nullable<System.TimeSpan> value);
451         partial void OnHiTimeRangeChanged();
452         partial void OnSimgaChanging(System.Nullable<double> value);
453         partial void OnSimgaChanged();
454         partial void OnMeanChanging(System.Nullable<double> value);
455         partial void OnMeanChanged();
456         partial void OnNValueChanging(System.Nullable<int> value);
457         partial void OnNValueChanged();
458         #endregion
```

```
459
460     public CameraStat()
461     {
462       this._Camera = default(EntityRef<Camera>);
463       OnCreated();
464     }
465
466     [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_TimeID", DbType="SmallInt
            NOT NULL", IsPrimaryKey=true)]
467     public short TimeID
468     {
469       get
470       {
471         return this._TimeID;
472       }
473       set
474       {
475         if ((this._TimeID != value))
476         {
477           this.OnTimeIDChanging(value);
478           this.SendPropertyChanging();
479           this._TimeID = value;
480           this.SendPropertyChanged("TimeID");
481           this.OnTimeIDChanged();
482         }
483       }
484     }
485
486     [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_CameraID", DbType="
            SmallInt NOT NULL")]
487     public short CameraID
488     {
489       get
490       {
491         return this._CameraID;
492       }
493       set
494       {
495         if ((this._CameraID != value))
496         {
497           if (this._Camera.HasLoadedOrAssignedValue)
498           {
499             throw new System.Data.Linq.ForeignKeyReferenceAlreadyHasValueException();
500           }
501           this.OnCameraIDChanging(value);
502           this.SendPropertyChanging();
503           this._CameraID = value;
504           this.SendPropertyChanged("CameraID");
505           this.OnCameraIDChanged();
506         }
507       }
508     }
509
510     [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_LoTimeRange", DbType="Time
            ")]
511     public System.Nullable<System.TimeSpan> LoTimeRange
512     {
513       get
514       {
515         return this._LoTimeRange;
516       }
517       set
518       {
519         if ((this._LoTimeRange != value))
520         {
521           this.OnLoTimeRangeChanging(value);
522           this.SendPropertyChanging();
523           this._LoTimeRange = value;
```

```
524             this.SendPropertyChanged("LoTimeRange");
525             this.OnLoTimeRangeChanged();
526         }
527       }
528     }
529
530     [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_HiTimeRange", DbType="Time
        ")]
531     public System.Nullable<System.TimeSpan> HiTimeRange
532     {
533       get
534       {
535         return this._HiTimeRange;
536       }
537       set
538       {
539         if ((this._HiTimeRange != value))
540         {
541           this.OnHiTimeRangeChanging(value);
542           this.SendPropertyChanging();
543           this._HiTimeRange = value;
544           this.SendPropertyChanged("HiTimeRange");
545           this.OnHiTimeRangeChanged();
546         }
547       }
548     }
549
550     [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_Simga", DbType="Float")]
551     public System.Nullable<double> Simga
552     {
553       get
554       {
555         return this._Simga;
556       }
557       set
558       {
559         if ((this._Simga != value))
560         {
561           this.OnSimgaChanging(value);
562           this.SendPropertyChanging();
563           this._Simga = value;
564           this.SendPropertyChanged("Simga");
565           this.OnSimgaChanged();
566         }
567       }
568     }
569
570     [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_Mean", DbType="Float")]
571     public System.Nullable<double> Mean
572     {
573       get
574       {
575         return this._Mean;
576       }
577       set
578       {
579         if ((this._Mean != value))
580         {
581           this.OnMeanChanging(value);
582           this.SendPropertyChanging();
583           this._Mean = value;
584           this.SendPropertyChanged("Mean");
585           this.OnMeanChanged();
586         }
587       }
588     }
589
590     [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_NValue", DbType="Int")]
```

```
591     public System.Nullable<int> NValue
592     {
593       get
594       {
595         return this._NValue;
596       }
597       set
598       {
599         if ((this._NValue != value))
600         {
601           this.OnNValueChanging(value);
602           this.SendPropertyChanging();
603           this._NValue = value;
604           this.SendPropertyChanged("NValue");
605           this.OnNValueChanged();
606         }
607       }
608     }
609
610     [global::System.Data.Linq.Mapping.AssociationAttribute(Name="Camera_CameraStat",
          Storage="_Camera", ThisKey="CameraID", OtherKey="CameraID", IsForeignKey=true,
          DeleteOnNull=true, DeleteRule="CASCADE")]
611     public Camera Camera
612     {
613       get
614       {
615         return this._Camera.Entity;
616       }
617       set
618       {
619         Camera previousValue = this._Camera.Entity;
620         if (((previousValue != value)
621               || (this._Camera.HasLoadedOrAssignedValue == false)))
622         {
623           this.SendPropertyChanging();
624           if ((previousValue != null))
625           {
626             this._Camera.Entity = null;
627             previousValue.CameraStats.Remove(this);
628           }
629           this._Camera.Entity = value;
630           if ((value != null))
631           {
632             value.CameraStats.Add(this);
633             this._CameraID = value.CameraID;
634           }
635           else
636           {
637             this._CameraID = default(short);
638           }
639           this.SendPropertyChanged("Camera");
640         }
641       }
642     }
643
644     public event PropertyChangingEventHandler PropertyChanging;
645
646     public event PropertyChangedEventHandler PropertyChanged;
647
648     protected virtual void SendPropertyChanging()
649     {
650       if ((this.PropertyChanging != null))
651       {
652         this.PropertyChanging(this, emptyChangingEventArgs);
653       }
654     }
655
656     protected virtual void SendPropertyChanged(String propertyName)
```

```
657     {
658       if ((this.PropertyChanged != null))
659       {
660         this.PropertyChanged(this, new PropertyChangedEventArgs(propertyName));
661       }
662     }
663   }
664
665   [global::System.Data.Linq.Mapping.TableAttribute(Name="dbo.DetectionEvent")]
666   public partial class DetectionEvent : INotifyPropertyChanging, INotifyPropertyChanged
667   {
668
669     private static PropertyChangingEventArgs emptyChangingEventArgs = new
          PropertyChangingEventArgs(String.Empty);
670
671     private int _EventID;
672
673     private short _EventCamera;
674
675     private System.TimeSpan _EventTimeIn;
676
677     private System.Nullable<System.TimeSpan> _EventTimeOut;
678
679     private System.Nullable<int> _isChecked;
680
681     private System.Nullable<short> _EntryType;
682
683     private System.Nullable<short> _ExitType;
684
685     private int _SuspectID;
686
687     private double _Confidence;
688
689     private EntitySet<Anomaly> _Anomalies;
690
691     private EntityRef<Camera> _Camera;
692
693     private EntityRef<Individual> _Individual;
694
695     #region Extensibility Method Definitions
696     partial void OnLoaded();
697     partial void OnValidate(System.Data.Linq.ChangeAction action);
698     partial void OnCreated();
699     partial void OnEventIDChanging(int value);
700     partial void OnEventIDChanged();
701     partial void OnEventCameraChanging(short value);
702     partial void OnEventCameraChanged();
703     partial void OnEventTimeInChanging(System.TimeSpan value);
704     partial void OnEventTimeInChanged();
705     partial void OnEventTimeOutChanging(System.Nullable<System.TimeSpan> value);
706     partial void OnEventTimeOutChanged();
707     partial void OnisCheckedChanging(System.Nullable<int> value);
708     partial void OnisCheckedChanged();
709     partial void OnEntryTypeChanging(System.Nullable<short> value);
710     partial void OnEntryTypeChanged();
711     partial void OnExitTypeChanging(System.Nullable<short> value);
712     partial void OnExitTypeChanged();
713     partial void OnSuspectIDChanging(int value);
714     partial void OnSuspectIDChanged();
715     partial void OnConfidenceChanging(double value);
716     partial void OnConfidenceChanged();
717     #endregion
718
719     public DetectionEvent()
720     {
721       this._Anomalies = new EntitySet<Anomaly>(new Action<Anomaly>(this.attach_Anomalies),
            new Action<Anomaly>(this.detach_Anomalies));
722       this._Camera = default(EntityRef<Camera>);
```

```
723         this._Individual = default(EntityRef<Individual>);
724         OnCreated();
725       }
726
727       [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_EventID", DbType="Int NOT
              NULL", IsPrimaryKey=true)]
728       public int EventID
729       {
730         get
731         {
732           return this._EventID;
733         }
734         set
735         {
736           if ((this._EventID != value))
737           {
738             this.OnEventIDChanging(value);
739             this.SendPropertyChanging();
740             this._EventID = value;
741             this.SendPropertyChanged("EventID");
742             this.OnEventIDChanged();
743           }
744         }
745       }
746
747       [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_EventCamera", DbType="
              SmallInt NOT NULL")]
748       public short EventCamera
749       {
750         get
751         {
752           return this._EventCamera;
753         }
754         set
755         {
756           if ((this._EventCamera != value))
757           {
758             if (this._Camera.HasLoadedOrAssignedValue)
759             {
760               throw new System.Data.Linq.ForeignKeyReferenceAlreadyHasValueException();
761             }
762             this.OnEventCameraChanging(value);
763             this.SendPropertyChanging();
764             this._EventCamera = value;
765             this.SendPropertyChanged("EventCamera");
766             this.OnEventCameraChanged();
767           }
768         }
769       }
770
771       [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_EventTimeIn", DbType="Time
              NOT NULL")]
772       public System.TimeSpan EventTimeIn
773       {
774         get
775         {
776           return this._EventTimeIn;
777         }
778         set
779         {
780           if ((this._EventTimeIn != value))
781           {
782             this.OnEventTimeInChanging(value);
783             this.SendPropertyChanging();
784             this._EventTimeIn = value;
785             this.SendPropertyChanged("EventTimeIn");
786             this.OnEventTimeInChanged();
787           }
```

```
788         }
789       }
790
791       [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_EventTimeOut", DbType="
              Time")]
792       public System.Nullable<System.TimeSpan> EventTimeOut
793       {
794         get
795         {
796           return this._EventTimeOut;
797         }
798         set
799         {
800           if ((this._EventTimeOut != value))
801           {
802             this.OnEventTimeOutChanging(value);
803             this.SendPropertyChanging();
804             this._EventTimeOut = value;
805             this.SendPropertyChanged("EventTimeOut");
806             this.OnEventTimeOutChanged();
807           }
808         }
809       }
810
811       [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_isChecked", DbType="Int")]
812       public System.Nullable<int> isChecked
813       {
814         get
815         {
816           return this._isChecked;
817         }
818         set
819         {
820           if ((this._isChecked != value))
821           {
822             this.OnisCheckedChanging(value);
823             this.SendPropertyChanging();
824             this._isChecked = value;
825             this.SendPropertyChanged("isChecked");
826             this.OnisCheckedChanged();
827           }
828         }
829       }
830
831       [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_EntryType", DbType="
              SmallInt")]
832       public System.Nullable<short> EntryType
833       {
834         get
835         {
836           return this._EntryType;
837         }
838         set
839         {
840           if ((this._EntryType != value))
841           {
842             this.OnEntryTypeChanging(value);
843             this.SendPropertyChanging();
844             this._EntryType = value;
845             this.SendPropertyChanged("EntryType");
846             this.OnEntryTypeChanged();
847           }
848         }
849       }
850
851       [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_ExitType", DbType="
              SmallInt")]
852       public System.Nullable<short> ExitType
```

```
853     {
854       get
855       {
856         return this._ExitType;
857       }
858       set
859       {
860         if ((this._ExitType != value))
861         {
862           this.OnExitTypeChanging(value);
863           this.SendPropertyChanging();
864           this._ExitType = value;
865           this.SendPropertyChanged("ExitType");
866           this.OnExitTypeChanged();
867         }
868       }
869     }
870
871     [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_SuspectID", DbType="Int
          NOT NULL")]
872     public int SuspectID
873     {
874       get
875       {
876         return this._SuspectID;
877       }
878       set
879       {
880         if ((this._SuspectID != value))
881         {
882           if (this._Individual.HasLoadedOrAssignedValue)
883           {
884             throw new System.Data.Linq.ForeignKeyReferenceAlreadyHasValueException();
885           }
886           this.OnSuspectIDChanging(value);
887           this.SendPropertyChanging();
888           this._SuspectID = value;
889           this.SendPropertyChanged("SuspectID");
890           this.OnSuspectIDChanged();
891         }
892       }
893     }
894
895     [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_Confidence", DbType="Float
          NOT NULL")]
896     public double Confidence
897     {
898       get
899       {
900         return this._Confidence;
901       }
902       set
903       {
904         if ((this._Confidence != value))
905         {
906           this.OnConfidenceChanging(value);
907           this.SendPropertyChanging();
908           this._Confidence = value;
909           this.SendPropertyChanged("Confidence");
910           this.OnConfidenceChanged();
911         }
912       }
913     }
914
915     [global::System.Data.Linq.Mapping.AssociationAttribute(Name="DetectionEvent_Anomaly",
          Storage="_Anomalies", ThisKey="EventID", OtherKey="EventID")]
916     public EntitySet<Anomaly> Anomalies
917     {
```

```
        get
        {
          return this._Anomalies;
        }
        set
        {
          this._Anomalies.Assign(value);
        }
      }

      [global::System.Data.Linq.Mapping.AssociationAttribute(Name="Camera_DetectionEvent",
          Storage="_Camera", ThisKey="EventCamera", OtherKey="CameraID", IsForeignKey=true,
          DeleteOnNull=true, DeleteRule="CASCADE")]
      public Camera Camera
      {
        get
        {
          return this._Camera.Entity;
        }
        set
        {
          Camera previousValue = this._Camera.Entity;
          if (((previousValue != value)
                || (this._Camera.HasLoadedOrAssignedValue == false)))
          {
            this.SendPropertyChanging();
            if ((previousValue != null))
            {
              this._Camera.Entity = null;
              previousValue.DetectionEvents.Remove(this);
            }
            this._Camera.Entity = value;
            if ((value != null))
            {
              value.DetectionEvents.Add(this);
              this._EventCamera = value.CameraID;
            }
            else
            {
              this._EventCamera = default(short);
            }
            this.SendPropertyChanged("Camera");
          }
        }
      }

      [global::System.Data.Linq.Mapping.AssociationAttribute(Name="Individual_DetectionEvent
          ", Storage="_Individual", ThisKey="SuspectID", OtherKey="ID", IsForeignKey=true,
          DeleteOnNull=true, DeleteRule="CASCADE")]
      public Individual Individual
      {
        get
        {
          return this._Individual.Entity;
        }
        set
        {
          Individual previousValue = this._Individual.Entity;
          if (((previousValue != value)
                || (this._Individual.HasLoadedOrAssignedValue == false)))
          {
            this.SendPropertyChanging();
            if ((previousValue != null))
            {
              this._Individual.Entity = null;
              previousValue.DetectionEvents.Remove(this);
            }
            this._Individual.Entity = value;
```

```
982             if ((value != null))
983             {
984               value.DetectionEvents.Add(this);
985               this._SuspectID = value.ID;
986             }
987             else
988             {
989               this._SuspectID = default(int);
990             }
991             this.SendPropertyChanged("Individual");
992           }
993         }
994       }
995
996       public event PropertyChangingEventHandler PropertyChanging;
997
998       public event PropertyChangedEventHandler PropertyChanged;
999
1000      protected virtual void SendPropertyChanging()
1001      {
1002        if ((this.PropertyChanging != null))
1003        {
1004          this.PropertyChanging(this, emptyChangingEventArgs);
1005        }
1006      }
1007
1008      protected virtual void SendPropertyChanged(String propertyName)
1009      {
1010        if ((this.PropertyChanged != null))
1011        {
1012          this.PropertyChanged(this, new PropertyChangedEventArgs(propertyName));
1013        }
1014      }
1015
1016      private void attach_Anomalies(Anomaly entity)
1017      {
1018        this.SendPropertyChanging();
1019        entity.DetectionEvent = this;
1020      }
1021
1022      private void detach_Anomalies(Anomaly entity)
1023      {
1024        this.SendPropertyChanging();
1025        entity.DetectionEvent = null;
1026      }
1027    }
1028
1029    [global::System.Data.Linq.Mapping.TableAttribute(Name="dbo.Individual")]
1030    public partial class Individual : INotifyPropertyChanging, INotifyPropertyChanged
1031    {
1032
1033      private static PropertyChangingEventArgs emptyChangingEventArgs = new
              PropertyChangingEventArgs(String.Empty);
1034
1035      private int _ID;
1036
1037      private string _Name;
1038
1039      private string _Details;
1040
1041      private EntitySet<DetectionEvent> _DetectionEvents;
1042
1043      private EntitySet<KnownImg> _KnownImgs;
1044
1045      #region Extensibility Method Definitions
1046      partial void OnLoaded();
1047      partial void OnValidate(System.Data.Linq.ChangeAction action);
1048      partial void OnCreated();
```

```
1049      partial void OnIDChanging(int value);
1050      partial void OnIDChanged();
1051      partial void OnNameChanging(string value);
1052      partial void OnNameChanged();
1053      partial void OnDetailsChanging(string value);
1054      partial void OnDetailsChanged();
1055      #endregion
1056
1057      public Individual()
1058      {
1059        this._DetectionEvents = new EntitySet<DetectionEvent>(new Action<DetectionEvent>(
              this.attach_DetectionEvents), new Action<DetectionEvent>(this.
              detach_DetectionEvents));
1060        this._KnownImgs = new EntitySet<KnownImg>(new Action<KnownImg>(this.attach_KnownImgs
              ), new Action<KnownImg>(this.detach_KnownImgs));
1061        OnCreated();
1062      }
1063
1064      [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_ID", AutoSync=AutoSync.
            OnInsert, DbType="Int NOT NULL IDENTITY", IsPrimaryKey=true, IsDbGenerated=true)]
1065      public int ID
1066      {
1067        get
1068        {
1069          return this._ID;
1070        }
1071        set
1072        {
1073          if ((this._ID != value))
1074          {
1075            this.OnIDChanging(value);
1076            this.SendPropertyChanging();
1077            this._ID = value;
1078            this.SendPropertyChanged("ID");
1079            this.OnIDChanged();
1080          }
1081        }
1082      }
1083
1084      [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_Name", DbType="VarChar(40)
            NOT NULL", CanBeNull=false)]
1085      public string Name
1086      {
1087        get
1088        {
1089          return this._Name;
1090        }
1091        set
1092        {
1093          if ((this._Name != value))
1094          {
1095            this.OnNameChanging(value);
1096            this.SendPropertyChanging();
1097            this._Name = value;
1098            this.SendPropertyChanged("Name");
1099            this.OnNameChanged();
1100          }
1101        }
1102      }
1103
1104      [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_Details", DbType="VarChar
            (100)")]
1105      public string Details
1106      {
1107        get
1108        {
1109          return this._Details;
1110        }
```

```
1111        set
1112        {
1113          if ((this._Details != value))
1114          {
1115            this.OnDetailsChanging(value);
1116            this.SendPropertyChanging();
1117            this._Details = value;
1118            this.SendPropertyChanged("Details");
1119            this.OnDetailsChanged();
1120          }
1121        }
1122      }
1123
1124      [global::System.Data.Linq.Mapping.AssociationAttribute(Name="Individual_DetectionEvent
          ", Storage="_DetectionEvents", ThisKey="ID", OtherKey="SuspectID")]
1125      public EntitySet<DetectionEvent> DetectionEvents
1126      {
1127        get
1128        {
1129          return this._DetectionEvents;
1130        }
1131        set
1132        {
1133          this._DetectionEvents.Assign(value);
1134        }
1135      }
1136
1137      [global::System.Data.Linq.Mapping.AssociationAttribute(Name="Individual_KnownImg",
          Storage="_KnownImgs", ThisKey="ID", OtherKey="IDImg")]
1138      public EntitySet<KnownImg> KnownImgs
1139      {
1140        get
1141        {
1142          return this._KnownImgs;
1143        }
1144        set
1145        {
1146          this._KnownImgs.Assign(value);
1147        }
1148      }
1149
1150      public event PropertyChangingEventHandler PropertyChanging;
1151
1152      public event PropertyChangedEventHandler PropertyChanged;
1153
1154      protected virtual void SendPropertyChanging()
1155      {
1156        if ((this.PropertyChanging != null))
1157        {
1158          this.PropertyChanging(this, emptyChangingEventArgs);
1159        }
1160      }
1161
1162      protected virtual void SendPropertyChanged(String propertyName)
1163      {
1164        if ((this.PropertyChanged != null))
1165        {
1166          this.PropertyChanged(this, new PropertyChangedEventArgs(propertyName));
1167        }
1168      }
1169
1170      private void attach_DetectionEvents(DetectionEvent entity)
1171      {
1172        this.SendPropertyChanging();
1173        entity.Individual = this;
1174      }
1175
1176      private void detach_DetectionEvents(DetectionEvent entity)
```

```
1177      {
1178        this.SendPropertyChanging();
1179        entity.Individual = null;
1180      }
1181
1182      private void attach_KnownImgs(KnownImg entity)
1183      {
1184        this.SendPropertyChanging();
1185        entity.Individual = this;
1186      }
1187
1188      private void detach_KnownImgs(KnownImg entity)
1189      {
1190        this.SendPropertyChanging();
1191        entity.Individual = null;
1192      }
1193    }
1194
1195    [global::System.Data.Linq.Mapping.TableAttribute(Name="dbo.KnownImgs")]
1196    public partial class KnownImg : INotifyPropertyChanging, INotifyPropertyChanged
1197    {
1198
1199      private static PropertyChangingEventArgs emptyChangingEventArgs = new
              PropertyChangingEventArgs(String.Empty);
1200
1201      private int _KnownImgID;
1202
1203      private int _IDImg;
1204
1205      private System.Data.Linq.Binary _Img;
1206
1207      private EntityRef<Individual> _Individual;
1208
1209      #region Extensibility Method Definitions
1210      partial void OnLoaded();
1211      partial void OnValidate(System.Data.Linq.ChangeAction action);
1212      partial void OnCreated();
1213      partial void OnKnownImgIDChanging(int value);
1214      partial void OnKnownImgIDChanged();
1215      partial void OnIDImgChanging(int value);
1216      partial void OnIDImgChanged();
1217      partial void OnImgChanging(System.Data.Linq.Binary value);
1218      partial void OnImgChanged();
1219      #endregion
1220
1221      public KnownImg()
1222      {
1223        this._Individual = default(EntityRef<Individual>);
1224        OnCreated();
1225      }
1226
1227      [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_KnownImgID", AutoSync=
              AutoSync.OnInsert, DbType="Int NOT NULL IDENTITY", IsPrimaryKey=true,
              IsDbGenerated=true)]
1228      public int KnownImgID
1229      {
1230        get
1231        {
1232          return this._KnownImgID;
1233        }
1234        set
1235        {
1236          if ((this._KnownImgID != value))
1237          {
1238            this.OnKnownImgIDChanging(value);
1239            this.SendPropertyChanging();
1240            this._KnownImgID = value;
1241            this.SendPropertyChanged("KnownImgID");
```

```
1242            this.OnKnownImgIDChanged();
1243        }
1244      }
1245    }
1246
1247    [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_IDImg", DbType="Int NOT
          NULL")]
1248    public int IDImg
1249    {
1250      get
1251      {
1252        return this._IDImg;
1253      }
1254      set
1255      {
1256        if ((this._IDImg != value))
1257        {
1258          if (this._Individual.HasLoadedOrAssignedValue)
1259          {
1260            throw new System.Data.Linq.ForeignKeyReferenceAlreadyHasValueException();
1261          }
1262          this.OnIDImgChanging(value);
1263          this.SendPropertyChanging();
1264          this._IDImg = value;
1265          this.SendPropertyChanged("IDImg");
1266          this.OnIDImgChanged();
1267        }
1268      }
1269    }
1270
1271    [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_Img", DbType="Image",
          UpdateCheck=UpdateCheck.Never)]
1272    public System.Data.Linq.Binary Img
1273    {
1274      get
1275      {
1276        return this._Img;
1277      }
1278      set
1279      {
1280        if ((this._Img != value))
1281        {
1282          this.OnImgChanging(value);
1283          this.SendPropertyChanging();
1284          this._Img = value;
1285          this.SendPropertyChanged("Img");
1286          this.OnImgChanged();
1287        }
1288      }
1289    }
1290
1291    [global::System.Data.Linq.Mapping.AssociationAttribute(Name="Individual_KnownImg",
          Storage="_Individual", ThisKey="IDImg", OtherKey="ID", IsForeignKey=true,
          DeleteOnNull=true, DeleteRule="CASCADE")]
1292    public Individual Individual
1293    {
1294      get
1295      {
1296        return this._Individual.Entity;
1297      }
1298      set
1299      {
1300        Individual previousValue = this._Individual.Entity;
1301        if (((previousValue != value)
1302              || (this._Individual.HasLoadedOrAssignedValue == false)))
1303        {
1304          this.SendPropertyChanging();
1305          if ((previousValue != null))
```

```
            {
              this._Individual.Entity = null;
              previousValue.KnownImgs.Remove(this);
            }
            this._Individual.Entity = value;
            if ((value != null))
            {
              value.KnownImgs.Add(this);
              this._IDImg = value.ID;
            }
            else
            {
              this._IDImg = default(int);
            }
            this.SendPropertyChanged("Individual");
          }
        }
      }

      public event PropertyChangingEventHandler PropertyChanging;

      public event PropertyChangedEventHandler PropertyChanged;

      protected virtual void SendPropertyChanging()
      {
        if ((this.PropertyChanging != null))
        {
          this.PropertyChanging(this, emptyChangingEventArgs);
        }
      }

      protected virtual void SendPropertyChanged(String propertyName)
      {
        if ((this.PropertyChanged != null))
        {
          this.PropertyChanged(this, new PropertyChangedEventArgs(propertyName));
        }
      }
    }

    [global::System.Data.Linq.Mapping.TableAttribute(Name="dbo.Anomaly")]
    public partial class Anomaly : INotifyPropertyChanging, INotifyPropertyChanged
    {

      private static PropertyChangingEventArgs emptyChangingEventArgs = new
          PropertyChangingEventArgs(String.Empty);

      private int _AnomalyID;

      private int _EventID;

      private System.Nullable<double> _Severity;

      private System.Nullable<short> _isChecked;

      private System.Nullable<short> _AnomalyType;

      private EntityRef<DetectionEvent> _DetectionEvent;

      #region Extensibility Method Definitions
      partial void OnLoaded();
      partial void OnValidate(System.Data.Linq.ChangeAction action);
      partial void OnCreated();
      partial void OnAnomalyIDChanging(int value);
      partial void OnAnomalyIDChanged();
      partial void OnEventIDChanging(int value);
      partial void OnEventIDChanged();
      partial void OnSeverityChanging(System.Nullable<double> value);
```

```
1373      partial void OnSeverityChanged();
1374      partial void OnisCheckedChanging(System.Nullable<short> value);
1375      partial void OnisCheckedChanged();
1376      partial void OnAnomalyTypeChanging(System.Nullable<short> value);
1377      partial void OnAnomalyTypeChanged();
1378      #endregion
1379
1380      public Anomaly()
1381      {
1382        this._DetectionEvent = default(EntityRef<DetectionEvent>);
1383        OnCreated();
1384      }
1385
1386      [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_AnomalyID", AutoSync=
              AutoSync.OnInsert, DbType="Int NOT NULL IDENTITY", IsPrimaryKey=true,
              IsDbGenerated=true)]
1387      public int AnomalyID
1388      {
1389        get
1390        {
1391          return this._AnomalyID;
1392        }
1393        set
1394        {
1395          if ((this._AnomalyID != value))
1396          {
1397            this.OnAnomalyIDChanging(value);
1398            this.SendPropertyChanging();
1399            this._AnomalyID = value;
1400            this.SendPropertyChanged("AnomalyID");
1401            this.OnAnomalyIDChanged();
1402          }
1403        }
1404      }
1405
1406      [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_EventID", DbType="Int NOT
              NULL")]
1407      public int EventID
1408      {
1409        get
1410        {
1411          return this._EventID;
1412        }
1413        set
1414        {
1415          if ((this._EventID != value))
1416          {
1417            if (this._DetectionEvent.HasLoadedOrAssignedValue)
1418            {
1419              throw new System.Data.Linq.ForeignKeyReferenceAlreadyHasValueException();
1420            }
1421            this.OnEventIDChanging(value);
1422            this.SendPropertyChanging();
1423            this._EventID = value;
1424            this.SendPropertyChanged("EventID");
1425            this.OnEventIDChanged();
1426          }
1427        }
1428      }
1429
1430      [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_Severity", DbType="Float")
              ]
1431      public System.Nullable<double> Severity
1432      {
1433        get
1434        {
1435          return this._Severity;
1436        }
```

```
1437          set
1438          {
1439            if ((this._Severity != value))
1440            {
1441              this.OnSeverityChanging(value);
1442              this.SendPropertyChanging();
1443              this._Severity = value;
1444              this.SendPropertyChanged("Severity");
1445              this.OnSeverityChanged();
1446            }
1447          }
1448        }
1449
1450        [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_isChecked", DbType="
               SmallInt")]
1451        public System.Nullable<short> isChecked
1452        {
1453          get
1454          {
1455            return this._isChecked;
1456          }
1457          set
1458          {
1459            if ((this._isChecked != value))
1460            {
1461              this.OnisCheckedChanging(value);
1462              this.SendPropertyChanging();
1463              this._isChecked = value;
1464              this.SendPropertyChanged("isChecked");
1465              this.OnisCheckedChanged();
1466            }
1467          }
1468        }
1469
1470        [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_AnomalyType", DbType="
               SmallInt")]
1471        public System.Nullable<short> AnomalyType
1472        {
1473          get
1474          {
1475            return this._AnomalyType;
1476          }
1477          set
1478          {
1479            if ((this._AnomalyType != value))
1480            {
1481              this.OnAnomalyTypeChanging(value);
1482              this.SendPropertyChanging();
1483              this._AnomalyType = value;
1484              this.SendPropertyChanged("AnomalyType");
1485              this.OnAnomalyTypeChanged();
1486            }
1487          }
1488        }
1489
1490        [global::System.Data.Linq.Mapping.AssociationAttribute(Name="DetectionEvent_Anomaly",
               Storage="_DetectionEvent", ThisKey="EventID", OtherKey="EventID", IsForeignKey=
               true, DeleteOnNull=true, DeleteRule="CASCADE")]
1491        public DetectionEvent DetectionEvent
1492        {
1493          get
1494          {
1495            return this._DetectionEvent.Entity;
1496          }
1497          set
1498          {
1499            DetectionEvent previousValue = this._DetectionEvent.Entity;
1500            if (((previousValue != value)
```

```
1501                || (this._DetectionEvent.HasLoadedOrAssignedValue == false)))
1502          {
1503            this.SendPropertyChanging();
1504            if ((previousValue != null))
1505            {
1506              this._DetectionEvent.Entity = null;
1507              previousValue.Anomalies.Remove(this);
1508            }
1509            this._DetectionEvent.Entity = value;
1510            if ((value != null))
1511            {
1512              value.Anomalies.Add(this);
1513              this._EventID = value.EventID;
1514            }
1515            else
1516            {
1517              this._EventID = default(int);
1518            }
1519            this.SendPropertyChanged("DetectionEvent");
1520          }
1521        }
1522      }
1523
1524      public event PropertyChangingEventHandler PropertyChanging;
1525
1526      public event PropertyChangedEventHandler PropertyChanged;
1527
1528      protected virtual void SendPropertyChanging()
1529      {
1530        if ((this.PropertyChanging != null))
1531        {
1532          this.PropertyChanging(this, emptyChangingEventArgs);
1533        }
1534      }
1535
1536      protected virtual void SendPropertyChanged(String propertyName)
1537      {
1538        if ((this.PropertyChanged != null))
1539        {
1540          this.PropertyChanged(this, new PropertyChangedEventArgs(propertyName));
1541        }
1542      }
1543    }
1544  }
1545  #pragma warning restore 1591
```

Listing 16.34: Code for the Database Configuration File of the User Interface

16.15 Identity Engine - Matlab Demo Code

```
1  function [name, cam, frameNum, action] = identity_engine(frame, camera, xpos, ypos, h, w,
       img)
2      % input = struct {
3      % int objID,
4      % int frame,
5      % int camera,
6      % int xpos,
7      % int ypos,
8      % int h,
9      % int w,
10     % image img }
11
12     persistent prevIDs;
13     persistent lastCam;
14     persistent lastFrame;
```

```
    persistent lastPos;
    delay = 6;

    % read image
    figure
    subplot(2,2,1)
    imshow(img);

    % create grayscale image
    g_img = rgb2gray(img);
    subplot(2,2,2)
    imshow(g_img);

    % edge detection
    edgy_img = edge(g_img, 'canny');
    subplot(2,2,3)
    imshow(edgy_img);

    % find target area
    x = w;
    y = h;

    min_x = x;
    min_y = y;
    max_x = 1;
    max_y = 1;

    % find max and min x and y coordinates
    for j = 1:y
        for i = 1:x
            if edgy_img(i,j) == 1
                if i < min_x
                    min_x = i;
                elseif i > max_x
                    max_x = i;
                end

                if j < min_y
                    min_y = j;
                elseif j > max_y
                    max_y = j;
                end
            end
        end
    end

    % MATLAB demo only
    demo_img = img;

    % find outer radius
    r = (abs(max_x-min_x)/2 + abs(max_y-min_y)/2)/2;
    r_out = 0.69 * r;
    r_mid_out = 0.49 * r;
    r_mid_in = 0.45 * r;
    r_in = 0.25 * r;

    for i = 1:1:x
        for j = 1:y
            p = sqrt((i-(x/2))^2 + (j-(y/2))^2);
            if p <= (r_out * 1.01) && p >= (r_out * 0.99)
                demo_img(i, j, 1) = 255;
                demo_img(i, j, 2) = 140;
                demo_img(i, j, 3) = 0;
            elseif p <= (r_mid_out * 1.01) && p >= (r_mid_out * 0.99)
                demo_img(i, j, 1) = 255;
                demo_img(i, j, 2) = 140;
                demo_img(i, j, 3) = 0;
            elseif p <= (r_mid_in * 1.01) && p >= (r_mid_in * 0.99)
```

```
                demo_img(i, j, 1) = 255;
                demo_img(i, j, 2) = 140;
                demo_img(i, j, 3) = 0;
            elseif p <= (r_in * 1.01) && p >= (r_in * 0.99)
                demo_img(i, j, 1) = 255;
                demo_img(i, j, 2) = 140;
                demo_img(i, j, 3) = 0;
            end
        end
    end

    subplot(2,2,4)
    imshow(demo_img);

    % ID color bands
    % vars for each band: pixel count, sum, avg
    band1_pix = 0;
    band1_sum = [0, 0, 0];
    band1_avg = [0, 0, 0];

    band2_pix = 0;
    band2_sum = [0, 0, 0];
    band2_avg = [0, 0, 0];

    for j = 1:y
        for i = 1:x
            p = sqrt((i-(x/2))^2 + (j-(y/2))^2);
            % inside band 1
            if p <= r_out && p >= r_mid_out
                band1_pix = band1_pix + 1;
                band1_sum(1) = band1_sum(1) + cast(img(i,j,1),'uint32');
                band1_sum(2) = band1_sum(2) + cast(img(i,j,2),'uint32');
                band1_sum(3) = band1_sum(3) + cast(img(i,j,3),'uint32');
            % inside band 2
            elseif p <= r_mid_in && p >= r_in
                band2_pix = band2_pix + 1;
                band2_sum(1) = band2_sum(1) + cast(img(i,j,1),'uint32');
                band2_sum(2) = band2_sum(2) + cast(img(i,j,2),'uint32');
                band2_sum(3) = band2_sum(3) + cast(img(i,j,3),'uint32');
            end
        end
    end

    band1_avg(1) = cast(ceil(band1_sum(1) / band1_pix), 'uint8');
    band1_avg(2) = cast(ceil(band1_sum(2) / band1_pix), 'uint8');
    band1_avg(3) = cast(ceil(band1_sum(3) / band1_pix), 'uint8');
    band2_avg(1) = cast(ceil(band2_sum(1) / band2_pix), 'uint8');
    band2_avg(2) = cast(ceil(band2_sum(2) / band2_pix), 'uint8');
    band2_avg(3) = cast(ceil(band2_sum(3) / band2_pix), 'uint8');

    band1 = [0, 0, 0];
    band2 = [0, 0, 0];

    % thershold
    for i = 1:3
        if band1_avg(i) <= 32
            band1(i) = 0;
        elseif band1_avg(i) > 32 && band1_avg(i) <= 96
            band1(i) = 64;
        elseif band1_avg(i) > 96 && band1_avg(i) <= 160
            band1(i) = 128;
        elseif band1_avg(i) > 160 && band1_avg(i) <= 224
            band1(i) = 192;
        elseif band1_avg(i) > 224
            band1(i) = 255;
        end

        if band2_avg(i) <= 32
```

```
            band2(i) = 0;
        elseif band2_avg(i) > 32 && band2_avg(i) <= 96
            band2(i) = 64;
        elseif band2_avg(i) > 96 && band2_avg(i) <= 160
            band2(i) = 128;
        elseif band2_avg(i) > 160 && band2_avg(i) <= 224
            band2(i) = 192;
        elseif band2_avg(i) > 224
            band2(i) = 255;
        end
    end

    name = strcat(dec2hex(band1(1)), dec2hex(band1(2)), dec2hex(band1(3)), dec2hex(band2
        (1)), dec2hex(band2(2)), dec2hex(band2(3)));
    cam(1) = camera;
    frameNum(1) = frame;

    prev = 0;
    for i = 1:size(prevIDs)
        if strcmp(name, prevIDs(i)) == 1
            prev = 1;
            lastFrame(i) = frame;
            lastPos(i) = [xpos, ypos];
        end
    end
    if prev == 0
        prevIDs = [prevIDs, cellstr(name)];
        lastCam = [lastCam, camera];
        lastFrame = [lastFrame, frame];
        lastPos = [lastPos, [xpos,ypos]];
    end

    frame_size = [1280, 800];
    % 0 = entered normally
    % 1 = exited normally
    % 2 = exited abnormally
    % 3 = entered abnormally
    if (xpos <= frame_size(1) * 0.15 || xpos >= frame_size(1) * 0.85) && prev == 0
        action(1) = 0;
    elseif (xpos >= frame_size(1) * 0.15 || xpos <= frame_size(1) * 0.85) && prev == 0
        action(1) = 3;
    end

    for i = 1:size(prevIDs)
        if (lastPos(i,1) <= frame_size(1) * 0.15 || lastPos(i,1) >= frame_size(1) * 0.85)
            && frame == lastFrame(i) + delay
            name = [name, cellstr(prevIDs(i))];
            cam = [cam, lastCam(i)];
            frameNum = [frameNum, lastFrame(i)];
            action = [action, 1];
        elseif (lastPos(i,1) >= frame_size(1) * 0.15 || lastPos(i,1) <= frame_size(1) *
            0.85) && frame == lastFrame(i) + delay
            name = [name, cellstr(prevIDs(i))];
            cam = [cam, lastCam(i)];
            frameNum = [frameNum, lastFrame(i)];
            action = [action, 2];
        end
    end
end
```

Listing 16.35: Object Identification Demo

Chapter 17

Appendix - FPGA Lab Notebook

FPGA Flow

Thursday, December 26, 2013 3:30 PM

Device Tree (From Quartus) specifies memory maps and which drivers to use
In example altera-gpio and legs-gpio are already packaged with linux shource AND NOT in a bitbake recepie
Bitbake takes the device tree -> /meta-altera/recipes-gsrd/linux/files/socfpga.dts

- The "compatable" attribute sets driver compatibility
- Shared Memory (FPGA Memory Accessed over h2f_axi) does not appear to be included in device tree (no driver needed)?

Looks like you can add driver using bitbake recipie

Notation:
H2f_lw_axi - HPS to FPGA Lightweight AXI
H2f_axi - HPS to FPGA (High Bandwidth) AXI

From the description in the GHRD Description, the onchip ram is used as an "scratch pad" and serves only as a reference on how to use the high bandwith bridge and how to use FPGA RAM, it is directly accessible to the processor as "
memory map of soft IP peripherals, as viewed by the microprocessor unit (MPU), starts at HPS-to-FPGAaddress offset 0xC000_0000"

From <http://rocketboards.org/foswiki/Documentation/GSRDGhrd>

The HPS therefore, does not have access to the entire memory space of the FPGA ram (only 3FFF_FFFF

Investigate

Thursday, December 26, 2013 3:31 PM

Decode image with linux.
Copy to memory which is shared with FPGA
FPGA fetches images from memory
FPGA puts result in shared memory

See if we can use DMA conrroller

Linux Libraries

Thursday, December 26, 2013 3:33 PM

FFMPEG - supports H.264 and M-JPEG. Supports RTSP

Libav - supports H.264 and M-JPEG. Supports RTSP
Has OpenEmbeded recipe -> migrated to OpenEmbeded core
Mentioned in http://libav.org/download.html

Formats supported in http://libav.org/general.html

VLC uses Libav
Ubuntu has switched to Libav

Both have very similar suites of programs
Libav is a fork of FFMPEG

Will test them in Ubuntu VM with camera
Command Info:
http://libav.org/avplay.html#rtsp - contains both avconv and avplay commands

	h.264	Mjpeg
Libav	christopher@seniorDesignUbuntuLTS:~/linux-altera-3.7/drivers/leds$ avplay -max_delay 500000 -rtsp_transport udp rtsp://192.168.1.132:554/live1.sdp avplay version 0.8.9-4:0.8.9-0ubuntu0.12.04.1, Copyright (c) 2003-2013 the Libav developers built on Nov 9 2013 19:08:00 with gcc 4.6.3 [rtsp @ 0x7f95580008c0] Estimating duration from bitrate, this may be inaccurate Input #0, rtsp, from 'rtsp://192.168.1.132:554/live1.sdp': Metadata: title : RTSP/RTP stream 1 from DCS-2310L comment : live1.sdp with v2.0 Duration: N/A, start: 0.066667, bitrate: N/A Stream #0.0: Video: h264 (High), yuvj420p, 1280x800, 15 fps, 1k tbr, 90k tbn, 30 tbc Stream #0.1: Audio: pcm_mulaw, 8000 Hz, 1 channels, s16, 64 kb/s [avsink @ 0x7f954c001020] auto-inserting filter 'auto-inserted scaler 0' between the filter 'src' and the filter 'out' [scale @ 0x7f954c001700] w:1280 h:800 fmt:yuvj420p -> w:1280 h:800 fmt:yuv420p flags:0x4 46.26 A-V: -0.067 s:0.0 aq= 26KB vq= 1077KB sq= 0B f=0/0 f=0/0 christopher@seniorDesignUbuntuLTS:~/linux-altera-3.7/drivers/leds$	ERROR - expects 8 bit, is given 16 bit UNSUPPORTED
FFMPEG	Ubunbu hard wired ffmpeg to Libav	

Libav can transcode RTSP to image files!!!!! - including uncompressed using avconv
Filename needs to be in printf format!!! If encoding to images
http://stackoverflow.com/questions/9271357/ffmpeg-on-fb0-from-nexus-galaxy-error-could-not-get-frame-filename-number-2
-vframes sets number of frames to convert
-vcodec codec force video codec ('copy' to copy stream)
-f fmt force format
-an (output)
Disable audio recording.
-threads sets the number of threads to be used

To fix bad frames being written, http://www.eonlinegratis.com/2013/avconv-ffmpeg-on-raspberry-pi-often-corrupts-extracted-video-still-images/#sthash.q3OlJjKl.dpbs
-fflags discardcorrupt

avconv -max_delay 500 -rtsp_transport udp -fflags discardcorrupt -i rtsp://192.168.1.132:554/live1.sdp -fflags discardcorrupt -an testimg_%010d.bmp

You can extract images from a video, or create a video from many images:

For extracting images from a video:

avconv -i foo.avi -r 1 -s WxH -f image2 foo-%03d.jpeg

This will extract one video frame per second from the video and will output them in files named foo-001.jpeg, foo-002.jpeg, etc. Images will be rescaled to fit the new WxH values.

If you want to extract just a limited number of frames, you can use the above command in combination with the -vframes or -t option, or in combination with -ss to start extracting from a certain point in time.

For creating a video from many images:

avconv -f image2 -i foo-%03d.jpeg -r 12 -s WxH foo.avi

The syntax "foo-%03d.jpeg" specifies to use a decimal number composed of three digits padded with zeroes to express the sequence number. It is the same syntax supported by the C printf function, but only formats accepting a normal integer are suitable.

---

image2
Image file muxer.

The image file muxer writes video frames to image files.

The output filenames are specified by a pattern, which can be used to produce sequentially numbered series of files. The pattern may contain the string "%d" or "%0Nd", this string specifies the position of the characters representing a numbering in the
filenames. If the form "%0Nd" is used, the string representing the number in each filename is 0-padded to N digits. The literal character '%' can be specified in the pattern with the string "%%".

If the pattern contains "%d" or "%0Nd", the first filename of the file list specified will contain the

number 1, all the following numbers will be sequential.

The pattern may contain a suffix which is used to automatically determine the format of the image files to write.

For example the pattern "img-%03d.bmp" will specify a sequence of filenames of the form img-001.bmp, img-002.bmp, ..., img-010.bmp, etc. The pattern "img%%-%d.jpg" will specify a sequence of filenames of the form img%-1.jpg, img%-2.jpg, ..., img%-10.jpg,
etc.

The following example shows how to use avconv for creating a sequence of files img-001.jpeg, img-002.jpeg, ..., taking one image every second from the input video:

```
avconv -i in.avi -vsync 1 -r 1 -f image2 'img-%03d.jpeg'
```

Note that with avconv, if the format is not specified with the "-f" option and the output filename specifies an image file format, the image2 muxer is automatically selected, so the previous command can be written as:

```
avconv -i in.avi -vsync 1 -r 1 'img-%03d.jpeg'
```

Note also that the pattern must not necessarily contain "%d" or "%0Nd", for example to create a single image file img.jpeg from the input video you can employ the command:

```
avconv -i in.avi -f image2 -frames:v 1 img.jpeg
```

Good Command

Friday, December 27, 2013 11:01 PM

avconv -max_delay 500 -rtsp_transport tcp -fflags discardcorrupt -i rtsp://192.168.1.132:554/live2.sdp -fflags discardcorrupt -an testimg_%010d.bmp

-threads sets the number of threads to be used

This work for the most part because feed 2 was set to 15 frames/sec. If higher than that, libav cannot keep up and there are errors. UDP is often responcible for lost packets (and therefore corruption). This is ok for humans but it is bad for computer vision

HOWEVER UDP has less latency! And that may be important for us

To record H.264 video in same format (no transcoding) (NO ERRORS b/c TCP):

Errors can occur if UDP is used (due to packet loss) Error appears in playback - it tries its best to recover

avconv -max_delay 500 -rtsp_transport tcp -fflags discardcorrupt -i rtsp://192.168.1.132:554/live1.sdp -fflags discardcorrupt -an -vcodec copy test.mov

To convert H.264 video to bmps. NO ERRORS!!!!!!!
avconv -fflags discardcorrupt -i ./test.mov -fflags discardcorrupt -an testimg_%010d.bmp

Playing video
avplay test.mov

Gstremaer

Thursday, December 26, 2013 10:47 PM

http://stackoverflow.com/questions/4192871/playing-rtsp-with-python-gstreamer

Does not work to my knowlege

Using MJPG

Monday, January 6, 2014 11:35 PM

SO, After days of digging, it was revealed in an e-mail that the RTSP/RTP demuxer does not support MJPG!!!!!!

Why can VLC play MJPEG over RTP/RTSP you may ask, because it uses a library called live555.

Live555 is capable of capturing the MJPEG stream and recording it (very clearly I may add - better than even libav

I have tried it in ubuntu and not on the board yet. However, not all is lost! You still need libav to PLAY the recorded MJPEG which avplay does quite well if it has the mjpg extension.

I had to force RTSP/RTP over TCP because of the school firewall. Apparently, this was causing the zero-size files I was seeing.

The commands are as follows: (the -t forces TCP, -d specifies how long to record)

openRTSP -t -d 10 rtsp://172.16.122.113:554/live2.sdp
mv video-JPEG-1 video-JPEG-1.mjpg
avplay video-JPEG-1.mjpg

Can only record video using
openRTSP -t -v -d 10 -F mjpg-video rtsp://172.16.122.113:554/live2.sdp > mjpeg-vid.mjpg

I will recompile and try it on the board

Live555 is a library and I should be able to extract raw data from it

http://live555.com/liveMedia/ is an example media player

Live555 has an OE recipie that is in the meta-oe repo. There is also a recipie in the oe-classic repo

http://layers.openembedded.org/layerindex/recipe/1176/

It work on the board!!!!!!

Is able to record MJPG at 30 FPS!!!
However, FPS drops the longer the capture runs. But no errors. Unlike libav which corrupts images in an attempt to catch up, live555 drops frames to keep up

I think this can be fixed by writing custom program that only uses ram. I think that the bottleneck is writing file to SDcard

-m option saves each frame as a separate file!!!!

Yacto:

Thursday, December 26, 2013 4:16 PM

How to include a recipe https://wiki.yoctoproject.org/wiki/How_do_I#Q:_How_do_I_put_my_recipe_into_Yocto.3F
Patches: https://wiki.yoctoproject.org/wiki/Recipe_%26_Patch_Style_Guide

https://wiki.yoctoproject.org/wiki/Recipe_%26_Patch_Style_Guide

Yacto has a http://en.wikipedia.org/wiki/GStreamer recipy which relies on FFMPEG libs

oe-core has libav

Need to find how to import oe-core layer into yacto
http://www.openembedded.org/wiki/Getting_started

Error with license
https://community.freescale.com/thread/316329

Steps:

1. Cd ~
2. Mkdir oe
3. Cd oe
4. git clone git://git.openembedded.org/openembedded-core oe-core
5. Copy oe/oe-core/meta/recipes-multimedia/libav to yocto-ghrd/meta/recipes-multimedia
6. Add libav to ~/yocto-ghrd/meta-altera/recipes-core/images/altera-gsrd-image.bb
7. Will get license error
8. Append LICENSE_FLAGS_WHITELIST = "commercial" to ~/yocto-ghrd/build/conf/local.conf (https://community.freescale.com/thread/316329)
9. There will be a dependence error with yasm
10. Copy ~/oe/oe-core/meta/recipes-devtools/yasm to ~/yocto-ghrd/meta/recipes-devtools (http://layers.openembedded.org/layerindex/recipe/5638/)
11. Will get dependence error with x264
12. Copy ~/oe/oe-core/meta/recipes-multimedia/x264 to ~/yocto-ghrd/meta/recipes-multimedia (http://layers.openembedded.org/layerindex/recipe/5642/)
13. source /home/christopher/yocto-ghrd/altera-init /home/christopher/yocto-ghrd/build
14. bitbake virtual/bootloader virtual/kernel altera-gsrd-image

Adding live555 library
Live555 has a recipie in meta

1. Cd ~/oe
2. git clone git://git.openembedded.org/meta-openembedded meta-oe
3. cd ~/yocto-ghrd/
4. Mkdir meta-oe
5. Cd meta-oe
6. cp -r ~/oe/meta-oe/meta-oe/conf/ .
7. cp -r ~/oe/meta-oe/meta-oe/licenses/ .
8. cp -r ~/oe/meta-oe/meta-oe/COPYING.MIT .
9. cp -r ~/oe/meta-oe/meta-oe/README .
10. mkdir recipes-multimedia
11. Cd recipes-multimedia

12. cp -r ~/oe/meta-oe/meta-oe/recipes-multimedia/live555/ .
13. Insert /home/christopher/yocto-ghrd/meta-oe \ into ~/yocto-ghrd/build/conf/bblayers.conf so that the file looks like:

    /home/christopher/yocto-ghrd/meta-oe \
    /home/christopher/yocto-ghrd/meta \
    /home/christopher/yocto-ghrd/meta-yocto \

14. In ~/yocto-ghrd/meta-oe/conf/layer.conf , modify the priority value to 1 (we want these packages to have the lowest precedent since we are integrating them into yacto and some may be incompatible)
15. Add live555 live555-openrtsp live555-mediaserver to ~/yocto-ghrd/meta-altera/recipes-core/images/altera-gsrd-image.bb
16. source /home/christopher/yocto-ghrd/altera-init /home/christopher/yocto-ghrd/build
17. bitbake virtual/bootloader virtual/kernel altera-gsrd-image
18. NOTE: I modified the Live555 bb recipie to get the current version as it is what the current example files are based on.

Camera Feed

Friday, December 27, 2013 10:33 PM

RTP over RTSP

Bitbake Add Layer

Saturday, December 28, 2013 1:25 AM

Layers are listed in ~/yocto-ghrd/build/conf/bblayer.conf
http://www.openembedded.org/wiki/Creating_a_new_Layer
http://www.yoctoproject.org/docs/current/dev-manual/dev-manual.html#prioritizing-your-layer

~/yocto-ghrd/build/conf/local.conf has other parameters
- How many threads bitbake uses
- The machine (fpga type)
- Distro name
- Image config options

Bitbake clean

Saturday, December 28, 2013 3:48 PM

bitbake -c clean TARGET

Making SD Card

Saturday, December 28, 2013 5:16 PM

Since Using older version of Yocto, instructions on
http://www.rocketboards.org/foswiki/Documentation/GSRDSdCard are not quite right
The help text given is not the same as the help text with the SD Card image script in the program

Use instead!!!!!
http://rocketboards.org/foswiki/Documentation/AlteraSoCDevelopmentBoardYoctoGettingStarted

```
$ cd ~/yocto-ghrd/build/tmp/deploy/images
$ sudo rm -rf rootfs
$ mkdir rootfs
$ cd rootfs
$ sudo tar xzf ../altera-gsrd-image-socfpga_cyclone5.tar.gz
$ cd ..
$ rm sd_image_yocto.bin
$ sudo /opt/altera-linux-ghrd/bin/make_sdimage.sh \
-k uImage,socfpga.dtb \
-rp u-boot-spl-socfpga_cyclone5.bin \
-t ~/altera/13.0sp1/embedded/host_tools/altera/mkpimage/mkpimage \
-b u-boot-socfpga_cyclone5.img \
-r rootfs/ \
-o sd_image_yocto.bin
```

From <http://rocketboards.org/foswiki/Documentation/AlteraSoCDevelopmentBoardYoctoGettingStarted>

Documentation to flash image to disk is at
http://www.rocketboards.org/foswiki/Documentation/GSRDBootLinuxSd
Instead of downloading image file, copy the built one from ~/yocto-ghrd/build/tmp/deploy/images/altera-gsrd-image-socfpga_cyclone5.tar.gz

//to list devices
cat /proc/partitions

sudo dd if=sd_image_yocto.bin of=/dev/sdb bs=1M
sudo sync

Printout

Saturday, December 28, 2013 6:56 PM

Linux Getting Started on Altera SoC Development Board - Using Yocto Source Package

Last modified by JohnCarter on 28 Nov 2013 - 00:39 - r9
Tagged Kernel, Linux, Yocto

Table of Contents

Introduction

This page presents the instructions on how to rebuild Linux by using the Yocto Source Package.

The Yocto Source package is an installer file provided by Altera that contains the Yocto build system, Yocto recipes and also the necessary dependencies to compile the Altera Linux bootloader, kernel and root filesystem.

Note: the Yocto build system and recipes for Altera Linux can be also obtained from the Git trees. See Linux - Getting Started Using Git Trees section for details on that. If the Git tree option is used, all the necessary dependencies would be downloaded from Internet.

For more details about Yocto see Altera Yocto Project User Manual (Danny).

Host Setup

The Altera Linux Yocto Source package was tested to build correctly with CentOS 6.3 and Ubuntu 12.04. It may also work with other distributions. For a list of distributions against which Yocto project was tested see https://wiki.yoctoproject.org/wiki/Distribution_Support. This section presents the packages that need to be installed on the host PC to allow the Yocto Source Package to be built. Since each machine may have been installed in a different way, the packages listed here are not an exhaustive list. Add packages as necessary if you encounter issues.

CentOS 6.3

These are the required packages that need to be installed:

```
$ sudo yum update
$ sudo yum groupinstall "Development Tools"
$ sudo yum install texi2html texinfo glibc-devel chrpath
```

If the host machine runs the 64bit version of the OS, then the following additional packages need to be installed:

```
$ sudo yum install glibc.i686 libgcc.i686 libstdc++.i686 glibc-devel.i686 ncurses-libs.i686 zlib.i686
```

Ubuntu 12.04

These are the required packages that need to be installed:

```
$ sudo apt-get update
$ sudo apt-get upgrade
$ sudo apt-get install sed wget cvs subversion git-core coreutils unzip texi2html texinfo libsdl1.2-dev docbook-ut
ils gawk python-pysqlite2 diffstat help2man make gcc build-essential g++ desktop-file-utils chrpath libgl1-mesa-de
v libglu1-mesa-dev mercurial autoconf automake groff libtool xterm
```

If the host machine runs the 64bit version of the OS, then the following additional packages need to be installed:

```
$ sudo apt-get install ia32-libs
```

Ubuntu 13.10 - Saucy Salamander

As per 12.04 above, except ia32-libs no longer exists in Saucy.

sudo apt-get install libqt4-gui:i386 itcl3:i386 itcl3-dev:i386 libgomp1:i386 libqtwebkit4:i386 libtbb2:i386 tcl8.5-lib:i386 tcl8.5-dev:i386

The above list was guessed from the output of ldd and apt-file

SoC EDS

The Altera SoC EDS toolset needs to be installed, since it provides the mkpimage tool that adds the BootROM signature on top of the Preloader image.

The SoC EDS can be obtained from https://www.altera.com/download/software/soc-eds. If you do not have a license, follow the instructions from the mentioned web page to obtain the free **Web Edition** license.

Obtaining Yocto Source Package

The Altera Yocto Source Package can be downloaded from the Altera website. The following commands can be used to dowload the package and mark it as executable:

↑

```
$ wget http://download.altera.com/akdlm/software/acdsinst/13.0/156/ib_installers/linux-socfpga-13.02-RC10-src.bsx
$ chmod +x linux-socfpga-13.02-RC10-src.bsx
```

Setting Up Yocto

The Yocto Source Package may be installed in a publicly accessible location, as this step can be shared by all the users on the system (or on the filesystem if your company uses a network share). The default location is `/opt/altera-linux`. If you wish to use this location, you will likely need root access in order to access this directory. This is why the command shown below is ran using `sudo`.

```
$ sudo ./linux-socfpga-13.02-RC10-src.bsx
```

The next step is to install a local set of Yocto recipes. This could be done in a shared location, but if someone wants or needs to modify them they should have their own version. The default install location for this is within your home directory:

```
$ /opt/altera-linux/bin/install_altera_socfpga_src.sh ~/yocto
```

Last step is to create a build directory. By keeping this separate from your yocto source you can erase your entire build without fear of deleting your yocto sources. Also, you can have several build directories, each with its own configuration, all based on the same yocto source. The script serves 2 purposes. First, it creates the new build directory using Altera's default configuration. Secondly, it sets some shell variables that are required for building.

```
$ source ~/yocto/altera-init ~/yocto/build
```

Note: If you start a new shell you will need to run the above command to set the shell variables again.

Building U-Boot/Kernel/Rootfs

In order to build u boot from within the build directory you need to execute this command:

```
$ bitbake virtual/bootloader
```

or the equivalent:

```
$ bitbake u-boot
```

In order to build linux from within the build directory:

```
$ bitbake virtual/kernel
```

or the equivalent:

```
$ bitbake linux-altera
```

In order to build the root filesystem:

```
$ bitbake altera-image
```

If you wish to build a much smaller filesystem for small boot mediums:

```
$ bitbake altera-image-minimal
```

The first time you build the Yocto Source Package it may take up to several hours depending on your host machine.

Once finished, all images should be generated in `~/build/tmp/deploy/images`. Some of the most important files created are:

File	Description
u-boot-spl-socfpga_cyclone5	SPL Preloader ELF file
u-boot-spl-socfpga_cyclone5.bin	SPL Preloader Binary
u-boot-socfpga_cyclone5	U-Boot ELF file
u-boot-socfpga_cyclone5.bin	U-Boot binary file
u-boot-socfpga_cyclone5.img	U-Boot image
socfpga_cyclone5.dtb	Device Tree Binary
vmlinux	Kernel ELF file
uImage	Kernel Image
altera-image-socfpga_cyclone5.cpio	Root filesystem in cpio archive format
altera-image-socfpga_cyclone5.ext3	Root filesystem as ext3 image
altera-image-socfpga_cyclone5.jffs2	Root filesystem as jffs2 image
altera-image-minimal-socfpga_cyclone5.jffs2	Minimal root filesystem as jffs2 image
altera-image-socfpga_cyclone5.tar.gz	Root filesystem in tar gzip archive format

↑

The above files are actually links to the actual build resulted files. Each build file has a timestamp attached to its name and each time it it rebuilt, Yocto updates the link to point to the latest file.

Creating SD Card Image

To boot the linux images on SoC FPGA development kit, you need to write the images you just built with Yocto into one of the two Flash devices: SDMMC or QSPI. For this guide, we will use SDMMC due to its easy detachability. For SDMMC boot, all boot images will be located inside SD/MMC card. A script is provided with the release that will create an SD card image, ready to be deployed.

The script relies on a tool, named `mkpimage`, which creates the correct preloader image that the SoCFPGA Boot ROM accepts.This tool is provided with the SoC EDS release. See SoC EDS section for instructions.

After SoC EDS is installed, run the the following commands to locate mkpimagetool:

```
$ ~/altera/13.0/embedded/embedded_command_shell.sh
$ which mkpimage
```

The following files resulted from the Yocto build will be used, from `~/yocto/build/tmp/deploy/images`:

File	Description
u-boot-spl-socfpga_cyclone5.bin	SPL Preloader Binary
u-boot-socfpga_cyclone5.img	U-Boot image
socfpga_cyclone5.dtb	Device Tree Binary
uImage	Kernel Image
altera-image-socfpga_cyclone5.tar.gz	Root File System

The following commands can be used to create the SD card image:

```
$ cd ~/yocto/build/tmp/deploy/images
$ cp -f socfpga_cyclone5.dtb socfpga.dtb
$ mkdir rootfs
$ cd rootfs
$ sudo tar xzf ../altera-image-socfpga_cyclone5.tar.gz
$ cd ..
$ sudo /opt/altera-linux/bin/make_sdimage.sh \
  -k uImage,socfpga.dtb\
  -rp u-boot-spl-socfpga_cyclone5.bin \
  -t /home/<user>/altera/13.0/embedded/host_tools/altera/mkpimage/mkpimage \
  -b u-boot-socfpga_cyclone5.img \
  -r rootfs/ \
  -o sd_image_yocto.bin
```

The above commands do the following:

- Make a copy of the `socfpga_cyclone5.dtb` as `socfpga.dtb` because that is the name that U-boot expects to see
- Extracts the root file system into a folder, using `sudo` to be able to create the device nodes
- Invokes the `make_sdimage.sh` script to create the SD card image named `sd_image_yocto.bin`. This is also executed using `sudo` because it uses system utilities.

Notes:

- The above commands assume default installation paths were used for all tools. Update accordingly if non-default paths were used.
- Please replace `<user>` with your user name.

You can also run make_sdimage.sh as follows, without parameters, to obtain more information about the tool:

```
$ /opt/altera-linux/bin/make_sdimage.sh
```

Using SD Card Image

See Linux Getting Started on Altera SoC Development Board - Using SD Card Image for instructions on how to use the SD card image to boot Linux. Simply use the newly created SD card image instead of the precompiled one.

Partially Updating SD Card

This section presents how to update various parts on the SD card, without having to re-create and write to the SD card the whole image. You will first need to have a working SD card before being able to run the commands presented in this section.

The commands presented assume the SD card device is `/dev/sdx`. Refer to Linux Getting Started on Altera SoC Development Board - Using SD Card Image for instructions on how to determine which device corresponds to the SD card reader on your host PC.

Updating SPL Preloader:

```
$ sudo dd if=u-boot-spl-socfpga_cyclone5.img of=/dev/sdx3 bs=64k seek=0
```

Updating U-boot Image

```
$ sudo dd if=u-boot-socfpga_cyclone5.img of=/dev/sdx3 bs=64K seek=4
```

Updating Device Tree Binary:

```
$ sudo mkdir sdcard
$ sudo mount /dev/sdx1 sdcard/
$ sudo cp socfpga_cyclone5.dtb sdcard/socfpga.dtb
$ sudo umount sdcard/
```

Updating Kernel Image:

```
$ sudo mkdir sdcard
$ sudo mount /dev/sdx1 sdcard/
$ sudo cp uImage-socfpga_cyclone5.bin sdcard/uImage
$ sudo umount sdcard/
```

Updating Root Filesystem:

```
$ sudo dd if=altera-image-socfpga_cyclone5.ext3 of=/dev/sdx2
```

↑

Results on Altera Board

Sunday, December 29, 2013 1:05 AM

H.264 Recording works with aconv!!!!!
Get many glitches is using 30 pfs stream (feed1)

Using feed 2 at 15 fps is much better (but still has some gliches)

Suggest using only 15 fps stream

BMP transcode did not work well even at 15 fps (or with 2 threads)

JPG transcode a bit better but still incredibly error prone (9 fps max and routiennly goes to 6)

Problem is that FPS being transcoded is slower than incoming resulting in a backlog

Will NEED to write software using the libav libraries that decodes individual frames and places them in shared mem (hopefully not having to re-encode will make things faster)

Able to avoid temporarily by encoding to jpg and having strem capped at 4 fps

Decoding of pre-recorded h.264 to jpg ranged from 6 to 10 fps

Investigate saving raw image (avoid re-encoding)

Recording raw MJPEG works well with Live555, even at 30 FPS

Avplay can be used to play and avconv can be used to transcode to bitmaps. The speed of transcoding to BMP degrades over time (down to 4 FPS)
The speed of transcoding to JPG is stuck at 9 FPS

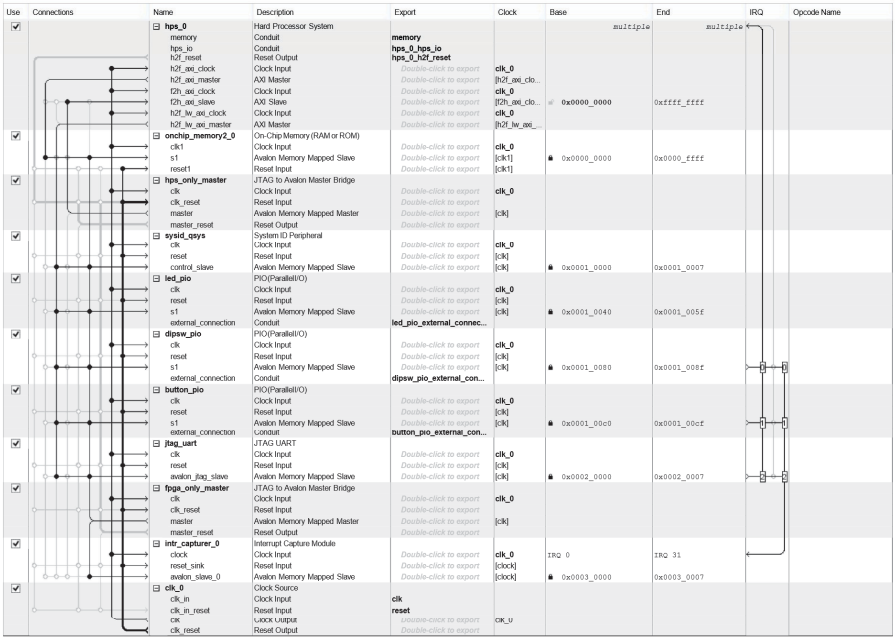
I believe this is due to a bottleneck in writing to the SD Card. For only 3 sec of video, transcoding to BMP takes ~300 MB!!! Each BMP is 4.6 MB

Is able to record MJPG at 30 FPS!!!
However, FPS drops the longer the capture runs. :(But no errors. Unlike libav which corrupts images in an attempt to catch up, live555 drops frames to keep up

I think this can be fixed by writing custom program that only uses ram. I think that the bottleneck is writing file to SDcard

Printout

Friday, January 3, 2014 3:53 PM


Use
Connections
Name
Description
Export
Clock
Base
End
IRQ
Opcode Name
hps_0
Hard Processor System
memory
Conduit
memory
multiple
multiple
hps_io
Conduit
hps_0_hps_io
h2f_reset
Reset Output
hps_0_h2f_reset
h2f_axi_clock
Clock Input
Double-click to export
clk_0
h2f_axi_master
AXI Master
Double-click to export
[h2f_axi_clo...
f2h_axi_clock
Clock Input
Double-click to export
clk_0
f2h_axi_slave
AXI Slave
Double-click to export
[f2h_axi_clo...
0x0000_0000
0xffff_ffff
h2f_lw_axi_clock
Clock Input
Double-click to export
clk_0
h2f_lw_axi_master
AXI Master
Double-click to export
[h2f_lw_axi_...
onchip_memory2_0
On-Chip Memory (RAM or ROM)
clk1
Clock Input
Double-click to export
clk_0
s1
Avalon Memory Mapped Slave
Double-click to export
[clk1]
0x0000_0000
0x0000_ffff
reset1
Reset Input
Double-click to export
[clk1]
hps_only_master
JTAG to Avalon Master Bridge
clk
Clock Input
Double-click to export
clk_0
clk_reset
Reset Input
Double-click to export
master
Avalon Memory Mapped Master
Double-click to export
[clk]
master_reset
Reset Output
Double-click to export
sysid_qsys
System ID Peripheral
clk
Clock Input
Double-click to export
clk_0
reset
Reset Input
Double-click to export
[clk]
control_slave
Avalon Memory Mapped Slave
Double-click to export
[clk]
0x0001_0000
0x0001_0007
led_pio
PIO (Parallel I/O)
clk
Clock Input
Double-click to export
clk_0
reset
Reset Input
Double-click to export
[clk]
s1
Avalon Memory Mapped Slave
Double-click to export
[clk]
0x0001_0040
0x0001_005f
external_connection
Conduit
led_pio_external_connec...
dipsw_pio
PIO (Parallel I/O)
clk
Clock Input
Double-click to export
clk_0
reset
Reset Input
Double-click to export
[clk]
s1
Avalon Memory Mapped Slave
Double-click to export
[clk]
0x0001_0080
0x0001_008f
external_connection
Conduit
dipsw_pio_external_con...
button_pio
PIO (Parallel I/O)
clk
Clock Input
Double-click to export
clk_0
reset
Reset Input
Double-click to export
[clk]
s1
Avalon Memory Mapped Slave
Double-click to export
[clk]
0x0001_00c0
0x0001_00cf
external_connection
Conduit
button_pio_external_con...
jtag_uart
JTAG UART
clk
Clock Input
Double-click to export
clk_0
reset
Reset Input
Double-click to export
[clk]
avalon_jtag_slave
Avalon Memory Mapped Slave
Double-click to export
[clk]
0x0002_0000
0x0002_0007
fpga_only_master
JTAG to Avalon Master Bridge
clk
Clock Input
Double-click to export
clk_0
clk_reset
Reset Input
Double-click to export
master
Avalon Memory Mapped Master
Double-click to export
[clk]
master_reset
Reset Output
Double-click to export
intr_capturer_0
Interrupt Capture Module
clock
Clock Input
Double-click to export
clk_0
IRQ 0
IRQ 31
reset_sink
Reset Input
Double-click to export
[clock]
avalon_slave_0
Avalon Memory Mapped Slave
Double-click to export
[clock]
0x0003_0000
0x0003_0007
clk_0
Clock Source
clk_in
Clock Input
clk
clk_in_reset
Reset Input
reset
clk
Clock Output
Double-click to export
clk_0
clk_reset
Reset Output
Double-click to export

Patching

Thursday, January 9, 2014 11:24 PM

Patched Live555 example to up buffer size in testRTSPClient.cpp

See .patch file for example

Live555 Info

Thursday, January 9, 2014 11:25 PM

Buffer had to be increased in demo program to handel MJPEG. Did this with a patch.

Can get timestamp for received frame!!!!!

Also, -m option saves each frame as a file!!!!

Decoding Video to Uncompressed Frame

Wednesday, January 15, 2014 12:23 AM

Libjpeg can be used to decode jpeg images which live555 can create from MJPEG streams. H.264 reqires Libav to decode

Libav decoding tutorial: http://www.inb.uni-luebeck.de/~boehme/using_libavcodec.html

Decoding JPEG

Friday, January 17, 2014 11:20 PM

There are 2 commonly used libraries for jpeg decoding, one is a fork of the other and both have oe recipes.

Libjpeg - the original library
Libjpeg - turbo - optimized library for SIMD including NEON (ARM)

Recipes can be found at
http://layers.openembedded.org/layerindex/branch/master/recipes/?q=libjpeg

Libjpeg - http://layers.openembedded.org/layerindex/recipe/606/ (part of oe core)
Libjpeg-turbo - http://layers.openembedded.org/layerindex/recipe/1139/ (part of meta-oe)

Libjpeg-turbo reports to have 2-4x performance increase over traditional library

Will probably go with libjpeg-turbo implementation due to performance and fact that ARM processor has NEON

Another bonus is that libjpeg-turbo is compliant with the libjpeg version 8 API.

Website for libjpeg-turbo is http://www.libjpeg-turbo.org/

Additional info about using libjpeg at http://www.cim.mcgill.ca/~junaed/libjpeg.php

Altera VIP Suite (Video Processing Suite)

Saturday, January 18, 2014 11:31 PM

Has FPGA Ip to convert from RGB to Avalon-ST Video (a bus standard used by their video processing IP) and back.

While this would work, we are getting RGB from the JPEGs, the converters are so-called "clocked converters". They expect video to be coming in at a regular rate and to do real time, low latency processing on that video (ie. Scaling, mixing, deinterlacing). While we are doing some of the same things, we don't have a guaranteed steady stream of video, nor are we treating this like video. We are looking at individual frames and are extracting figures from them. It is a different sort of problem, we are not simply taking in video, processing it, and outputting the result.

~~While we can keep this library in mind, I think we will wind up writing our own IP to do much of this. At the very least, to use it, we need some IP to provide the RGB data and generate a dummy clock for the converter IP provided by altera.~~

SRATCH that. The VIP reference design pulls images from memory and overlays them on clocked video!!! It uses an ip component called a "Frame Reader". This is sort of what we are doing (except there is no clocked video input). It may be worth looking into this. However, it is still overkill for what we are doing but it may provide the key for getting video into and out of the FPGA.

However, getting the video back into ram is a little more difficult. There are frame buffers but they are meant to take video in and stream it back out. Since we would need to write our own video logic that is compliant to this standard if we use it, this will probably not be a huge problem to make a to memory output. Plus, our to memory output is pieces of the image, not the whole thing

After reviewing custom IP tutorial, we may want to use avalon-st (streaming). This is because our video processing is likely unidirectional and we don't want the trouble of dealing with a memory map between DSP components (yuck). Avalon-ST is how almost all DSP is done with altera. Can write custom component which takes avalon st data and outputs it to memory via avalon-mm. Could be that on the output of object detection, it only raises the valid flag when there is an object to report. The Avalon-ST to Avalon-MM would take that message and store it in mem.

Should likely use Avalon-ST Video as the way to transmit the image data as it supplies both the standard way to transmit the pixle data aswell as control packets and user defined packets

Linux Module Development

Saturday, April 26, 2014 7:04 PM

Rocketboards - in the forum
One person has made a sample module for writing a driver for the GHRD

This was written by

http://rocketboards.org/foswiki/Projects/MyFirstModule

ProFromDover

Generating Device Tree

Saturday, April 26, 2014 7:57 PM

The steps to generate the device tree are at
http://rocketboards.org/foswiki/Documentation/GSRDDeviceTreeGenerator

As one would expect by this point, driver devlopment for the SoC kit is not venilla Linux driver devlopment. Unfortunatly, because Altera already had the NIOS, there can be a lot of confusion on what the correct way to devlope a driver is. In many cases, altera IP has been devloped so that it can work with either the NIOS or the linux soc. I should be clear that not all IP has linux drivers. It appears that one must devlpe a driver or use an existing one. The driver, IP link is accomplished in the _hw.tcl file by setting a "compatible with" flag which corresponds to the same value in a device driver.
http://rocketboards.org/foswiki/Documentation/DeviceTreeGenerator#Adding_Device_Tree_Generation_Support_to_an_IP_Block

Copy the resulting socfpga.dts file onto the linux build system in the
~/yocto-ghrd/meta-altera/recipes-gsrd/linux/files and replace the current file

NOTE in 13.0sp1, there is no HPS timing file, so don't try to include it as you will not be able to find it

Rerun bitbake on the linux image

Running these commands caused linux to fail to load, The message from serial was

```
U-Boot SPL 2012.10 (Jan 09 2014 - 17:06:33)
SDRAM : Initializing MMR registers
SDRAM : Calibrationg PHY
SEQ.C: Preparing to start memory calibration
SEQ.C: CALIBRATION PASSED
DESIGNWARE SD/MMC: 0


U-Boot 2012.10 (Jan 09 2014 - 17:06:33)

CPU   : Altera SOCFPGA Platform
BOARD : Altera SOCFPGA Cyclone 5 Board
DRAM:  1 GiB
MMC:   DESIGNWARE SD/MMC: 0
*** Warning - bad CRC, using default environment

In:    serial
Out:   serial
Err:   serial
Net:   mii0
Warning: failed to set MAC address

Hit any key to stop autoboot:  0
reading uImage

2693952 bytes read
reading socfpga.dtb
```

```
9488 bytes read
## Booting kernel from Legacy Image at 00007fc0 ...
   Image Name:   Linux-3.7.0
   Image Type:   ARM Linux Kernel Image (uncompressed)
   Data Size:    2693888 Bytes = 2.6 MiB
   Load Address: 00008000
   Entry Point:  00008000
## Flattened Device Tree blob at 00000100
   Booting using the fdt blob at 0x00000100
   XIP Kernel Image ... OK
OK
   Loading Device Tree to 0fff9000, end 0fffe50f ... OK

Starting kernel ...
```

Based on info from, http://www.alteraforum.com/forum/showthread.php?t=41647
need to remove the sysID from the device tree file

Driver Stuff

Sunday, April 27, 2014 3:41 AM

Drivers can be found in the linux_altera_3.7/drivers

Altera PIO driver can be found in drivers/gpio/altera-gpio
This is based on another driver.

The driver uses the platform_device.h library which can be found under includes/kernel
Documentation is at /Documentation/driver-model

Flashing FPGA

Saturday, May 3, 2014 3:33 PM

Follow instructions on http://rocketboards.org/foswiki/Documentation/GSRDQuartusProgrammer

1. Compile Hardware design (instructions at http://rocketboards.org/foswiki/Documentation/GSRDCompileHardwareDesign I figured out a lot of this myself as we made many modifications). It should just work.)
    a. Generate Qsys system.
    b. Run tcl scripts for memory assignments (if necessary, they should have already been run)
    c. run compiler
2. Flash SD card but DO NOT insert it into dev board
3. Set SW2 DW 2 (Fact Load) to OFF - all switches to right
4. Start programmer from Quartus II
5. Follow instructions in ug_cv_sco_dev_kit

Writing to memory

Sunday, May 4, 2014 12:42 AM

Memory can be written to directly using mmap and /dev/mem

when opening /dev/mem, the O_SYNC option must be set to ensure write actually gets to memory (equiv to calling sync after each write according to man page)

HPS to FPGA DDR3 Notes

Friday, May 9, 2014 5:25 PM

FPGA does not have access to entire FPGA DDR3 memory space. The h2f AXI master starts at 0xC000 0000 from the perspective of the ARM. Only 28 bits of the 32 bit address space of the memory can be accessed, yielding only 268.4 MB or 1/4 of the DDR3 memory space. This is consistent with the concept of bridge address spaces as detailed in http://www.altera.com/literature/ug/ug_sopc_builder.pdf. This should be OK for us since the amount of data being stored is relatively low. There are of course ways around this but that would be a step for the next prototype.

New Note:
It looks like the 0x0FFF FFFF is a limitation in Qsys. More investigation is required to figure out why this is the case but I will not focus on this now

QoS Stats RTSP MJPEG

Thursday, May 22, 2014 2:57 AM

```
christopher@seniorDesignUbuntuLTS:~/live555$ openRTSP -t -v -Q -F mjpg-video
rtsp://192.168.11.31:554/live2.sdp > /dev/null
Opening connection to 192.168.11.31, port 554...
...remote connection opened
Sending request: OPTIONS rtsp://192.168.11.31:554/live2.sdp RTSP/1.0
CSeq: 2
User-Agent: openRTSP (LIVE555 Streaming Media v2011.12.23)


Received 152 new bytes of response data.
Received a complete OPTIONS response:
RTSP/1.0 200 OK
CSeq: 2
Date: Thu, May 22 2014 09:56:44 GMT
Public: OPTIONS, DESCRIBE, SETUP, TEARDOWN, PLAY, PAUSE, GET_PARAMETER, SET_PARAMETER


Sending request: DESCRIBE rtsp://192.168.11.31:554/live2.sdp RTSP/1.0
CSeq: 3
User-Agent: openRTSP (LIVE555 Streaming Media v2011.12.23)
Accept: application/sdp


Received 574 new bytes of response data.
Received a complete DESCRIBE response:
RTSP/1.0 200 OK
CSeq: 3
Date: Thu, May 22 2014 09:56:44 GMT
Content-Base: rtsp://192.168.11.31/live2.sdp/
Content-Type: application/sdp
Content-Length: 410

v=0
o=- 1400752572597196 1 IN IP4 192.168.11.31
s=RTSP/RTP stream 2 from DCS-2332L-CAM1
i=live2.sdp with v2.0
t=0 0
a=type:broadcast
a=control:*
a=range:npt=0-
a=x-qt-text-nam:RTSP/RTP stream 2 from DCS-2332L-CAM1
a=x-qt-text-inf:live2.sdp
m=video 0 RTP/AVP 26
c=IN IP4 0.0.0.0
b=AS:1500
a=x-dimensions:1280,800
a=control:track1
m=audio 0 RTP/AVP 0
```

c=IN IP4 0.0.0.0
b=AS:64
a=control:track2

Opened URL "rtsp://192.168.11.31:554/live2.sdp", returning a SDP description:
v=0
o=- 1400752572597196 1 IN IP4 192.168.11.31
s=RTSP/RTP stream 2 from DCS-2332L-CAM1
i=live2.sdp with v2.0
t=0 0
a=type:broadcast
a=control:*
a=range:npt=0-
a=x-qt-text-nam:RTSP/RTP stream 2 from DCS-2332L-CAM1
a=x-qt-text-inf:live2.sdp
m=video 0 RTP/AVP 26
c=IN IP4 0.0.0.0
b=AS:1500
a=x-dimensions:1280,800
a=control:track1
m=audio 0 RTP/AVP 0
c=IN IP4 0.0.0.0
b=AS:64
a=control:track2

Created receiver for "video/JPEG" subsession (client ports 34692-34693)
Ignoring "audio/PCMU" subsession, because we've asked to receive a single video session only
Sending request: SETUP rtsp://192.168.11.31/live2.sdp/track1 RTSP/1.0
CSeq: 4
User-Agent: openRTSP (LIVE555 Streaming Media v2011.12.23)
Transport: RTP/AVP/TCP;unicast;interleaved=0-1

Received 179 new bytes of response data.
Received a complete SETUP response:
RTSP/1.0 200 OK
CSeq: 4
Date: Thu, May 22 2014 09:56:44 GMT
Transport: RTP/AVP/TCP;unicast;destination=192.168.11.20;source=192.168.11.31;interleaved=0-1
Session: 6B8B4567

Setup "video/JPEG" subsession (client ports 34692-34693)
Outputting data from the "video/JPEG" subsession to 'stdout'
Sending request: PLAY rtsp://192.168.11.31/live2.sdp/ RTSP/1.0
CSeq: 5
User-Agent: openRTSP (LIVE555 Streaming Media v2011.12.23)
Session: 6B8B4567
Range: npt=0.000-

Received a complete PLAY response:
RTSP/1.0 200 OK

```
CSeq: 5
Date: Thu, May 22 2014 09:56:44 GMT
Range: npt=0.000-
Session: 6B8B4567
RTP-Info: url=rtsp://192.168.11.31/live2.sdp/track1;seq=57466;rtptime=
0,url=rtsp://192.168.11.31/live2.sdp/track2;seq=0;rtptime=0


Started playing session
Receiving streamed data (signal with "kill -HUP 31603" or "kill -USR1 31603" to terminate)...
Got shutdown signal
begin_QOS_statistics
subsession    video/JPEG
num_packets_received 15988
num_packets_lost    2
elapsed_measurement_time      33.000209
kBytes_received_total  22824.376000
measurement_sampling_interval_ms 1000
kbits_per_second_min 1081.594110
kbits_per_second_ave 5533.147017
kbits_per_second_max 10433.606571
packet_loss_percentage_min    0.000000
packet_loss_percentage_ave    0.012508
packet_loss_percentage_max    0.502513
inter_packet_gap_ms_min 0.009000
inter_packet_gap_ms_ave 2.027994
inter_packet_gap_ms_max 1292.886000
end_QOS_statistics
Sending request: TEARDOWN rtsp://192.168.11.31/live2.sdp/ RTSP/1.0
CSeq: 6
User-Agent: openRTSP (LIVE555 Streaming Media v2011.12.23)
Session: 6B8B4567
```

QoS Stats RTSP H.264

Thursday, May 22, 2014 4:10 AM

ristopher@seniorDesignUbuntuLTS:~/live555$ openRTSP -v -t -Q rtsp://192.168.11.31:554/live1.sdp > /dev/null
Opening connection to 192.168.11.31, port 554...
...remote connection opened
Sending request: OPTIONS rtsp://192.168.11.31:554/live1.sdp RTSP/1.0
CSeq: 2
User-Agent: openRTSP (LIVE555 Streaming Media v2011.12.23)

Received 152 new bytes of response data.
Received a complete OPTIONS response:
RTSP/1.0 200 OK
CSeq: 2
Date: Thu, May 22 2014 11:06:23 GMT
Public: OPTIONS, DESCRIBE, SETUP, TEARDOWN, PLAY, PAUSE, GET_PARAMETER, SET_PARAMETER

Sending request: DESCRIBE rtsp://192.168.11.31:554/live1.sdp RTSP/1.0
CSeq: 3
User-Agent: openRTSP (LIVE555 Streaming Media v2011.12.23)
Accept: application/sdp

Received 816 new bytes of response data.
Received a complete DESCRIBE response:
RTSP/1.0 200 OK
CSeq: 3
Date: Thu, May 22 2014 11:06:23 GMT
Content-Base: rtsp://192.168.11.31/live1.sdp/
Content-Type: application/sdp
Content-Length: 652

v=0
o=- 1400752572595686 1 IN IP4 192.168.11.31
s=RTSP/RTP stream 1 from DCS-2332L-CAM1
i=live1.sdp with v2.0
t=0 0
a=type:broadcast
a=control:*
a=range:npt=0-
a=x-qt-text-nam:RTSP/RTP stream 1 from DCS-2332L-CAM1
a=x-qt-text-inf:live1.sdp
m=video 0 RTP/AVP 96
c=IN IP4 0.0.0.0
b=AS:1500
a=rtpmap:96 H264/90000
a=fmtp:96 packetization-mode=1;profile-level-id=640028;sprop-parameter-sets=Z2QAKK2EBUViuKxUdCAqKxXFYqOhAVFYrisVHQgKisVxWKjoQFRWK4rFR0ICorFcVio6ECSFITk8nyfk/k

/J8nm5s00IEkKQnJ5Pk/J/J+T5PNzZprQCgDLSpAAAAwHgAAA4QYEAABfXhAAAvrwr3vheEQjU,aO48sA==
a=control:track1
m=audio 0 RTP/AVP 0
c=IN IP4 0.0.0.0
b=AS:64
a=control:track2

Opened URL "rtsp://192.168.11.31:554/live1.sdp", returning a SDP description:
v=0
o=- 1400752572595686 1 IN IP4 192.168.11.31
s=RTSP/RTP stream 1 from DCS-2332L-CAM1
i=live1.sdp with v2.0
t=0 0
a=type:broadcast
a=control:*
a=range:npt=0-
a=x-qt-text-nam:RTSP/RTP stream 1 from DCS-2332L-CAM1
a=x-qt-text-inf:live1.sdp
m=video 0 RTP/AVP 96
c=IN IP4 0.0.0.0
b=AS:1500
a=rtpmap:96 H264/90000
a=fmtp:96 packetization-mode=1;profile-level-id=640028;sprop-parameter-
sets=Z2QAKK2EBUViuKxUdCAqKxXFYqOhAVFYrisVHQgKisVxWKjoQFRWK4rFR0ICorFcVio6ECSFITk8nyfk/k
/J8nm5s00IEkKQnJ5Pk/J/J+T5PNzZprQCgDLSpAAAAwHgAAA4QYEAABfXhAAAvrwr3vheEQjU,aO48sA==
a=control:track1
m=audio 0 RTP/AVP 0
c=IN IP4 0.0.0.0
b=AS:64
a=control:track2

Created receiver for "video/H264" subsession (client ports 54318-54319)
Ignoring "audio/PCMU" subsession, because we've asked to receive a single video session only
Sending request: SETUP rtsp://192.168.11.31/live1.sdp/track1 RTSP/1.0
CSeq: 4
User-Agent: openRTSP (LIVE555 Streaming Media v2011.12.23)
Transport: RTP/AVP/TCP;unicast;interleaved=0-1

Received 179 new bytes of response data.
Received a complete SETUP response:
RTSP/1.0 200 OK
CSeq: 4
Date: Thu, May 22 2014 11:06:23 GMT
Transport: RTP/AVP/TCP;unicast;destination=192.168.11.20;source=192.168.11.31;interleaved=0-1
Session: 0F18C5D3

Setup "video/H264" subsession (client ports 54318-54319)
Outputting data from the "video/H264" subsession to 'stdout'
Sending request: PLAY rtsp://192.168.11.31/live1.sdp/ RTSP/1.0
CSeq: 5
User-Agent: openRTSP (LIVE555 Streaming Media v2011.12.23)

```
Session: 0F18C5D3
Range: npt=0.000-


Received a complete PLAY response:
RTSP/1.0 200 OK
CSeq: 5
Date: Thu, May 22 2014 11:06:23 GMT
Range: npt=0.000-
Session: 0F18C5D3
RTP-Info: url=rtsp://192.168.11.31/live1.sdp/track1;seq=57896;rtptime=
0,url=rtsp://192.168.11.31/live1.sdp/track2;seq=0;rtptime=0


Started playing session
Receiving streamed data (signal with "kill -HUP 32050" or "kill -USR1 32050" to terminate)...
Got shutdown signal
begin_QOS_statistics
subsession    video/H264
num_packets_received 16216
num_packets_lost    1
elapsed_measurement_time      54.670178
kBytes_received_total  22363.859000
measurement_sampling_interval_ms 1000
kbits_per_second_min 470.622294
kbits_per_second_ave 3272.549652
kbits_per_second_max 34736.226995
packet_loss_percentage_min    0.000000
packet_loss_percentage_ave    0.006166
packet_loss_percentage_max    0.473934
inter_packet_gap_ms_min 0.008000
inter_packet_gap_ms_ave 3.292083
inter_packet_gap_ms_max 1791.568000
end_QOS_statistics
Sending request: TEARDOWN rtsp://192.168.11.31/live1.sdp/ RTSP/1.0
CSeq: 6
User-Agent: openRTSP (LIVE555 Streaming Media v2011.12.23)
Session: 0F18C5D3


^C
christopher@seniorDesignUbuntuLTS:~/live555$ openRTSP -v -t -Q rtsp://192.168.11.31:554/live1.sdp >
/dev/null
Opening connection to 192.168.11.31, port 554...
...remote connection opened
Sending request: OPTIONS rtsp://192.168.11.31:554/live1.sdp RTSP/1.0
CSeq: 2
User-Agent: openRTSP (LIVE555 Streaming Media v2011.12.23)


Received 152 new bytes of response data.
Received a complete OPTIONS response:
RTSP/1.0 200 OK
```

CSeq: 2
Date: Thu, May 22 2014 11:08:12 GMT
Public: OPTIONS, DESCRIBE, SETUP, TEARDOWN, PLAY, PAUSE, GET_PARAMETER, SET_PARAMETER

Sending request: DESCRIBE rtsp://192.168.11.31:554/live1.sdp RTSP/1.0
CSeq: 3
User-Agent: openRTSP (LIVE555 Streaming Media v2011.12.23)
Accept: application/sdp

Received 816 new bytes of response data.
Received a complete DESCRIBE response:
RTSP/1.0 200 OK
CSeq: 3
Date: Thu, May 22 2014 11:08:12 GMT
Content-Base: rtsp://192.168.11.31/live1.sdp/
Content-Type: application/sdp
Content-Length: 652

v=0
o=- 1400752572595686 1 IN IP4 192.168.11.31
s=RTSP/RTP stream 1 from DCS-2332L-CAM1
i=live1.sdp with v2.0
t=0 0
a=type:broadcast
a=control:*
a=range:npt=0-
a=x-qt-text-nam:RTSP/RTP stream 1 from DCS-2332L-CAM1
a=x-qt-text-inf:live1.sdp
m=video 0 RTP/AVP 96
c=IN IP4 0.0.0.0
b=AS:1500
a=rtpmap:96 H264/90000
a=fmtp:96 packetization-mode=1;profile-level-id=640028;sprop-parameter-
sets=Z2QAKK2EBUViuKxUdCAqKxXFYqOhAVFYrisVHQgKisVxWKjoQFRWK4rFR0ICorFcVio6ECSFITk8nyfk/k
/J8nm5s00IEkKQnJ5Pk/J/J+T5PNzZprQCgDLSpAAAAwHgAAA4QYEAABfXhAAAvrwr3vheEQjU,aO48sA==
a=control:track1
m=audio 0 RTP/AVP 0
c=IN IP4 0.0.0.0
b=AS:64
a=control:track2

Opened URL "rtsp://192.168.11.31:554/live1.sdp", returning a SDP description:
v=0
o=- 1400752572595686 1 IN IP4 192.168.11.31
s=RTSP/RTP stream 1 from DCS-2332L-CAM1
i=live1.sdp with v2.0
t=0 0
a=type:broadcast
a=control:*
a=range:npt=0-
a=x-qt-text-nam:RTSP/RTP stream 1 from DCS-2332L-CAM1

a=x-qt-text-inf:live1.sdp
m=video 0 RTP/AVP 96
c=IN IP4 0.0.0.0
b=AS:1500
a=rtpmap:96 H264/90000
a=fmtp:96 packetization-mode=1;profile-level-id=640028;sprop-parameter-sets=Z2QAKK2EBUViuKxUdCAqKxXFYqOhAVFYrisVHQgKisVxWKjoQFRWK4rFR0ICorFcVio6ECSFITk8nyfk/k/J8nm5s00IEkKQnJ5Pk/J/J+T5PNzZprQCgDLSpAAAAwHgAAA4QYEAABfXhAAAvrwr3vheEQjU,aO48sA==
a=control:track1
m=audio 0 RTP/AVP 0
c=IN IP4 0.0.0.0
b=AS:64
a=control:track2

Created receiver for "video/H264" subsession (client ports 54608-54609)
Ignoring "audio/PCMU" subsession, because we've asked to receive a single video session only
Sending request: SETUP rtsp://192.168.11.31/live1.sdp/track1 RTSP/1.0
CSeq: 4
User-Agent: openRTSP (LIVE555 Streaming Media v2011.12.23)
Transport: RTP/AVP/TCP;unicast;interleaved=0-1

Received 179 new bytes of response data.
Received a complete SETUP response:
RTSP/1.0 200 OK
CSeq: 4
Date: Thu, May 22 2014 11:08:12 GMT
Transport: RTP/AVP/TCP;unicast;destination=192.168.11.20;source=192.168.11.31;interleaved=0-1
Session: 5524F612

Setup "video/H264" subsession (client ports 54608-54609)
Outputting data from the "video/H264" subsession to 'stdout'
Sending request: PLAY rtsp://192.168.11.31/live1.sdp/ RTSP/1.0
CSeq: 5
User-Agent: openRTSP (LIVE555 Streaming Media v2011.12.23)
Session: 5524F612
Range: npt=0.000-

Received a complete PLAY response:
RTSP/1.0 200 OK
CSeq: 5
Date: Thu, May 22 2014 11:08:12 GMT
Range: npt=0.000-
Session: 5524F612
RTP-Info: url=rtsp://192.168.11.31/live1.sdp/track1;seq=24603;rtptime=0,url=rtsp://192.168.11.31/live1.sdp/track2;seq=0;rtptime=0

Started playing session
Receiving streamed data (signal with "kill -HUP 32059" or "kill -USR1 32059" to terminate)...
Got shutdown signal

begin_QOS_statistics
subsession video/H264
num_packets_received 9355
num_packets_lost 6
elapsed_measurement_time 32.000355
kBytes_received_total 12889.925000
measurement_sampling_interval_ms 1000
kbits_per_second_min 0.000000
kbits_per_second_ave 3222.445501
kbits_per_second_max 12897.314977
packet_loss_percentage_min 0.000000
packet_loss_percentage_ave 0.064096
packet_loss_percentage_max 0.875274
inter_packet_gap_ms_min 0.009000
inter_packet_gap_ms_ave 3.362048
inter_packet_gap_ms_max 1689.810000
end_QOS_statistics
Sending request: TEARDOWN rtsp://192.168.11.31/live1.sdp/ RTSP/1.0
CSeq: 6
User-Agent: openRTSP (LIVE555 Streaming Media v2011.12.23)
Session: 5524F612