VIRTUAL PAINT ON DE2 FPGA BOARD THROUGH GESTURE DETECTION AND COLOR SEGMENTATION TECHNIQUE

MOHD SALLEHUDIN BIN SAAD

A thesis submitted in
fulfillment of the requirement for the award of the
Degree of Master of Electrical Engineering

Faculty of Electrical and Electronic Engineering
Universiti Tun Hussein Onn Malaysia

JANUARY 2014

ABSTRACT

Object recognition and tracking with real-time smart camera plays an important role in many modern vision applications such as work force tracking, intelligence surveillance, fire detection and many more. This research will be present a FPGA-based digital control for a Virtual Paint. Virtual Paint project use DE2 board, 5 megapixel camera and a VGA Monitor, which is uses the camera to capture hand movements, so you can hand in the air painting and displayed on a LCD display, the whole process without using mouse or joystick like device. The main motivation of this project is to extract the gesture detection and color segmentation technique from CMOS camera sensor to perform virtual paint. The control algorithm will be using a verilog language based on the use of logical state diagram.

ABSTRAK

Pengiktirafan objek dan pengesanan dengan masa nyata kamera pintar memainkan peranan penting dalam banyak aplikasi visi moden seperti pengesanan tenaga kerja, pengawasan kecerdasan, pengesan kebakaran dan banyak lagi. Kajian ini berdasarkan kawalan digital berasaskan FPGA untuk cat maya. Projek cat maya menggunakan papan DE2, kamera 5 megapiksel dan paparan VGA, yang menggunakan kamera untuk menangkap pergerakan tangan, jadi anda boleh menyerahkan lukisan udara dan dipaparkan pada paparan LCD, seluruh proses tanpa menggunakan tetikus atau kayu bedik seperti peranti. Motivasi utama projek ini adalah untuk mengeluarkan pengesanan isyarat dan teknik ambang warna dari sensor kamera CMOS untuk melaksanakan cat maya. Algoritma kawalan akan menggunakan bahasa verilog berdasarkan penggunaan gambarajah keadaan logik.

CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS AND ABBREVIATIONS

ASIC - Application-Specific Integrated Circuit
CCD - Charge-Coupled Device
CFA - Color Filter Array
FIFO - First-In First-Out
FPGA - Field-Programmable Gate Array
FPS - Frames Per Second
HAL - Hardware Abstraction Layer
HDL - Hardware Descriptive Language
IP - Intellectual Property
LE - Logic Element
MMU - Memory Management Unit
PLL - Phase-Locked Loop
RISC - Reduced Instruction-Set Computing
SDRAM - Synchronous Dynamic Random Access Memory
SRAM - Static Random Access Memory
VGA - Video Graphics Array

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

This chapter presents the introduction of the thesis, including with a short overview of virtual paint systems. Furthermore, it details the aims of the project, continuing with the objective as well as the scope of the project and finishing with the outline of the thesis.

1.1 Introduction

Virtual paint is a system which can be used to draw colored pictures on any plain surfaces like a wall using hand movements and fingers with colored bands. This project draws its motivation from the sixth sense where using different gesture detection and color segmentation techniques while a software running on a handheld computing device performs a variety of different tasks like making notes in the air, reading a newspaper with projected videos, capturing photographs using hand actions and even painting on walls. Intrigued by the scope of the project, we decided to do something similar in hardware. So, within the constraints of availability of hardware and time we implemented Virtual Paint. Figure 1.1 shows a virtual paint illustration.

Figure 1.1: Virtual paint illustration.

1.2 Objective of the Project

The intention of this research project is to build a system of Virtual Paint using Verilog language. It is summarized in two points as stated below:

1. To develop an embedded system hardware and software for virtual paint system by using verilog code on DE2 FPGA board.
2. To apply color segmentation and gesture detection technique into embedded system of virtual paint.

1.3 Problem statement

Object recognition and tracking with real-time smart camera plays an important role in many modern vision applications such as work force tracking, intelligence surveillance, fire detection and many more. This research will be present a FPGA-based digital control for a Virtual Paint. Virtual Paint project use DE2 board, 5 megapixel

camera and a VGA monitor, which is uses the camera to capture hand movements, so you can hand in the air painting and displayed on a monitor, the whole process without using mouse or joystick like device. The main motivation of this project is to extract the gesture detection and color segmentation technique from CMOS camera sensor to perform virtual paint. The control algorithm will be using a Verilog language based on the use of logical state diagram.

1.4 Scope of project

This project will be implemented by using Altera DE2 FPGA board. The language used in the hardware design is Verilog coding. Secondly, this project is to develop a system that can paint directly to VGA monitor by using image segmentation technique and gesture detection based on red and yellow color detection on CMOS camera sensor.

1.5 Thesis Outline

This report is arranged and distributed into five chapters. Chapter 1 has presents a brief introduction of the project mainly about virtual paint system, the problem statements, the objectives of the project and its scope, and the limitations identified using the proposed approach.

Chapter 2 includes literature survey related to this project as per referred to previous studies and results obtained by past researchers. It also contains some important findings from past researchers such as a review of existing image segmentation and gesture detection methods. Their respective advantages and disadvantages, with specific reference to my research are discussed.

Chapter 3 provides methodology in how this project is conducted in sequence. It also includes the system architecture, block diagram and main description of virtual paint system.

Chapter 4 contains the results and findings of the project. A simulation results run on main program using Verilog coding in Quartus 8.1 have been analyzed and studied properly in this chapter.

Lastly is chapter 5 where this chapter concludes the dissertation. It presents a summary of research achievements together with a discussion of their significance. Some recommended future work also presented in this chapter.

CHAPTER 2

LITERATURE REVIEW

M. Petouris, A. Kalantzopoulos and E. Zigouris [1] entitled “An Fpga-Based Digital Camera System Controlled from an LCD Touch Panel” proposed presents the design and implementation of such an open FPGA based on Digital Camera System for image capturing and real-time image processing. Images captured with a CMOS sensor are initially stored in the system's memory and then they are displayed on an LCD Touch Panel. The main goal of this proposed architecture is to be used as a platform to implement and test advanced image processing algorithms. Apart of this, the system supports the control of the image sensor, through the LCD Touch Panel. In addition, has the ability to communicate with a PC through a JTAG interface for storing the images on it. The proposed FPGA-based Digital Camera System, due to the FPGA flexibility, is mainly targeted to be used as an open and low cost platform for implementing and testing real-time image processing algorithms. In addition the exploitation of LCD Touch Panel can effectively assist in the control of more cameras’ parameters. Image processing algorithms can take place before or after the data storing and because of the FPGA' s presence, system has the ability to be easily modified. Future plans are to embed and test more advance image processing algorithms due to the fact that there is enough space left in the FPGA. In addition, we intend to create an extended menu for the LCD touch panel. Developing such a menu the user can fully and in a friendly manner

control camera's functionality. Through this menu the user can also easily select the execution of the desirable image processing algorithm.

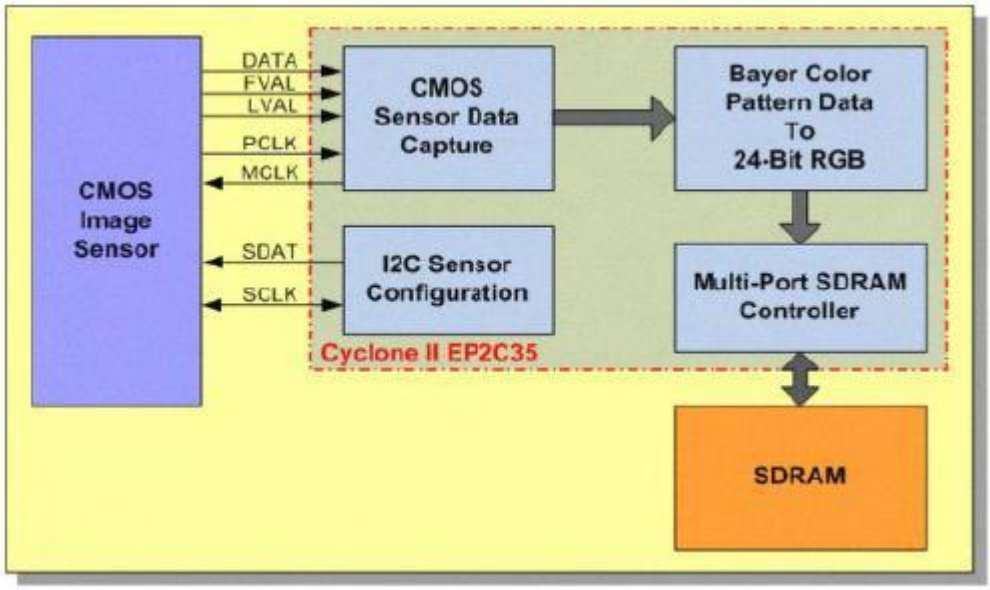

Figure 2.1: Block Diagram of camera sub system

Meanwhile, Suh Ho Lee, Seon Wook Kim, and Suki Kim [2] entitled "Implementation of a Low Power Motion Detection Camera Processor Using a Cmos Image Sensor" proposed presents a low power motion detection camera processor, called Bluebox. It includes an ARM embedded microprocessor and several hardwired modules using pseudo Advanced Microprocessor Bus Architecture (AMBA). For reducing power consumption, we use simple, but very efficient software and hardware techniques. We propose a new motion detection algorithm to use only one bit per pixel without loss of accuracy, and therefore it needs very small computing power and satisfies real-time processing requirements. And we design a power management scheme to control system clocks for power reduction. Our proposed architecture is implemented and verified in an FPGA with a CIS (CMOS Image Sensor). It was shown that replacing the arithmetic calculations with a binary Boolean XOR function without loss of accuracy could significantly reduce the computational complexity of the previous algorithm, low cost, and low power consumption.

In addition to the algorithm a power manages the scheme was proposed for further power reduction, and about 33% in our test evaluation. The unused blocks of a full system are properly put in idle, sleep and stop modes. A system for real time motion detection has been implemented and verifies in an FPGA video surveillance system.

Furthermore, Mohamed Nasir Bin Mohamed Shukor, Lo Hai Hiung and Patrick Sebastian [3] entitled “Implementation of Colour Filtering on Fpga” proposed presents the construction of a real time hardware image processing system on Field Programmable Gate Array (FPGA). The chosen image processing algorithm is a single colour Filtering algorithm. This work utilizes Altera DE2 development board empowered by the Cyclone II FPGA paired with a 1.3 Mega pixel CMOS camera from Terasic Technologies. Verilog HDL is chosen as the hardware programming language for this system and its compiled using Quartus II program. The functionality of the algorithm is first verified in Matlab, simulating the expected output of the system before implementing it onto the FPGA development board. colour Filtering algorithms are successfully implemented on Cyclone II FPGA. The double band-pass filter algorithm is found to be more effective to capture a wider spectrum of blue colour compared to that of a single band-pass filter algorithm. Currently work is done to quantify a performance metric for the system before implementing and testing the triple and quadruple band-pass filter algorithm to determine if there were further improvement could be achieved.

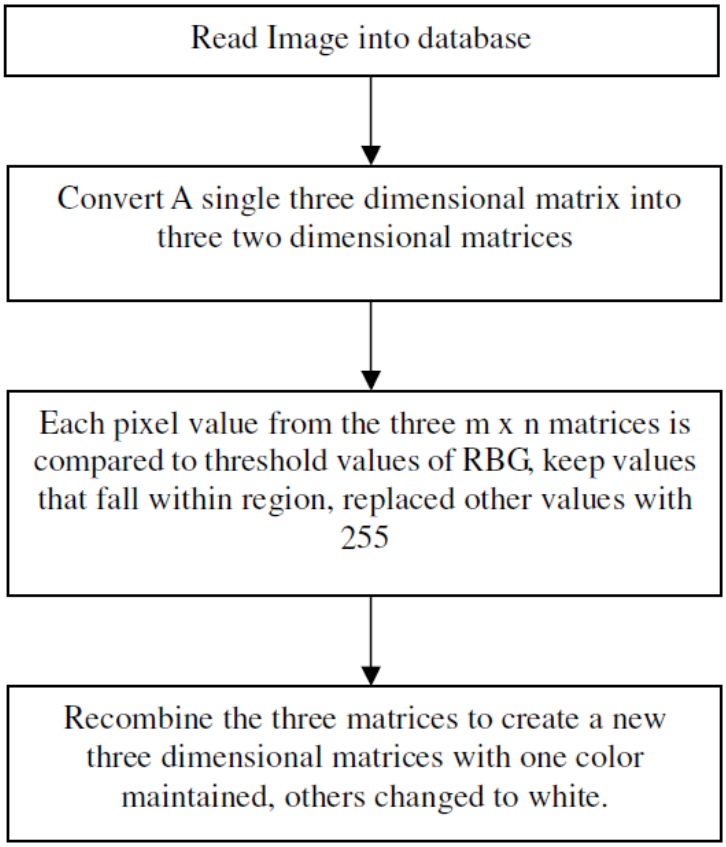

Figure 2.2: The flow of colour filtering algorithm

Lastly, Nai-Jian Wang, Sheng-Chieh Chang and Pei-Jung Chou [4] entitled "A Real-Time Multi-Face Detection System Implemented On FPGA" proposed a real-time multi-face detection system based on hardware design to enhance the processing time. The proposed hardware architecture is implemented on Altera DE2-70 development board to test the feasibility of our hardware design. The implementation of our system requires 15,223 logic elements. It can operate in real-time at a frame rate of 30fps, and detect up to five faces simultaneously. The experimental result shows that our proposed face detection architecture attains a real-time reliable system with low cost and high detection rate. The hardware system could be easily implemented and it can reduce not only processing time but the hardware resources. For 320x240 resolution, the system use NTSC format input to achieve 30 execution efficiency. Table 2.1 shows a literature review of selected paper discuss based on methods and the advantages of the projects.

Table 2.1: Literature review

No.	Researcher	Title	Method /Descriptions	Advantages
1	M. Petouris, A. Kalantzopoulos and E. Zigouris.	An FPGA Based Digital Camera System Controlled From An LCD Touch Panel.	This paper presents the design and implementation of such an open FPGA based on Digital Camera System for image capturing and real-time image processing. Images captured with a CMOS sensor are initially stored in the system's memory and then they are displayed on an LCD Touch Panel.	The proposed FPGA-based Digital Camera System, due to the FPGA flexibility, is mainly targeted to be used as an open and low cost platform for implementing and testing real-time image processing algorithms. In addition the exploitation of LCD touch panel can effectively assist in the control of more cameras' parameters.
2	Suh Ho Lee, Seon Wook Kim, and Suki Kim.	Implementation Of A Low Power Motion Detection Camera Processor Using a Cmos Image Sensor.	This paper presents a low power motion detection camera processor, called Bluebox. It includes an ARM embedded microprocessor and several hardwired modules using pseudo Advanced Microprocessor Bus Architecture (AMBA).	It was shown that replacing the arithmetic calculations with a binary Boolean XOR function without loss of accuracy could significantly reduce the computational complexity of the previous algorithm, low cost, and low power consumption. A system for real time motion detection has been implemented and verifies in an FPGA video surveillance system.

3	Mohamed Nasir Bin Mohamed Shukor, Lo Hai Hiung, Patrick Sebastian.	Implementation of Color Filtering on FPGA.	This paper is to construct a real time hardware image processing system on Field Programmable Gate Array (FPGA). The chosen image processing algorithm is a single color filtering algorithm. This work utilizes Altera DE2 development board empowered by the Cyclone II FPGA paired with a 1.3 Mega pixel CMOS camera sensor.	Color filtering algorithms are successfully implemented on Cyclone II FPGA. The double band-pass filter algorithm is found to be more effective to capture a wider spectrum of blue color compared to that of a single band-pass filter algorithm.
4	Nai-Jian Wang, Sheng-Chieh Chang, Pei-Jung Chou.	A Real-Time Multi-Face Detection System Implemented on FPGA.	This paper proposed a real-time multi face detection system based on hardware design to enhance the processing time. The proposed hardware architecture is implemented on Altera DE2-70 development board to test the feasibility of our hardware design.	The hardware system could be easily implemented and it can reduce not only processing time but the hardware resources. For 320x240 resolution, the system use NTSC format input to achieve 30 execution efficiency.

CHAPTER 3

METHODOLOGY

3.1 Introduction

Virtual paint is a similar wall can be used on any flat surface with the same movement of the hand and fingers to draw a color image system. Though it cannot use a mouse or joystick like any other control devices, as long as the application that people can move in the air, hands and fingers to paint.

3.2 Implementation plan on hardware device

The virtual Paint system is to be connected to the Altera DE2-70 board to complete the system design. The DE2 board will act as a microcontroller to control and communicate with the external component which is in this case, TRDB_D5M, 5 Mega Pixel camera and VGA Monitor. Figure 3.1 shows a hardware device implementation on the virtual paint system.

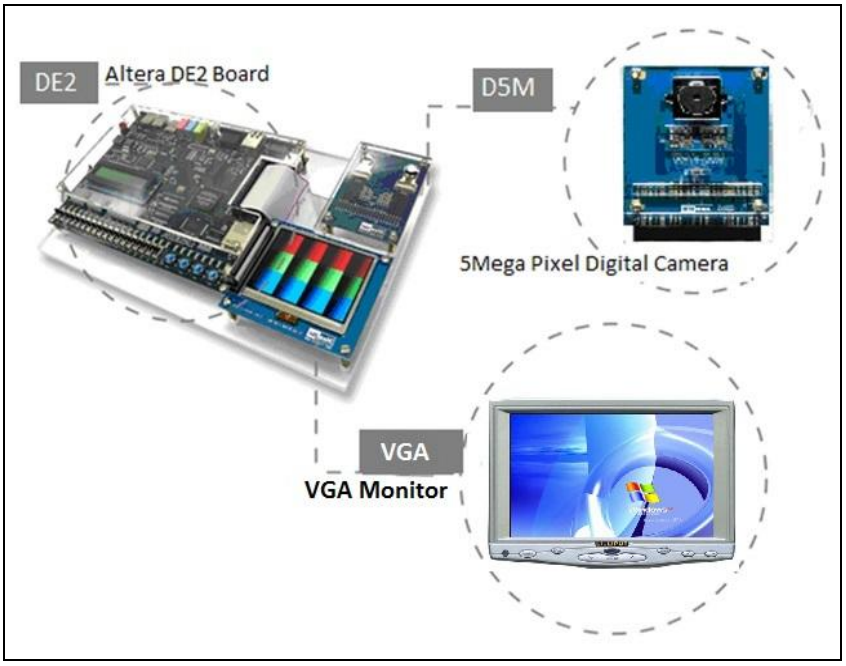

Figure 3.1: Hardware device implementation

3.3 Implementation plan on software

The software placed in the FPGA is implemented in Verilog. It provides module level simulation and verification, which is allows building the design up from smaller modules. Once the simulation, synthesis and time verification is verified in the Quartus software, then the programmer device tool will load the bit file of Verilog onto the FPGA using the USB blaster. The process of programming hardware language has several steps before being loaded onto the FPGA. Figure 3.2 below shows HDL programming process flow

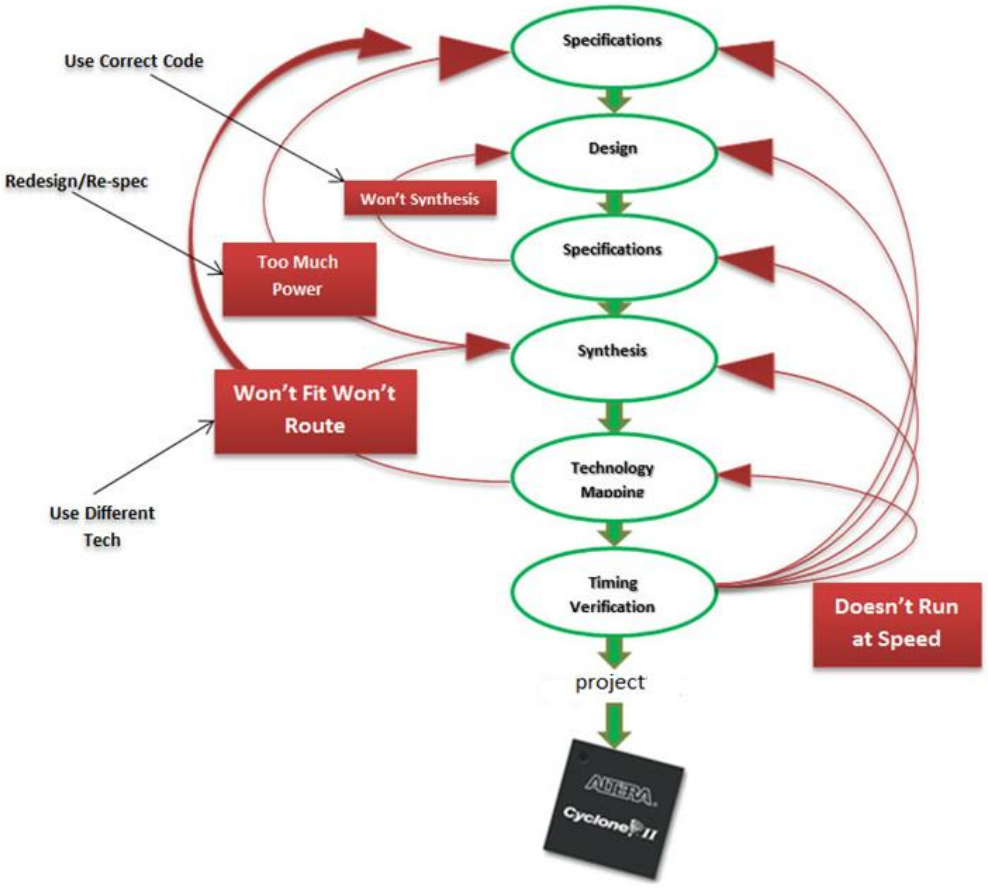

Figure 3.2: HDL programming process flow

3.4 System architecture

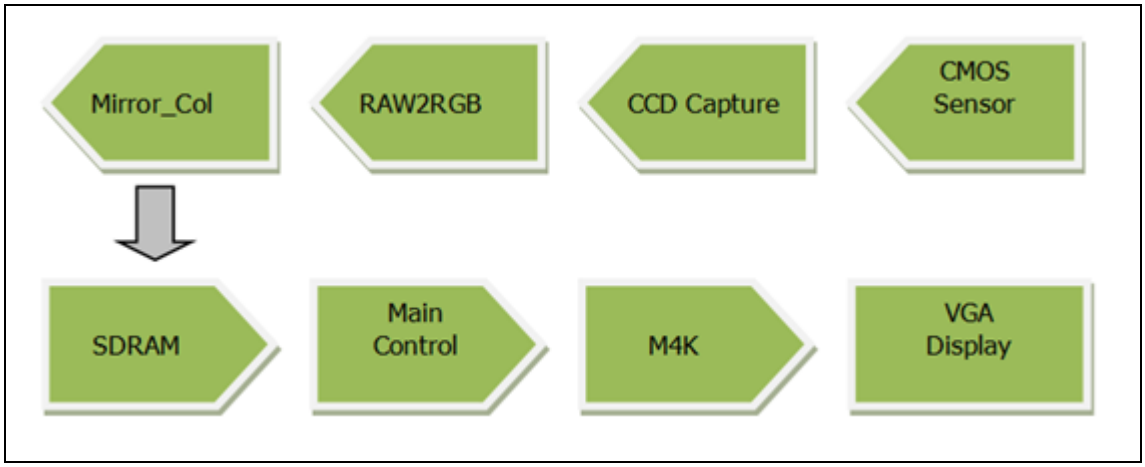

Figure 3.3: Virtual paint system architecture

3.5.1 Features of the module

1. **CCD_Capture:**

   Processed by the module coming from the CMOS sensor pixel data. Main function is to produce horizontal, vertical and frame counter to insert data from the camera to the GPIO1 in.

2. **RAW2RGB:**

   This module is used to convert the image format from the Bayer to RGB format and the red, green and blue values of the 10-bit.

3. **Mirror_Col:**

   As from the camera's image sensor to capture the left and right side is the reverse, mirror module will need to change it back, to match with the actual situation.

4. **SDRAM:**

   As a frame buffer to store each pixel value for all RGB values. An RGB total of 30, three colors of the 10, the SDRAM requires two lines to store data as a pixel address. The main central part, directly reads SDRAM, do the calculation.

5. **Main Ctrl:**

   This part is the most important, is responsible for the color segmentation and detection. Which in turn reads the SDRAM frame buffer pixel and the RGB values of each pixel and the relative intensity of a component-based detection of different colors. When we draw on the screen when it calculates the center of the color marker and selected color value is written M4K.

6. **M4K:**

   The memory address used to store the RGB values for each pixel, VGA controller M4K reads directly from the RGB values. When we are in a different value M4K drawing can choose up to three colors. Memory data width is 2 bits.

7. **VGA:**

   VGA controller and contains a VGA monitor. VGA controller is responsible for receiving from the memory of the host side and the RGB data for each pixel the RGB data are sent to a VGA DAC. In addition, VGA controller is also used in painting is displayed on the screen when the color palette to select different.

3.6 Main part of the description

This part is to calculate, draw lines, and averages modules. A module for drawing lines, we have for the detection of the center pixel color space draw lines, but also for smoothing the successive image frames in the two pixels between the calculated center pixel. For basic operation section, the most important is to realize how to detect the red, yellow and green. By comparing the RGB values and the threshold value, and calculates the scan and the RGB values of each pixel to achieve correlation. After detection of a specific color, according to the detected color of the specified value is stored in memory. Part of the decision by a line drawing stored address.

For M4K reading part is continuously read values and sent to the VGA. Yellow for painting and select the color, red is used to control pen up and pen down operation, so when the red is detected penup mode is triggered, stop drawing. In order to eliminate noise on the screen, we specify before and now stored in the pixel memory 20 should have a length addresses. M4K data stored values are two color choices. It shows us what is the color used for drawing which one, you can use the color selection. In summary, we use four colors of the drawing with white erased. The eraser size of 3 × 3 pixels square. Our red detection value to control whether the drawing. M4K memory in each value in

the address corresponding to the VGA display pixel address. When we read M4K, we according to the data read from the memory to output specific RGB color values. 01 output red, 10 green, 11 blue, 00 white. VGA_controller module reads these values and output them on the screen. The monitor output color bar by moving the cursor to the designated area, select the color throughput pen to paper, pen is red. As in the mouse, click and release.

3.7 System instructions for virtual paint

Users of this project on the surface with a space probe to paint hand movements. Here, we use two fingers of the user to control all functions of the application. Whenever two fingertips detected by the sensor, a cursor will be displayed on the screen to help users locate the moment he, where he was on the screen. This is just to control the on-screen action or activity, and to maintain this gesture does not occur when drawing or other activities. As long as users fold the little finger and cannot be detected, the cursor starts drawing on the screen. Therefore, this gesture will be used to draw on the screen. Users can easily switch between these two gestures to draw pictures or control the cursor. For the choice of colors or the eraser choice for users need to move the cursor to the desired color, from the left of the screen select the color palette. With the little finger gesture showed repeated mouse clicks, first fold the little finger and then expand it to select the color. And then continue with the selected color graphics on the screen. Eraser and color is also used to convert between similar manner.

3.8 Hardware Design

And associated hardware modules as shown in Figure 3.3:

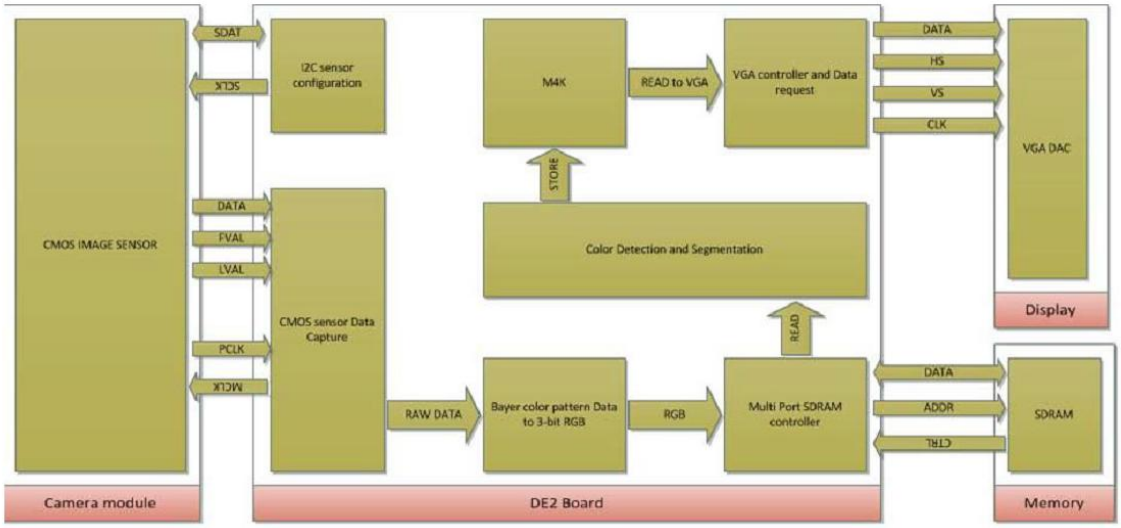

Figure 3.4: Virtual paint block diagram

3.8.1 Frame grabber

Frame grabbers includes a 1.5 megapixel CMOS image sensor and an IDE cable is used to connect to a port on the DE2 board. Camera for capturing an image, the image directly on the VGA display module.

3.8.2 Image processing and control unit

It is the central unit of the system in the DE2 development board used in this project the following ports: USB blaster for FPGA programming port for connecting to VGA port, as well as for the camera module and extensions head an IDE cable is connected between the GPIO1. Toggle switch is used to control the image sensor exposure settings, if you want to set new exposure, you need to reset the board. The data from the sensors are in an image sensor or CCD captures capture module and then is fed into RGB Bayer color pattern data conversion module. SDRAM memory via the sensor data stored in the

SDRAM controller. Main modules are color detection and segmentation module, SDRAM image buffer can be read from the pixel values for processing. M4K subsequently stored in the block corresponding to the color value of the pixel. VGA controller reads M4K memory location and the address stored in the corresponding color information is sent to the display unit to the VGA DAC. RAW2RGB module is used to convert an image from RGB Bayer pattern image data format.

Color detection and segmentation unit is the main control unit can be further divided into three parts, color detection and segmentation unit, average and central computing unit and the image color select units. For color detection and segmentation, we use the threshold comparison and every pixel RGB components of the relative intensity of the comparative analysis, choose red and yellow because these colors are high-intensity display, and white distinctly different so that we can filter out background noise. Average value calculation unit and the center, we use the counter value of the detected number of pixels of color calculation, and summarizes the X and Y coordinates, and then use these coordinates calculating the average of the pixel centers X and Y center. Color selection and graphics unit: The use of a palette displayed on the screen, modifying the VGA controller in order to leave a specific part of the screen and display a specific color. This is accomplished by attaching a color flow VGA input multiplexer, as long as the VGA pixel address falls within a certain range, to display a specific value.

3.8.3 Memory unit

Transmitted by CMOS sensor image data is stored in a frame buffer in SDRAM 8MB. We use the on-chip M4K stores color information corresponding to the frame are divided to produce the desired color of the pixel address.

3.8.4 Display unit

Using DE2 development board provides a standard VGA port for connecting a liquid crystal display unit and the projector. Not available at the appropriate time as a portable projector, we show that the concept of virtual painting paint on any ordinary classroom walls and over the head with a color projection image surface.

CHAPTER 4

RESULTS AND DISCUSSION

4.1 INTRODUCTION

In order to fulfill the overall system for virtual paint, simulations by using the Verilog language in Quartus 8.1 are executed. Back to the objective that the simulations are performed to completely detect gesture movement for cursor location and image segmentation for painting part.

4.2 RESULTS

The project has been created based on Verilog language which is run using the Quartus 8.1 software. Collected information provided on the website and research paper about procedures and software operation is carefully analyzed. Based on collecting information, the main challenge is to set up cursor movement that not response to the specified target color. With a different interface for 1.5 megapixel camera properties, reallocation of the pin assignment for 1.5 megapixel camera have been made. Figure 4.1 show full compilation process successfully done.

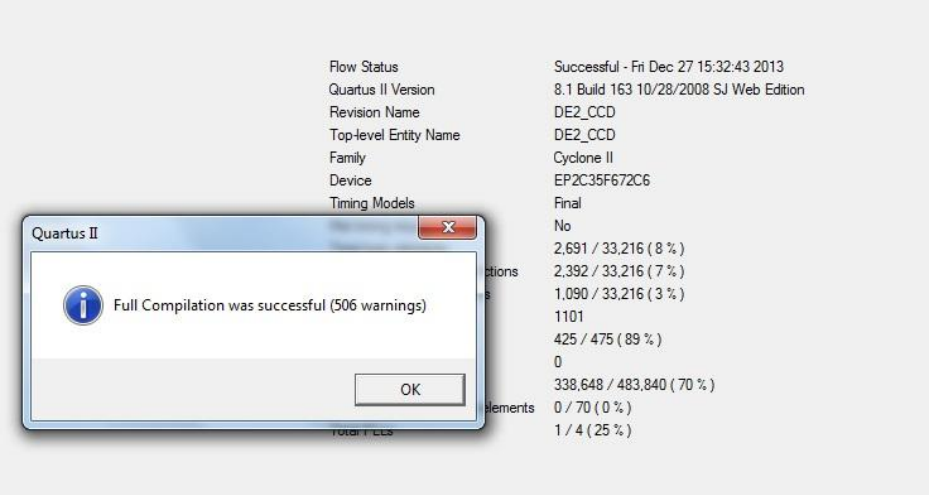

Figure 4.1: Full compilation success

Even successfully compiled and downloaded to the board, monitor running normally, but the camera still does not work. Figure 4.2 show a result, in the canvas area for painting.

Figure 4.2: Camera not working

Replacement of the CCD control file, CCD_Capture.v, I2C_CCD_Config.v, I2C_Controller.v and Reset_Delay.v in submodule files has been made and CCD camera module still does not work. Recompile the compiler get an error because the main function module refers to original I2C_Controller.v to define a function. This function is used to control the exposure time, so references to the function iExposure have been removed. Figure 4.3 shows coding for iExposure function was disable.

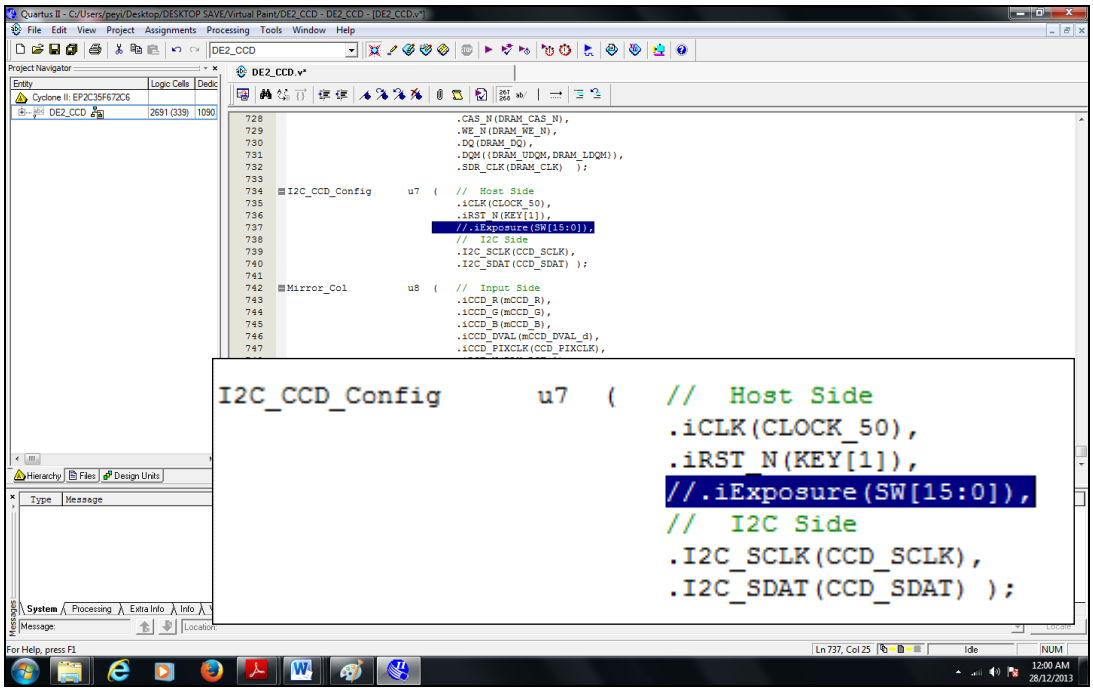

Figure 4.3: Remove iExposure function

Next compilation procedures were successfully done. Downloaded to the board running well, but still no success for camera function. Change on pin assignment for 1.5 megapixel camera pin out have been made and the compilation is successful but camera cannot correctly read images but found out that the rate of cursor movement and frame rates become better.

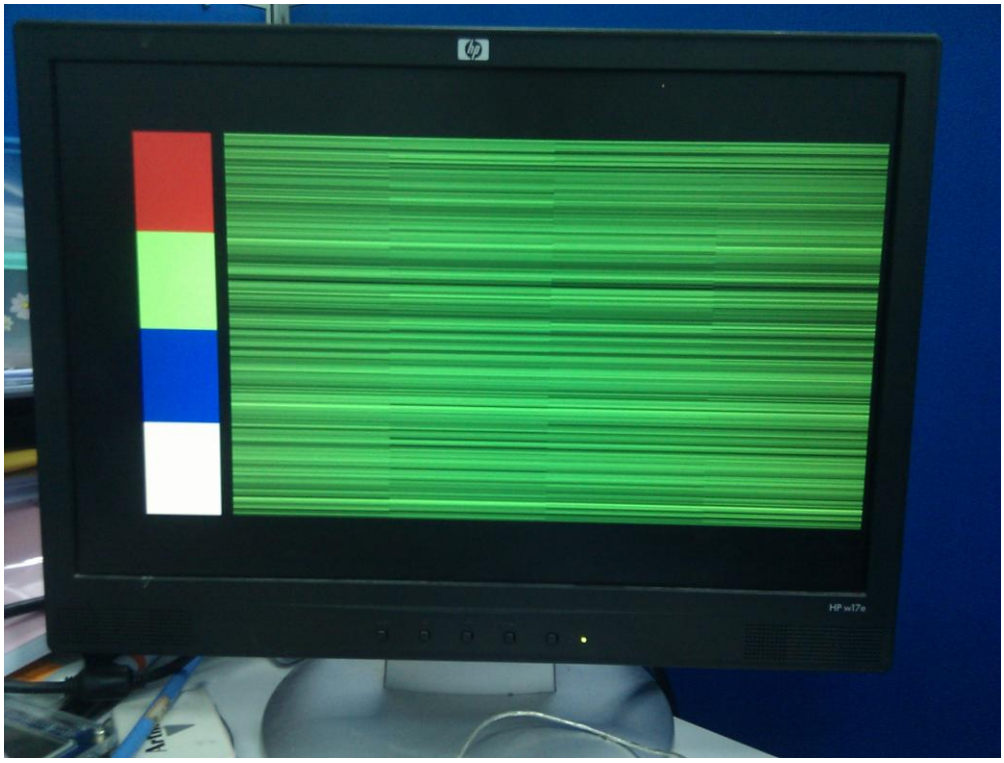

Figure 4.4: Raw image detection

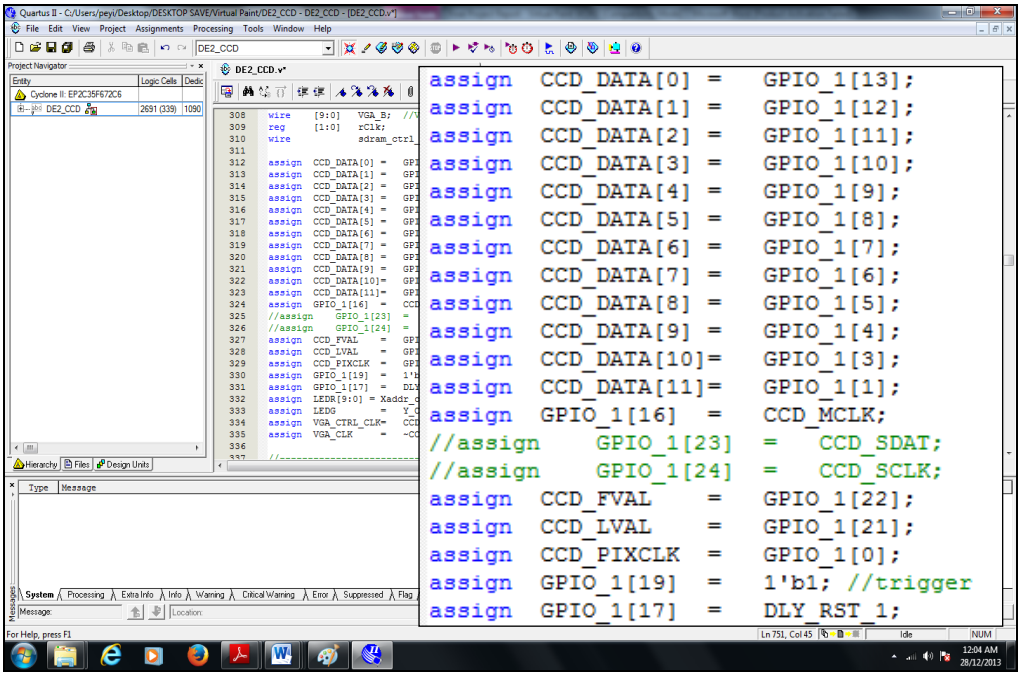

Figure 4.5: Pin assignment for camera TRDB D5M

CCD_DATA bit wide for 1.5 megapixel camera is 12 bits. Modifications and compilation successfully done. Download and run program, camera still cannot correctly read images, but the images rich in color than before. It explains webcam working now because if any movements detect over the lens, there will be on the screen moving images. Figure 4.6 shows an image with hidden camera which image becomes dark.

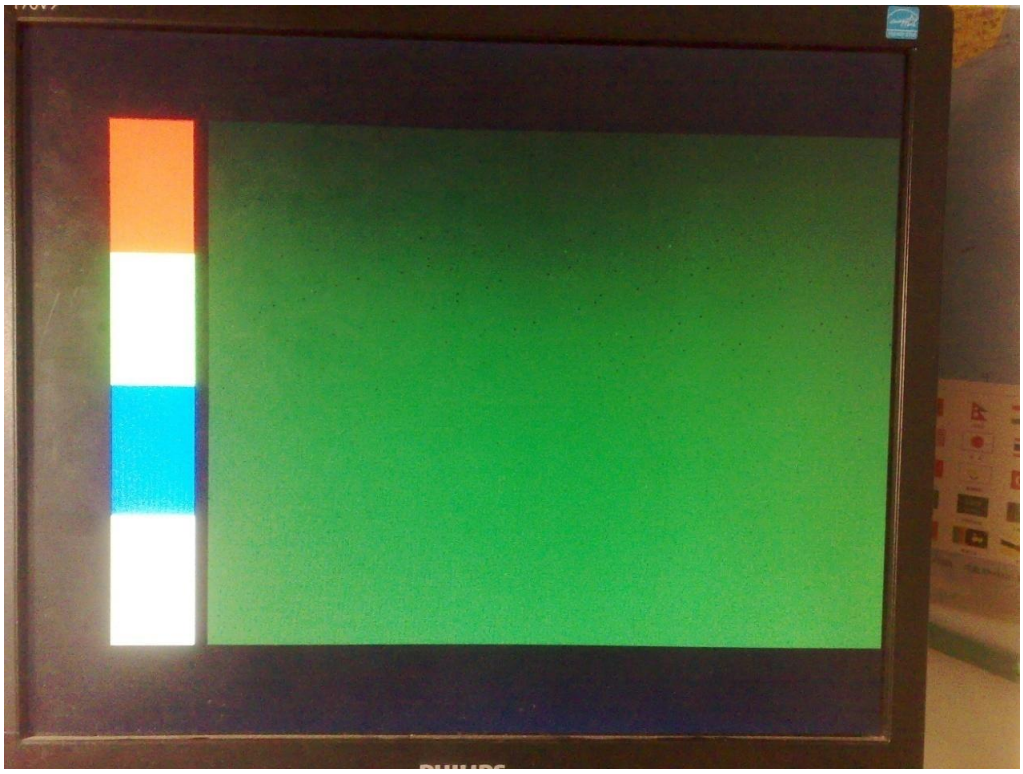

Figure 4.6: Image for hidden camera

After programming, the white screen appeared a few scattered dots but the longer run the more points occurs on the screen show that it is still not possible to control brush function. At this point, the image points increased significantly which is Red, Green and Blue dots appear randomly based on three colors corresponding palette.

Figure 4.7: Painting area on monitor

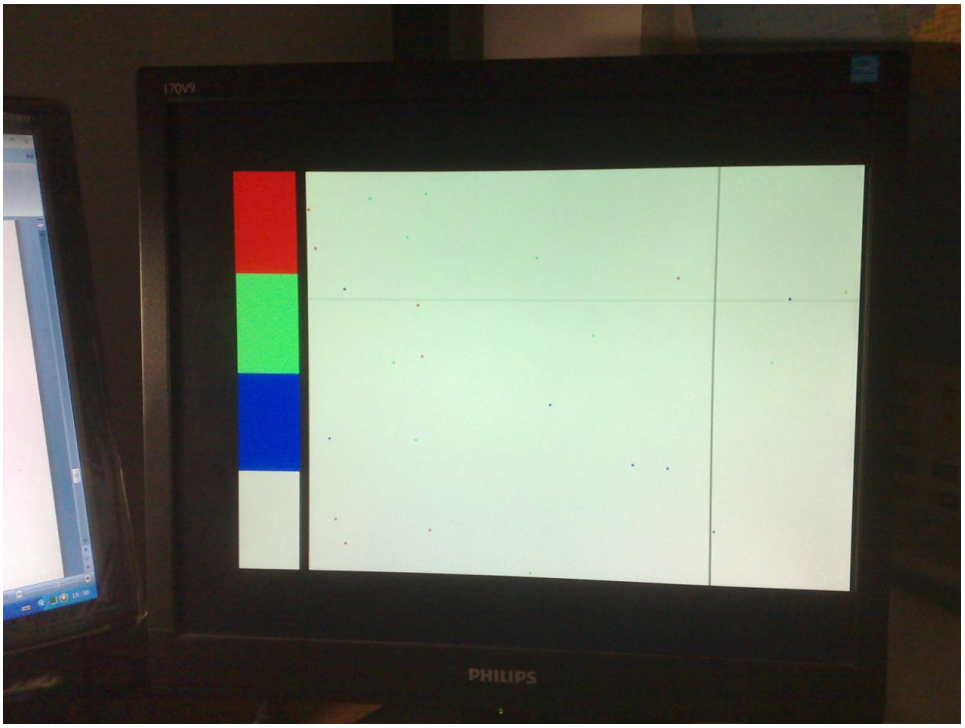

Figure 4.8: Random dot color appear on monitor

REFERENCES

1. M. Petouris, A. Kalantzopoulos and E. Zigouris (2005), “An FPGA-based Digital Camera System Controlled from an LCD Touch Panel”, Electronics Laboratory, Electronics and Computers Div., Department of Physics, University ofPatras, GR-265 00 Patras, GREECE.

2. Suh Ho Lee, Seon Wook Kim and Suki Kim (2004), “Implementation Of A Low Power Motion Detection Camera Processor Using A Cmos Image Sensor”, ISCAS 2004, Department of Electronics Engineering, Korea University, Seoul, Korea.

3. Mohamed Nasir Shukor, Lo H. Liong and P. Sebastian (2007), “Implementation of Color Filtering on FPGA”, Electrical and Electronic Engineering **,** Universiti Teknologi PETRONAS, *International Conference on Intelligent and Advanced Systems.*

4. Nai-Jian Wang, Sheng-Chieh Chang, Pei-Jung Chou (2012), “A Real-time Multi-face Detection System, Implemented on FPGA” *IEEE International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS).*

5. C. T. Johnston, K.T. Gribbon, and D. G. Bailey (2004), "Implementing Image Processing Algorithms on FPGAs", *Proceedings of the Eleventh ElectronicsNew Zealand Conference, ENZCon’04.*

6. A.Downton and D. Crookes (1998), "Parallel architectures for image processing," *Electronics& Communication Engineering Journal*, 1998.

7. Zoltan Pele, Tatj ana Samardzic, Neboj sa Maluckov and Nikola Teslic (2003),"One Application of FPGA Integrated Circuits CMOS Camera and LCD Display Controller Design", *6th International Conference on*

Telecommunications in Modem Satellite, Cable and Broadcasting Services, TELSIKS 2003, Nis, Serbia & Modenegro.

8. G. L. Foresti (1999), "Object recognition and tracking for remote video surveillance," *Circuits and Systems for Video Technology, IEEE Transactions on.*

9. G. L. Foresti, C. S. Regazzoni (1994), "A change detection method for multiple object localization in real scenes", *Industrial Electronics, Control and Instrumentation, IECON '94, 20th International Conference on.*
10. Jacky Baltes (2010), "Efficient Image Processing for Increased Resolution and Color Correctness of CMOS Image Sensors", *5th Robot World Cup Soccer Games and Conferences.*

11. Yakun Wu, Xiaohui Duan, Zhouhui Lian, Tiantian Shen, Shihu Zhao and Lei Van (2008), "The Design of Storage System for Digital Airborne Camera Based SOPC", *Internation al Congress on Image and Signal Processing, CISP 2008, Sanya, Hainan, China*.

12. S.Y. Chien, S. Y. Ma, and L. G. Chen (2002), "Efficient moving object segmentation algorithm using background registration technique, " *IEEE Transactions on Circuits and Systems for Video Technology, Vol. 12, NO. 7,* pp. 577-586.

13. Y. C. Chung, J. M. Wang, and S. W. Chen (2002), "Progressive Background Image Generation, " *in Proc. of 15th IPPR Conf. on Computer Vision, Graphics and Image Processing, pp.* 858-865.

14. L. Maddalena and A. Petrosino (2008), "A Self-Organizing Approach to Background Subtraction for Visual Surveillance Applications," *IEEE Transactions on Image Processing, Vol. 17, NO. 7,* pp. 1168 - 1177.

15. S. S. Ghidary, Y. Nakata, T. Takamori, and M. Hattori (2000), "Human Detection and Localization at Indoor Environment by Home Robot," *in Proc. of IEEE International Conference of Systems, Man, and Cybernetics, Vol. 2,* pp.1360–1365.

16. K. K. Ken, S. H. Cho, H. J. Kim, and J. Y. Lee (2005), "Detecting and tracking moving object using an active camera, " *in Proc. of 7th International Conference of Advanced Communication Technology, ICACT , Vol. 2,* pp. 817–820.

17. D. Murray and A. Basu (1994), "Motion tracking with an active camera", *IEEE Trans. On Pattem Analysis and Machine Intelligence, 16(5),* pp. 449-459.

18. M. Irani, R. Roousso, and S. Peleg (1994), "Recovery of ego-motion using image stabilization*", En Proc. Of the IEEE Computer Vision and Pattern Recognition,* pp. 454-460.

19. B. Jung and G. S. Sukhatme (2010), "Detecting moving objects using a single camera on a mobile robot in an outdoor environment", *In the 8'Conf. on Intelligent Autonomous Systems,* pp. 980-987.

20. Ronald J. Tocci, Neal S. Widmer (2001), "*Digital Systems: Principles and Application",* 8th Edition, Prentice Hall.

21. Rafikha Aliana A. Raof, Norina Idris, Phaklen Eh Kan, Mohammad Nazri Md. Noor (2006), "*Digital Electronics Design",* Prentice Hall.

22. Altera (2006), *"DE2 Education and Development Board User Manual",* Version 1.4.

23. Terasic (2007), *"TRDB-LTM 4.3 Inch Digital Touch Panel Development Kit".*

24. Altera (2008), *"Creating Low Cost Intelligent Display Modules With an FPGA and Embedded Processor",* White Paper.

25. Terasic (2008), *"TRDB-D5M Terasic D5M Hardware Specification"*.

26. John Williams (2008), *"Digital VLSI Design with Verilog"*, Springer.

27. Peter Wilson (2007), *"Design Recipes for FPGAs"*, Elsevier.

28. TRDB D5M User Guide http://www.terasic.com.

29. DE2-70 Datasheets, http://www.altera.com.