FEASIBILITY OF THE POWERPC 603E™ FOR A LEO SATELLITE ON-BOARD COMPUTER

JACU VOS

THESIS PRESENTED IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTER OF SCIENCE (ELECTRONIC ENGINEERING) AT THE UNIVERSITY OF STELLENBOSCH

Supervisor: Prof. P.J. Bakkes

December 2002

DECLARATION

I, the undersigned, hereby declare that the work presented in this thesis is my own original work and that I have not previously in its entirety or in part submitted this work at any university.

Signature

Date

ABSTRACT

For space designs, just as for terrestrial applications, the appetite for more computing power is virtually insatiable. Further, like portable applications, space use implies severe power constraints. Among currently available commercial processors, the PowerPC family ranks high in Million Instructions Per Second (MIPS) per watt, but its suitability for space applications outside low-earth orbits (LEOs) may be limited by the radiation environment, particularly single event effects (SEE).

This thesis covers the feasibility of using the PowerPC 603e™ processor for LEO satellite applications. The PowerPC architecture is well established with an excellent roadmap, which makes for a baseline microprocessor with long-term availability and excellent software support. The evaluation board design leverages Commercial Off-The-Shelf (COTS) technologies, allowing early integration and test. It provides a clear path to upgrades and provides a high performance platform to suit multiple missions.

OPSOMMING

Die soeke na rekenaars met hoër werkverrigting sal nooit ophou nie. Dit geld vir beide rekenaars op aarde as satelliet aanboord rekenaars. Rekenaars vir ruimte gebruik word ook streng drywingsbeperkings opgelê. Die PowerPC familie vergelyk baie goed met ander verwerkers, maar hul bruikbaarheid vir ruimte toepassings kan dalk beperk word tot lae wentelbane waar die ruimte radiasie omgewing meer toeganklik is.

Die skrywe behandel die bruikbaarheid van die PowerPC 603e verwerker vir lae wentelbaan satelliet gebruik. Die welgestelde argitektuur, bekombaarheid en uitstekende sagteware ondersteuning verseker 'n standvastige fondasie. Kommersiële komponente het voorkeur geniet in die hardeware ontwerp wat spoedige ontwikkeling sowel as aanpasbaarheid verseker. Die ontwerp bied 'n hoë werkverrigting en maklik opgradeerbare oplossing vir 'n groot verskeidenheid gebruike.

ACKNOWLEDGMENTS

Most importantly, I wish to thank the Lord God without whom nothing is possible. I hereby also thank the following people for their invaluable help in the successful completion of this thesis: My family for their patience and their interest in my work; Gregor Dreijer for his assistance in bridging the many obstacles I encountered and also for his imperative moral support; Francois Retief for compiling the PowerPC EABI toolchain; and my supervisor, Prof. Bakkes, for his guidance and open-mindedness.

CONTENTS

DECLARATION

ABSTRACT I

OPSOMMING II

ACKNOWLEDGMENTS III

CONTENTS IV

LIST OF FIGURES VIII

LIST OF TABLES IX

LIST OF ACRONYMS XI

OVERVIEW 1

1. INTRODUCTION 2

2. THE POWERPC PROCESSOR 6

2.1. EXECUTION UNITS 6

2.2. BUS INTERFACE 8

2.3. MEMORY MANAGEMENT 8

2.4. POWER MANAGEMENT 9

2.4.1. Dynamic Power Management 10

2.4.2. Static Power Management 11

2.5. TIME BASE/DECREMENTER 12
2.6. TEST INTERFACE 13
2.7. CLOCK MULTIPLIER 13
2.8. PROGRAMMING MODEL 14
2.9. PACKAGE PARAMETERS 15
2.10. RADIATION FACETS 18
2.11. ROADMAP 19
2.12. CONCLUSION 20

3. HARDWARE DESIGN 23

3.1. DESIGN OVERVIEW 23
3.2. FUNCTIONAL DESIGN 24
3.2.1. CPU *25*
3.2.1.1. System Interface 25
3.2.1.2. Power Supply Filtering 27
3.2.1.3. Thermal Considerations 27
3.2.2. FPGA *29*
3.2.2.1. Memory Controller 29
3.2.2.2. EDAC 33
3.2.2.3. Interrupt Controller 35
3.2.2.4. UARTs 36
3.2.2.5. I^2C Controller 36
3.2.2.6. Reset Controller 37
3.2.2.7. Debug & Expansion 37
3.2.3. FLASH ROM *38*
3.2.4. SRAM *38*
3.2.5. Clock Generator *38*
3.2.6. Real-Time Clock *39*
3.2.7. Reset Generator *39*
3.2.8. SCC *40*
3.2.9. LVDS *41*
3.2.10. Temperature Sensor *42*
3.2.11. Current Sensor *42*
3.2.12. ADC *43*
3.2.13. Power *43*
3.3. PHYSICAL LAYOUT 44
3.4. CONCLUSION 44

4. SOFTWARE DEVELOPMENT **47**

4.1. DEVELOPMENT ENVIRONMENT 47

4.2. THE POWERPC EABI 47

4.3. PROCESSOR INITIALIZATION 49

4.4. SOFTWARE INTERFACING 51

4.4.1. UARTs *51*

4.4.2. I^2C Controller *53*

4.4.3. RTC *54*

4.4.4. Temperature Sensor *54*

4.4.5. Clock Generator *56*

4.4.6. Interrupt Controller *56*

4.4.7. LVDS *57*

4.4.8. SCC *58*

4.4.9. System Management *58*

4.4.10. Debug & Expansion *58*

4.5. CODE UPLOAD 59

4.6. DEMO PROGRAM 60

4.7. SRAM DUMP 61

4.8. CONCLUSION 62

5. TEST AND MEASUREMENT **63**

5.1. PROCESSOR TEMPERATURE 63

5.2. POWER CONSUMPTION 64

5.3. POWER-UP CURVES 64

5.4. SRAM TEST 66

5.5. CONCLUSION 66

6. CONCLUSIONS AND RECOMMENDATIONS **67**

REFERENCES **69**

APPENDIX A : REGISTER BIT ASSIGNMENTS **A-1**

A1. UART REGISTERS A-1

A2. I^2C CONTROLLER REGISTERS A-3

A3. INTERRUPT CONTROLLER REGISTERS A-4

A4. LVDS REGISTERS A-7

A5. SYSTEM MANAGEMENT REGISTER A-7

A6. DEBUG/EXPANSION REGISTERS A-8

APPENDIX B : SUPPLEMENTARY CD **B-1**

LIST OF FIGURES

Figure 1-1. Simplified Block Diagram of SUNSAT 3
Figure 2-1. Block Diagram of the 603e™ 7
Figure 2-2. Execution Unit Idle Time for SPEC92 10
Figure 2-3. 603e Register Set 14
Figure 2-4. CBGA Solder Fatigue Field Life Projections 16
Figure 2-5. CBGA and Compliant Lead System 17
Figure 2-6. E-tec BGA Adapter System 17
Figure 2-7. PowerPC™ Roadmap 20
Figure 3-1. Hardware Block Diagram 24
Figure 3-2. PLL Power Supply Filter Circuit 27
Figure 3-3. Simplified Thermal Network of a C4/CBGA Package mounted to a PCB 28
Figure 3-4. Memory Controller Architecture 31
Figure 3-5. Single-Beat Reads Showing Data-Delay Controls 34
Figure 3-6. EDAC Unit Simulation Results 35
Figure 3-7. Internal Structure of I^2C Controller 37
Figure 3-8: Integration of Reset Sources with Reset Controller 40
Figure 3-9. PowerPC 603e™ OBC Evaluation Board Layout 45
Figure 4-1. Flow Diagram of RTC Read Procedure 55
Figure 4-2. Block Diagram of S-Record Programmer 60
Figure 5-1. Temperature and Current Curve at 200MHz 64
Figure 5-2. Temperature and Current Curve at 150MHz 65
Figure 5-3. Temperature and Current Curve at 30MHz 65

LIST OF TABLES

Table 1-1. Processor Specification Summary 5
Table 2-1. Page Access Protection 9
Table 2-2. Dynamic Power Management Results 11
Table 2-3. Static Power Management Results 12
Table 3-1. Evaluation Board Memory Space 32
Table 4-1. PowerPC EABI Registers 49
Table 4-2. UART Registers 52
Table 4-3. I2C Controller Registers 54
Table 4-4. Interrupt Controller Registers 57
Table 4-5. LVDS Registers 57
Table 4-6. System Management Registers 58
Table 4-7. Debug/Expansion Registers 59
Table 5-1. Temperature Measurements 63
Table A1-1. UART Control Register A-1
Table A1-2. UART Status Register A-2
Table A2-1. I^2C Control Register A-3
Table A2-2. I^2C Command Register A-3
Table A2-3. I^2C Status Register A-4
Table A3-1. Interrupt Mask Register A-4
Table A3-2. Interrupt Pending Register A-5
Table A4-1. LVDS Control Register A-7
Table A4-2. LVDS Status Register A-7

Table A5-1. System Management Register A-7
Table A6-1. Port Register A-8
Table A6-2. Direction Register A-9
Table A6-3. LED Register A-9
Table A6-4. Switch Status Register A-9

LIST OF ACRONYMS

ACRONYM	DESCRIPTION
ADC	Analog-to-Digital Converter
BGA	Ball-Grid Array
BLVDS	Bus LVDS
C4	Controlled-Collapse-Chip-Connection
CBGA	Ceramic Ball-Grid Array
COP	Common On-chip Processor
COTS	Commercial Off The Shelf
CPU	Central Processing Unit
DMA	Direct Memory Access
EAB	Embedded Array Block
EDAC	Error Detection And Correction
FIT	Failures In Time
FPGA	Field-Programmable Gate Array
FR4	Woven Glass, Flame Retardant Epoxy Resin
I^2C	Inter-Integrated Circuit
I_2O	Intelligent Input/Output
JTAG	Joint Test Action Group
LEO	Low-Earth Orbit

ACRONYM	DESCRIPTION
LET	Linear Energy Transfer
LUT	Look-Up Table
LVDS	Low Voltage Differential Signalling
MIPS	Million Instructions Per Second
MSR	Machine State Register
OS	Operating System
PBGA	Plastic Ball-Grid Array
PCI	Peripheral Component Interconnect
PGA	Pin Grid Array
PLL	Phase Locked Loop
RAM	Random-Access Memory
ROM	Read-Only Memory
RTC	Real-Time Clock
SCC	Serial Communications Controller
SEE	Single Event Effects
SEL	Single Event Latchup
SEU	Single Event Upsets
SOI	Silicon-on-Insulator
SRAM	Static Random-Access Memory
TCE	Thermal Coefficient of Expansion
TID	Total Ionising Dose

OVERVIEW

This thesis is divided into 6 chapters and gives detail descriptions on the design process, test results and measurements. The first two chapters provides some background on OBCs and the PowerPC 603e™ architecture. Estimated performance and power figures are also presented here and will be compared to actual measured data in the final chapters.

The following chapters introduce the hardware and elucidate the design process in detail. The hardware includes the modules pertained in the support FPGA in addition to the physical devices.

Following the hardware design, the software driver routines and initialisation code is presented. The development environment and the PowerPC Embedded Applications Binary Interface (EABI) are also explained together with some debugging routines.

The final chapters present measurements and simulation results and the thesis ends with some conclusive remarks and recommendations.

1

INTRODUCTION

On-board computers, or OBCs, are an essential subsystem of any spacecraft. Other subsystems may be intelligent and be able to make decisions, but the main course of action taken by them is still calculated by the OBC. The OBC interfaces with most of the other subsystems on a digital and analog level and its design is highly dependent on the mission objective. The most important tasks of an OBC are:

- Regulation of attitude
- Communication with earth
- Measurement of analogue signals, e.g. temperature, battery voltage, etc.
- Payload specific tasks, e.g. camera control.

Figure 1-1 contains a simplified block diagram of the *SUNSAT* micro satellite showing how the OBC connects to the various other satellite subsystems. From the figure, it is clear that the OBC directs the other subsystems. It is therefore necessary for the OBC to communicate via the various protocols of the other subsystems. *SUNSAT* was built using only COTS components.

COTS electronics offer compelling value not just because they are commercial or off-the-shelf. Their greatest value for use in high reliability applications lies in their high-volume production and widespread use: both drive down cost and drive up yield, performance and reliability. Higher speed and superior electrical performance his led to an increased interest in the possible use of unhardened commercial microprocessors in space. The use of

COTS open standards drives down development and life-cycle support costs as well as reducing time to market for new products.

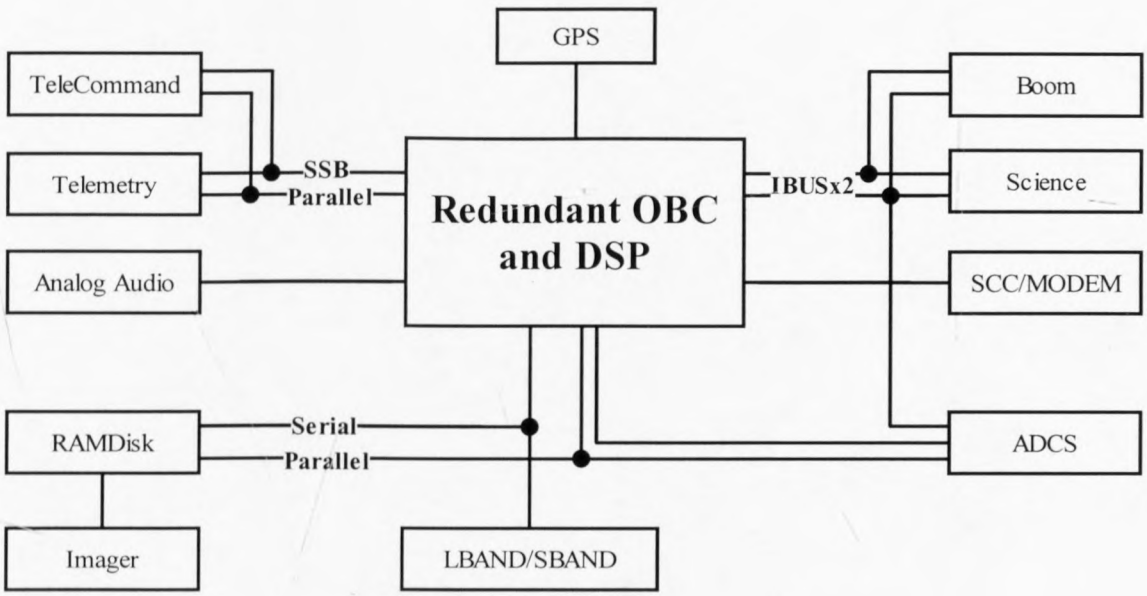


Figure 1-1. Simplified Block Diagram of SUNSAT

While a debate continues in the space community regarding the merits and even the definition of COTS, most everyone can agree that adopting some aspects of the COTS electronics industry technology into the space electronics industry is a certainty.[2]

Still, using COTS devices all over the design will not ensure that the high-reliability objective, present in all satellite OBC designs, be met; especially not in the acrimonious space radiation environment. Unhardened devices are susceptible to upset and degradation from radiation and information is needed to determine the resulting effects and how they can be detected and overcome.

COTS devices are used throughout the design and data integrity is ensured by an EDAC-enabled memory controller implemented in a FPGA. This also makes the design extremely flexible as all peripherals and IO devices are controlled by this custom FPGA that acts as a bridge between the processor and its memory space.

Devices were selected with the low-power and high performance objectives, inherent to any satellite subsystem, in mind. Most of the peripherals used boast power-saving modes with fast recover times, allowing maximum efficiency and performance. For example, the FLASH ROM features an automatic power-down mode when it has been idle for a fixed amount of time and wakes up instantaneously on bus activity, inflicting no performance penalty. Lessons learned with the development of *SUNSAT* were also incorporated. The use of the Serial Communications Controller (SCC), for example, was based solely on its high reliability and successful use throughout the mission. In addition, the OBCs used on *SUNSAT* were used as reference designs to ensure a realistic OBC test platform.

The first step in the selection of a processor for an OBC is the compilation of a trade's table. Table 1-1 was used in this processor selection with each processor evaluated based on their merit for the respective categories. The table is adopted from a similar study done by the Southwest Research Institute. They successfully developed a VME space qualified radiation hardened workstation class processor board around the Thompson 603e processor. Advanced high performance microprocessors, especially reduced instruction set, or RISC, are highly desired in the design and implementation of OBCs. PowerPCs are powerful RISC processors with significant inherent radiation tolerance.[5] As shown in section 1.10, the SEE (Single Event Effects) rates and TID (Total Ionising Dose) tolerance makes the PowerPC 603e processor suitable for a wide range of commercial aerospace applications.[3]

The table clearly shows the merit of the PowerPC processors. The radiation-hardened parts are software and pin-to-pin compatible with the commercial parts offering fast and inexpensive development. The ARM architecture uses significantly less power but with less fixed point performance as the PowerPC devices. The StrongARM devices are also not equipped with a floating-point unit. No information regarding the radiation tolerance or event susceptibility of the StrongARM processor could be obtained.

Table 1-1. Processor Specification Summary

PROCESSOR/ MANFACTURER	DVELOPMENT TOOLS	CLOCK RATE	MIPS	POWER	RADIATION		
					TID KRAD	SEUs/ DAY	SELs/ DAY
603r/Motorola	Extensive	300MHz	480	4W	26	$3x10^{-1}$	N/A
603ev/Thompson	Extensive	100MHz	110/	3W	60	$1x10^{-1}$	$>1x10^{-10}$
603ev/Honeywell	Extensive	100MHz	110	N/A	100	N/A	N/A
StrongARM/ Intel	Good	206MHz	235	0.35W	N/A	N/A	N/A
486/Intel	Extensive	66MHz	54	4.8W	30	$3x10^{-1}$	$>4x10^{-5}$
Pentium/Intel	Extensive	100MHz	108	6.45W	100	N/A	N/A
RAD6000/Lock heed Martin	Minimal	33MHz	34.6	3.86W	>100	$1x10^{-1}$	$>1x10^{-10}$
Thor	Minimal	50MHz	50	4.4W	>100	$>1x10^{-6}$	$>1x10^{-10}$
AD2010/Temic	Minimal	20MHz	20	2.15W	>100	$>1x10^{-7}$	$>1x10^{-10}$

The MPC603RRX266LC part from Motorola was used mainly due to the excellent software and hardware support Motorola offer and the ease with which the part was acquired. It can operate at frequencies up to 266MHz with the bus clock frequency as high as 75MHz. Motorola provides a wide range of reference designs and application notes for the entire PowerPC family.

The following chapter provides some background on the PowerPC 603e™ processor and its architecture.

2

THE POWERPC PROCESSOR

The 603e is a low-power implementation of the PowerPC microprocessor family of RISC microprocessors. It implements the 32-bit portion of the PowerPC architecture, which provides 32-bit effective addresses, integer data types of 8, 16 and 32 bits, and floating point data types of 32 and 64 bits.

The 603e is manufactured in a 2.5/3.3V 0.29 μm CMOS fabrication process using five layers of metal in a fully static design. The die size is 44mm^2 and the operating die-junction temperature can range from 0°C to 105 °C.

The architectural organisation of the 603e focuses on high performance, low cost and power management. This chapter enumerates the main design features of the PowerPC 603e™ microprocessor and its architecture.

2.1. EXECUTION UNITS

The 603e is a high-performance, superscalar microprocessor and capable of issuing/retiring as many as three instructions per clock. It contains five independent executions units, hence can have as many as five instructions in execution per clock since most instructions are single-cycle executable.

These five units include the following:

- Branch Processing Unit featuring static branch prediction, zero cycle branch capability (branch folding), and Conditon Register (CR) lookahead operations

- 32-bit Integer Unit
- Fully IEEE 754-complient pipelined FPU for both single- and double-precision operations
- Load-Store Unit
- System Register Unit that executes CR and Special Purpose Register (SPR) instructions and dispatches/ executes multiple integer add/compare instructions per cycle

Figure 2-1 provides a high-level block diagram of the 603e that shows how these execution units operate independently and in parallel.

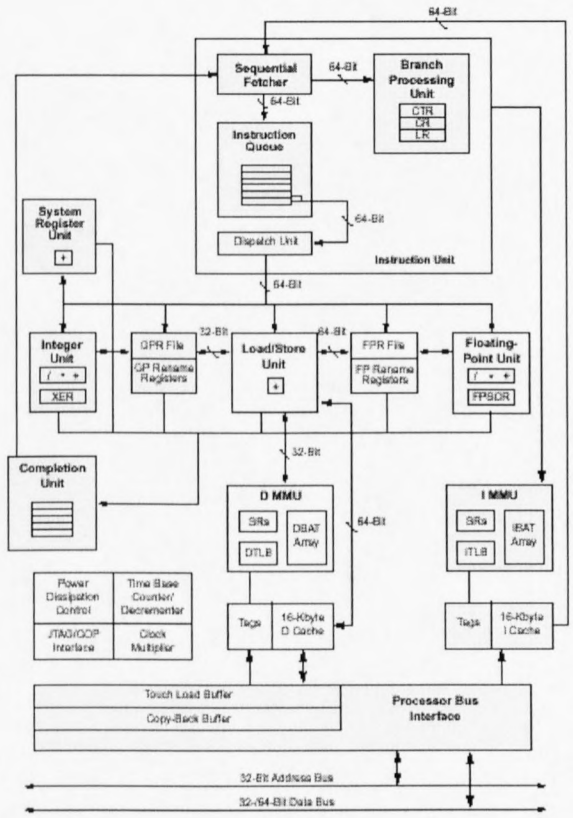


Figure 2-1. Block Diagram of the 603e™

2.2. BUS INTERFACE

The 603e features a selectable 32- or 64-bit data bus and a 32-bit address bus. The bus protocol allows split-transactions, burst transfers, and out-of-order transactions with one-level address pipelining. The efficient processor bus interface facilitates access to main memory and other bus subsystems. It also allows multiple masters to compete for system resources through a central external arbiter. This arbitration is flexible, allowing the 603e to be integrated into systems that implement various fairness and bus parking procedures to avoid arbitration overhead.

Typically, memory accesses are weakly ordered, maximising the efficiency of the bus without sacrificing coherency of the data. Overall performance is improved by dynamically optimising run-time ordering of load/store traffic.

The 603e maintains full interface compatibility with TTL devices and all pins are 5V compatible. Parity is also optionally provided on the data and address buses and is software-enabled.

2.3. MEMORY MANAGEMENT

The 603e provides independent 16-Kbyte on-chip data and instruction caches with separate memory management units (MMUs). The MMUs support up to 4 Petabytes (2^{52}) of virtual memory and 4 Gigabytes (2^{32}) of physical memory. Demand-paged virtual memory is supported by the PowerPC 603e and permits execution of programs larger than the size of physical memory. Demand-paged implies that individual pages are loaded into physical memory from system memory only when they are first accessed by an executing program.

The caches are four-way set-associative, physically addressed and are enabled/disabled via software. The MMUs contain 64-entry, two-way set-associative, data and instruction translation look-aside buffers (TLBs). Both the caches and TLBs use a least recently used (LRU) replacement algorithm. The 603e also supports block address translation using two

independent instruction and data block address translation (IBAT and DBAT) arrays of four entries each.

The MMUs also directs the address translation and enforces the protection hierarchy programmed by the operating system in relation to the supervisor/user privilege level of the access and in relation to whether the access is a load or store. Table 2-1 shows the eight protection options supported by the MMUs.

Table 2-1. Page Access Protection

Option	User Read		User Write	Supervisor Read		Supervisor Write
	I-Fetch	Data		I-Fetch	Data	
Supervisor-only	—	—	—	√	√	√
Supervisor-only-no-execute	—	—	—	—	√	√
Supervisor-write-only	√	√	—	√	√	√
Supervisor-write-only-no-execute	—	√	—	—	√	√
Both user/supervisor	√	√	√	√	√	√
Both user/supervisor-no-execute	—	√	√	—	√	√
Both read-only	√	√	—	√	√	—
Both read-only-no-execute	—	√	—	—	√	—

Note:
√ Access permitted
— Protection violation

2.4. Power Management

The 603e provides four software controllable power-saving modes. Three of the modes (nap, doze, and sleep) are static in nature, and progressively reduce the amount of power dissipated by the processor. The fourth is a dynamic power management mode that causes the functional units in the 603e to automatically enter a low-power mode when the functional units are idle without affecting operational performance, software execution, or any external hardware.

2.4.1. Dynamic Power Management

Dynamic power management is a software-enabled mode that turns on power-saving logic for the execution units, caches and memory management units during normal operation. Once the mode is enabled, no other software intervention is necessary.

The dynamic power management logic automatically manipulates clock regenerators to reduce average power consumption by eliminating clock switching and inhibiting change of registered values. Because the 603e features a fully static design, if registered values do not change, the logic those values feed does not switch and thus reduces power consumption.

The effectiveness of using dynamic power management in the execution units varies with the type of code run. For example, the floating-point unit clocks will be frozen continuously if integer-only code is run. However, if code is scheduled such that all of the units remain busy continuously, few clocks will be frozen.

Figure 2-2 illustrates that for some applications, each functional unit may be idle during a large percentage of the run time.[6] Since dynamic power management freezes execution unit clocks during this idle time, it can be effective in reducing the average power consumption of the execution units. The statistics were collected while running SPEC92 benchmark traces.

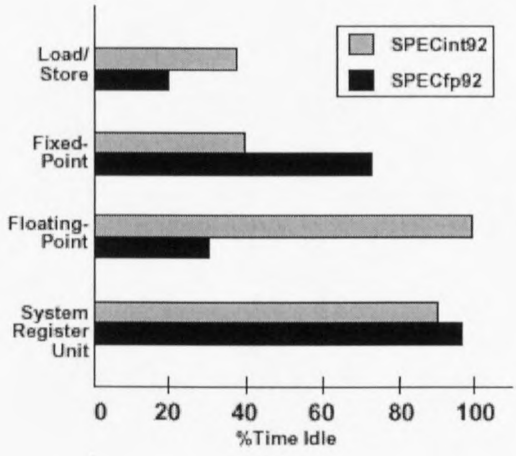


Figure 2-2. Execution Unit Idle Time for SPEC92

Since the 603e has blocking caches, all other accesses are held off until cache miss data is returned from memory. During this time the caches and MMUs are idle and can have their clocks disabled.

Table 2-2 shows the percentage decrease in internal (core) power dissipation for various applications if dynamic power management is used.[6] The table data was obtained using an earlier 603e which was manufactured in a 0.5µm 3.3V CMOS process using 4 layers of metal and a die size of 98mm^2. Still, these figures provide a feel for power savings obtainable with the dynamic power management feature.

Table 2-2. Dynamic Power Management Results

C LINPACK	DHRYSTONE	HANOI	HEAPSORT	NSIEVE
8.5%	14.0%	13.8%	14.2%	16.4%

2.4.2. Static Power Management

As mentioned earlier, the 603e provides three static power management modes, Doze, Nap and Sleep. They allow the operating system or power management software to reduce average power consumption when the 603e is idle for any extended period. Once in one of these modes, an external event, such as the external interrupt pin, will bring the 603e out of that mode and instruction execution will resume by jumping to the address of the appropriate interrupt vector.

Doze mode retains cache coherency as well as uninterrupted timer functionality. The 603e will exit Doze mode in ten system clocks (SYSCLKs) or less, depending on the processor to bus clock ratio. In Nap mode, further power savings are achieved with the data cache and snooping logic disabled. The wake-latency for Nap mode is also ten SYSCLKs or less. Sleep mode allows for maximum power savings by disabling the clocks to all units, including the PLL and/or SYSCLK if desired. This allows the system to dynamically

manage system power by changing the SYSCLK frequency, or change the Phase Locked Loop (PLL) configuration. The wake-up time for Sleep mode is ten SYSCLKs or less is the PLL and SYSCLK remain active and unaltered. If they were disabled, a maximum of 100μsec is required for the PLL to lock.

Table 2-3 shows total power consumed by the 603e using the static power management modes. Again, these results were obtained using the earlier 603e and serve only as a guide to obtainable power savings.

Table 2-3. Static Power Management Results

Freq (MHz)	Power (mW)				
	Doze	Nap	Sleep		
			PLL on CLK on	PLL off CLK on	PLL off CLK off
25	133.2	49.4	38.8	12.8	4.7
33	168.0	62.0	47.8	13.5	4.7
50	241.7	88.8	66.2	15.1	4.7
66	307.1	113.0	88.5	17.7	4.7
80	366.1	135.1	105.5	19.3	4.7

2.5. Time Base/Decrementer

The time base is a 64-bit register that is incremented once every four bus clock cycles while the decrementer is a 32-bit register is decremented once every four bus clock cycles. The decrementer generates an interrupt exception after a programmable delay and can be used as a software watchdog.

If the operating system initialises the Time Base to some reasonable value and the update frequency of the Time Base is constant, it can be used for time-stamps in trace entries (scheduler algorithms) and for computing time of day.

2.6. TEST INTERFACE

Board testing and chip debugging is facilitated primarily by IEEE 1149.1 (JTAG) and Common On-chip Processor (COP) functions. The JTAG interface provides a means for boundary-scan testing of the 603e and the board. The COP function provide a means for executing test routines, and facilitates chip and software debugging using a PC with dedicated hardware and software. The standard COP interface header was used for connection to the target system.

2.7. CLOCK MULTIPLIER

A voltage-controlled oscillator-based PLL generates and synchronises the internal clocking from/to the external clock signal, SYSCLK. The PLL can be programmed via four external signals to allow for different internal clock frequencies that are multiples of SYSCLK. This configuration of the PLL can be read by software.

The PLL is designed to lock to a wide range of SYSCLK frequencies from 16 MHz up to 75 MHz. If lower frequencies are desired, the PLL can be bypassed altogether, allowing SYSCLK to drive the internal clocking circuitry directly. Care should however be taken when using this mode, since the timing specifications of the 603e is not guaranteed in PLL-bypass mode.

The PLL configuration cannot be changed while the processor is running. It can be changed:

- While the processor is off
- While the processor is in reset
- While the processor is in Sleep mode with the PLL and SYSCLK disabled

The last two options allow the system to change the processor clocking speed dynamically without affecting system memory. The operating system can simply put the 603e into Sleep mode and alter the SYSCLK frequency and/or the PLL configuration.

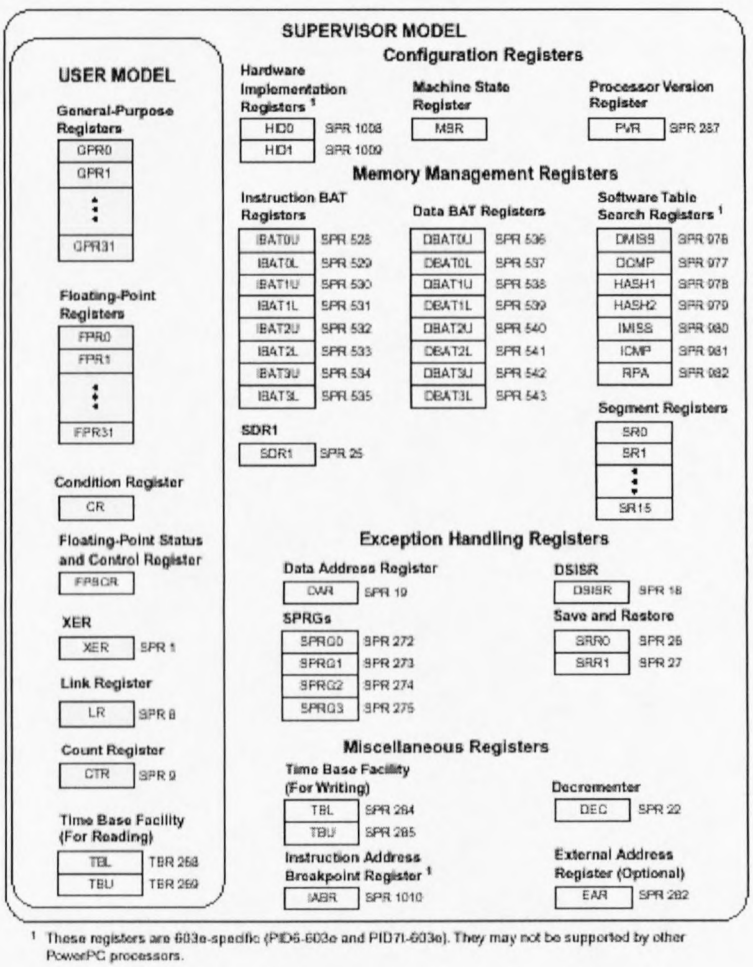
2.8. PROGRAMMING MODEL

PowerPC processors have two levels of privilege:

- Supervisor mode of operation is usually used by the operating system, and
- User mode of operation is used by the application software.

Having access to privileged instructions, registers, and other resources allows the operating system to control the application environment and protect critical resources.

Figure 2-3 shows all the 603e registers available at both the user and supervisor level.



1 These registers are 603e-specific (PID5-603e and PID7t-603e). They may not be supported by other PowerPC processors.

Figure 2-3. 603e Register Set

The user-level registers can be accessed by all software with either user or supervisor privileges. The supervisor model register set can be accessed only by supervisor-level instructions. A supervisor-level exception will be taken when access is attempted with user-level instructions.

2.9. PACKAGE PARAMETERS

The PowerPC 603e™ part from Motorola comes in Ceramic Ball-Grid Array (CBGA) and Plastic Ball-Grid Array (PBGA) packages. From the PCB assembler's point of view, there are many advantages in using BGA packages as well as some disadvantages. A key advantage is the significantly better assembly yields for BGA packages.[2] There are no bent leads and no coplanarity problems. The most apparent disadvantage is that most of the solder joints are hidden under the package and makes visual inspection of the joints virtually impossible. In addition, the PBGA packages are moisture sensitive at present. CBGA packages are high thermal mass packages, and as such, require careful reflow profile management to insure sufficient and uniform heat.

Solder ball metallurgy also differs between the PBGA and CBGA packages. The PBGA solder balls melt at ~179°C, thus reflow and collapse in standard surface mount reflow processes to provide ~20 mils stand off from the board. The CBGA solder balls melt at ~301°C and do not melt in standard surface mount processes. Eutectic solder paste printed on the pads of the PC board melts at standard surface mount reflow temperatures and wicks up to the high lead solder ball. Because the high temperature ball does not melt and collapse, the package will be held about 35 mills above the board compared to 15mils for PBGA packages.

Due to the excellent, self-centring nature of all BGAs, both ceramic and plastic, the placement accuracy requirement is very loose. Placement can be off by 12 mils or more and the package will perfectly self-centre.

In this design, the high-volume CBGA device was used as it comes off the production line. The CBGA package has much lower thermal resistances with respect to the die than the

PBGA package and is thermally more reliable. The PBGA package only ships at 200MHz while the CBGA parts come in 200MHz to 300MHz versions. Also, it was estimated that the solder joint fatigue life for the 21mm x 21mm CBGA package at an average of on-off ΔT of 20°C to be over 25 years, where ΔT is the temperature rise above ambient at the solder joint.[11]

Figure 2-4 shows the solder joint fatigue life for the 21mm x21mm CBGA, the package for the PowerPC 603 microprocessors, and the 21mm x 25mm CBGA, the package for the MPC 106 bridge devices. In the figure, each cycle represents the full temperature excursion. It can be seen that a "worst-case" desktop situation with two on-off cycles per day would still yield over 10 years of life for both packages.

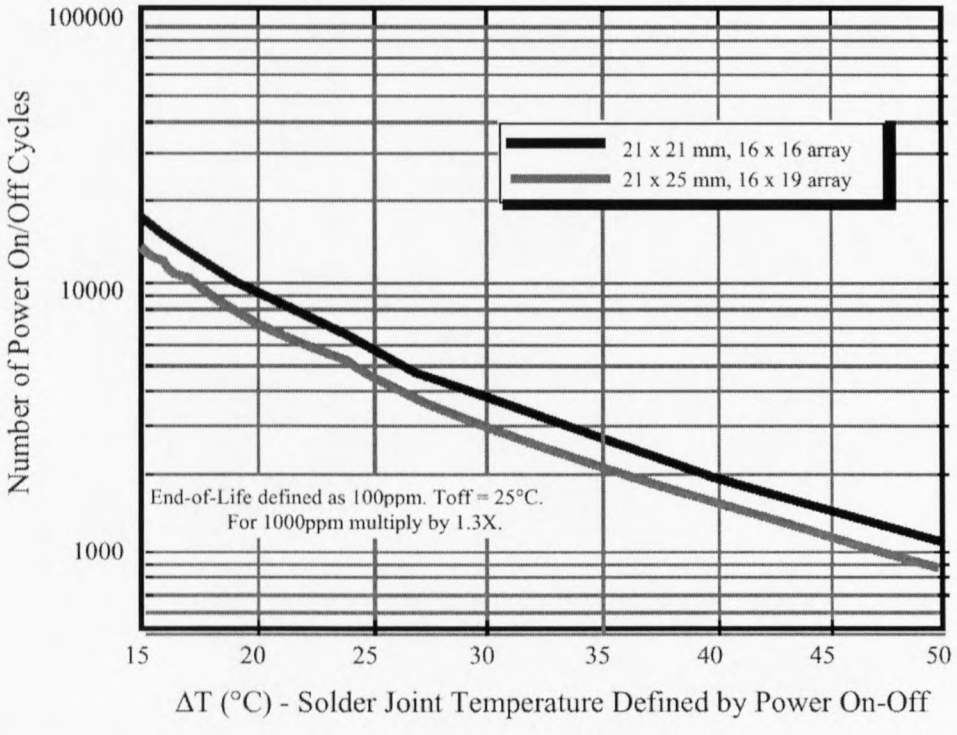


Figure 2-4. CBGA Solder Fatigue Field Life Projections

A compliant lead system can be added to the commercial packaging to increase the thermal-cycling capability of the device. Figure 2-5 shows a compliant lead system that can be

added to the CBGA package to eliminate the TCE (Transfer Coeëficient of Expansion) mismatch problem.[2]

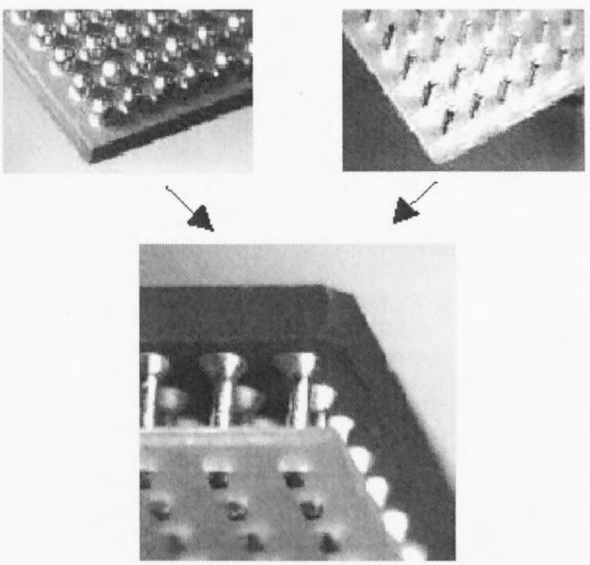


Figure 2-5. CBGA and Compliant Lead System

Figure 2-6 shows another solution for increasing the thermal capabilities of the device. This adapter system is manufactured by *E-tec* and comprises two elements: a BGA solder adapter and a PGA socket that is soldered to the PCB. The BGA chip is soldered to the adapter, converting the BGA to a PGA, and can then be plugged into the socket. The adapter system is made from glass epoxy and has an operating temperature range of -66°C to +130°C.

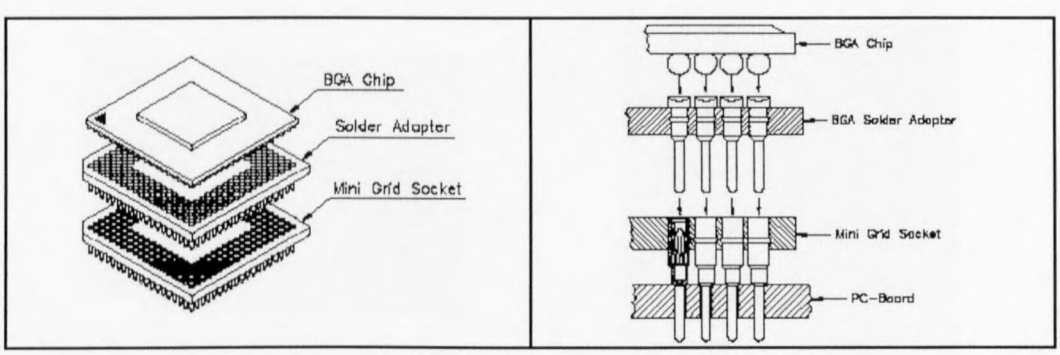


Figure 2-6. E-tec BGA Adapter System

Care must be taken when using BGA socket adapters as they increase the power and ground inductance. One might expect the power and ground inductance to increase by 3% to 7% as a result of the short distance between the balls and PCB.

2.10. RADIATION FACETS

Compared to previous generations, many modern deep sub-micron CMOS microcircuits are very tolerant to TID radiation, but they are still sensitive to SEUs.[1] Due to its fabrication technology, the COTS PowerPC has excellent total dose performance and is immune to latchup.[2,3] The custom support FPGA solves its SEU problems and helps exploiting the full performance of the processor while maintaining full software compatibility with COTS technology.

The results of radiation tests performed on an older version of the PowerPC 603 family, the PowerPC 603e™, were studied.[3] This processor was manufactured in a 0.5µm, 4 metal layer process with approximately 2.6 million transistors on a 98mm^2 die and operated at 100MHz and 3.3V. The on-chip data and instruction caches were enabled for all exposures and the test module was intended to be cache and I/O intensive.

Their measurements indicated that the device has a total dose tolerance of approximately 26 krad (Si) before functional failure. SEE measurements on complex microcircuits, such as a processor, are highly sensitive to the application software used to perform the tests. Nonetheless, the tests showed an overall event threshold of 17.2 to 18.2 MeV and an asymptotic cross section of $5\text{x}10^{-9}$ to $1\text{x}10^{-8}$ cm^2/device for energetic protons. For its sensitivity to heavy ions, an application of the Calvel PROFIT model was used to estimate the event threshold to be between 2.2 and 2.5 MeV/(mg/cm^2) and an asymptotic cross section 0.021 to 0.042 cm^2/device. Using these results the projected event rate for a sample circular low-earth orbit of 1400km, 48-degree inclination angle, Solar Max, and 100mil Aluminium shielding to be between 1.8 and 3.8 days per event. Other tests show that the inherent TID tolerance of the PowerPC 603™ processors are adequate for orbits below 1400km and above 8000km for 15 years.[2]

Radiation tests performed by the Jet Propulsion Laboratory measured SEU in commercial and silicon-on-insulator (SOI) PowerPC microprocessors.[4] They used SOI PowerPC 750™(G3) and PowerPC 740™(G4) processors from both Motorola and IBM and compared the results to that of the commercial processors with bulk substrates. They found that the SEU sensitivity of SOI PowerPC devices were nearly the same as those of the standard commercial processors from the same manufacturers. The cross sections of the SOI processors are lower than their counterparts with bulk substrates, but the linear energy transfer (LET) threshold is about the same, offering little advantage over the commercial parts. The results also show that the upset rates are low enough to allow these devices to be used in space applications where occasional register or functional operating errors can be tolerated. The SEL rate was estimated to be one in 25 years resulting from galactic cosmic rays in deep space.

STM manufactures a 60krad and Honeywell a 100krad version of the PowerPC 603 processor. These can be used for increased reliability and lifetime.

2.11. ROADMAP

The PowerPC 603e is software- and bus-compatible with the PowerPC 604, PowerPC 740 and PowerPC 750 microprocessor families, which allows for rapid upgrades. Figure 2-7 shows Motorola's PowerPC processor strategy. The G5 series are available in parts capable of operating at frequencies of 2GHz. The upward compatibility policy of the PowerPC strategy lets new designs take advantage of the increased performance of the new devices immediately.

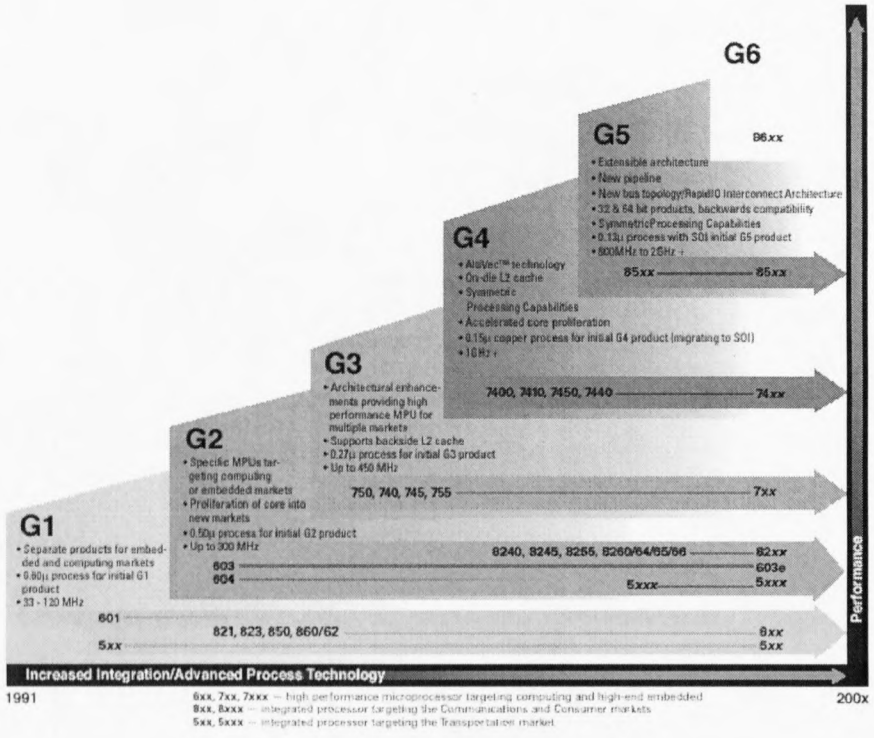


Figure 2-7. PowerPC™ Roadmap

2.12. CONCLUSION

These design features optimise the 603e for both power and performance, creating an ideal microprocessor solution for portable applications. Its inherent radiation tolerance makes it equally ideal for OBCs. Its excellent roadmap and the extensive suite of software tools available facilitates fast and easy development. The memory management unit makes software development easier by disallowing processes to access the same resources.

Regarding its feasibility for aerospace applications, the commercial PowerPC 603e™ processor embraces the following:

1. Good immunity to SEL[2],
2. Built by two world-class suppliers (Motorola and IBM),

3. Excellent reliability (low FIT rate),
4. Good MIPS/Watt performance,
5. Roadmap for increased performance,
6. Extensive software support, and
7. Long-term availability.

CMOS technology and careful physical design can provide low semiconductor junction and joint temperatures for longer component life and higher reliability.

The conditions affecting solder joint reliability are:

- Mechanical vibration or impact
- Bending of boards
- Thermal shock
- External temperature
- Power dissipation internally
- Thermal cycling
- Solder paste composition

A satellite is subjected to about all of these conditions during its lifetime. The temperature conditions can be improved to some extent by using a different PCB fabrication material. The CBGA package, as a high thermal mass package, has a low CTE. Thermount®, for example, more closely match the CTE of the package. Thermount® has a CTE of 9 to 11 ppm/°C while FR-4 typically has a CTE of 15-17/50ppm/°C (x-y/z). It is therefore recommended by PCB manufacturers for high-reliability CBGA applications.

The PowerPC family of microprocessors are inherently immune to SEL and their SEU sensitivity is good enough for them to be used in orbits of lower than 1400km and higher than 800km.[2,3,4] Upset rates under these conditions are projected to range from once every 25 years, resulting from galactic cosmic rays in deep space, to about one every 1,8 to 2,8 days, when subjected to radiation in a 1400km circular orbit, with only modest improvements when using SOI technology.

The TID tolerance of the processor allows a projected lifetime in a 1400km circular orbit of about 15 years. The PowerPC processor is thus adequate for LEO orbits where occasional register or functional operating errors are tolerable.

The focus of the next chapter, Chapter 2, is the hardware design of the evaluation board. The devices used and their interfacing with the processor will be discussed.

3

HARDWARE DESIGN

This chapter is devoted to the hardware design of the PowerPC™ OBC evaluation board. Firstly, the functions and features of the devices used in the design will be discussed. The integration of these devices will then be presented followed by some conclusive remarks.

COTS devices offer many advantages over their space-qualified counterparts and are therefore used extensively throughout the design. Flexibility is one of the main concerns when designing an OBC. Hardware flexibility enables the adding and removal of certain divides (modules) with no adverse affect on the overall performance of the system. This allows for adaptation when certain devices become defective. Flexible software allows the use of operating systems and routines across platforms and specific peripherals. This expedites further work on the same system, but also expedites the design and turn-around time of new systems if advantage of such software is taken.

3.1. DESIGN OVERVIEW

The requirements of an on-board computer is usually mission dependent, i.e. different satellites will require different approaches to the design thereof. In the development of the OBC evaluation board, the design scope was organized to attempt to satisfy all mission independent requirements.

The following figure shows the block diagram of the design. From the diagram, it is clear that the processor and memory controller together form the core of the design.

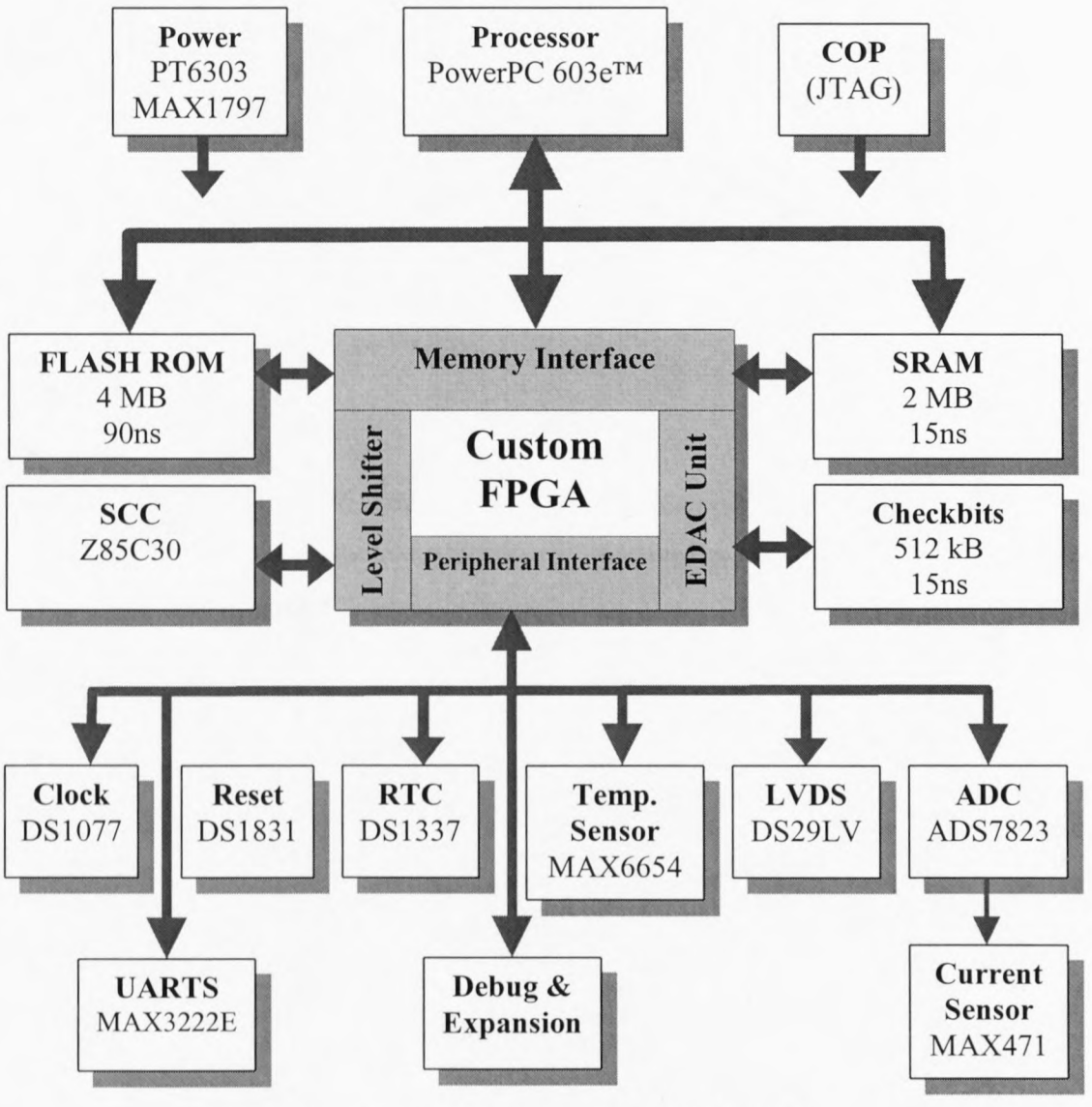


Figure 3-1. Hardware Block Diagram

In the following sections, more detail is given on the individual components and their functions.

3.2. FUNCTIONAL DESIGN

The selection of devices used to build an on-board computer has to go hand in hand with the purpose, dependability and availability of each device and of the devices with which it

interfaces. The following section motivates why the devices were used and explains their respective roles in the evaluation OBC.

3.2.1. CPU

Chapter 1 gave some background on the processor and its architecture. This chapter elaborates on its implementation. The system interface is described in the next section.

The system interface is specific for each PowerPC microprocessor implementation. The 60x family provide a versatile system interface that allows for a wide range of implementations and include a 32-bit address bus, 32/64-bit data bus, and 56 control and information signals.

3.2.1.1. System Interface

Split address/data tenuring, bus snooping, multiprocessing support, cache coherency support, and other advanced features can be used to obtain additional performance for high-performance systems, but for the purpose of a small, high-speed embedded controller, such as an OBC, many of these complications can be avoided, especially since there is only one bus master. Nevertheless, in the design of the evaluation board provision was made to make use of these features, including multiple masters, by connecting the respective signals to the support FPGA. Support for multiple bus masters permit, for example, the implementation of a DMA (Direct Memory Access) controller. The support FPGA is discussed in section 2.2.

The address and transfer attribute signals are pulled up through weak 10kΩ pull-up resistors to minimise sleep-mode power consumption. Since the 603e continually monitors these signals for snooping, a float condition may cause excessive power consumption by the input receivers on the processor. These address and transfer signals include the address bus, A[0-31], address parity, AP[0-3], transfer type, TT[0-4], transfer burst, TBSTn, and the global signal, GBLn. Similarly, the cache inhibit output signal, CIn, is also pulled up via a weak 10kΩ pull-up resistor to minimize sleep-mode power consumption. The APE

and DPE (address and data parity error) signals are open drain and are left unconnected. Their input receivers are turned off when no read operation is in progress.

The parity buses of the 603e are not used since EDAC is implemented and since the parity bus does not continue inside the processor, as is the case with the PowerPC 604™ processor. It will therefore provide no better redundancy than an EDAC system.

The CIn, GBLn, WTn, CSE[0-1], TC[0-1], DBWOn, RSRVn, TBEN and TLBISYNCn signals are not needed for a minimal design such as the evaluation board and were not connected to the memory controller or any other device. Some of these signals are pulled up via weak 10kΩ pull-up resistors to ensure initial start-up. The CSE and TC signal groups provide debug status and are connected to solder pads to make them accessible with an oscilloscope or logic analyser.

The checkstop signals are provided as an aid in problem determination. When the processor is in the checkstop state, all latches are frozen so that the state of the processor can be analysed. The two checkstop signals, CKSTP_INn and CKSTP_OUTn, are both pulled up through weak pull-up resistors. The processor will halt immediately if CKSTP_INn is not pulled up whereas CKSTP_OUTn is an open drain output and needs to pulled up to monitor its status. The three LSSD test signals are pulled up through strong 1kΩ pull-up resistors to keep noise from coupling to them.

The CLK_OUT signal is useful for debugging purposes only. It cannot be used as a clock source and is connected to a solder pad for debugging purposes.

The PLL_CFG[0-3] signals set the internal CPU frequency of operation. They are connected to the support FPGA to allow the system to dynamically alter the PLL configuration while the CPU is in reset. For completion, they are also connected to a DIP-switch to allow static configuration of the PLL.

The VOLTDETGNDn pin is internally tied to GND and is used to indicate to the power supply that a low-voltage processor is present. Since the processor is to be soldered directly on to the board with no socket mechanism, this pin was left unconnected.

3.2.1.2. Power Supply Filtering

One decoupling capacitor was placed at each Vdd and OVdd pin of the processor. This ensures a clean power supply to the 603e and prevents switching noise from the processor to reach other components on the board. 10nF capacitors are used for the decoupling of the 2.5V Vdd power rail and 100nF capacitors for the 3.3V OVdd rail. In addition, a couple of 10µF AVX TPS tantalum capacitors are used for bulk storage and quick recharging of the smaller capacitors. Only surface mount (SMT) capacitors were used to minimise lead inductance.

The PLL is powered from the Avdd power pin. The power supplied to this pin should be filtered to ensure a stable internal clock. Figure 3-2 shows the circuit used to filter the PLL power supply. The trace from the circuit to the Avdd pin was made as short as possible to minimise noise.

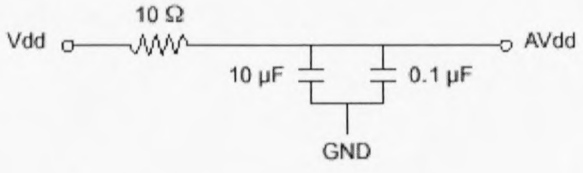


Figure 3-2. PLL Power Supply Filter Circuit

3.2.1.3. Thermal Considerations

The C4/CBGA offers the same connections in a significantly smaller board area than a comparable C4/CQFP package. The CQFP (Ceramic Quad Flat Pack) gull-wing leads require additional space for wiring fan-out for electrical routing. Thus, the CBGA package requires less board routing space than its CQFP counterpart.

This reduction of board area results in an approximate four-fold increase in package-level thermal flux density.[12] The primary heat transfer path for CBGA packages is through an attached heat sink. However, for low-power applications the use of a heat sink is not necessary. Heat conducted through the silicon may be removed to the ambient air by means of convection. In addition, a second parallel heat flow path exists by conduction. Heat flows through the C4 bumps and the epoxy underfill to the ceramic substrate for further convection cooling off the edges. Heat is also conducted from the ceramic substrate via the leads/balls to the board whereupon the primary mode of heat transfer is by convection and/or radiation. These heat transfer paths are illustrated in Figure 3-3.

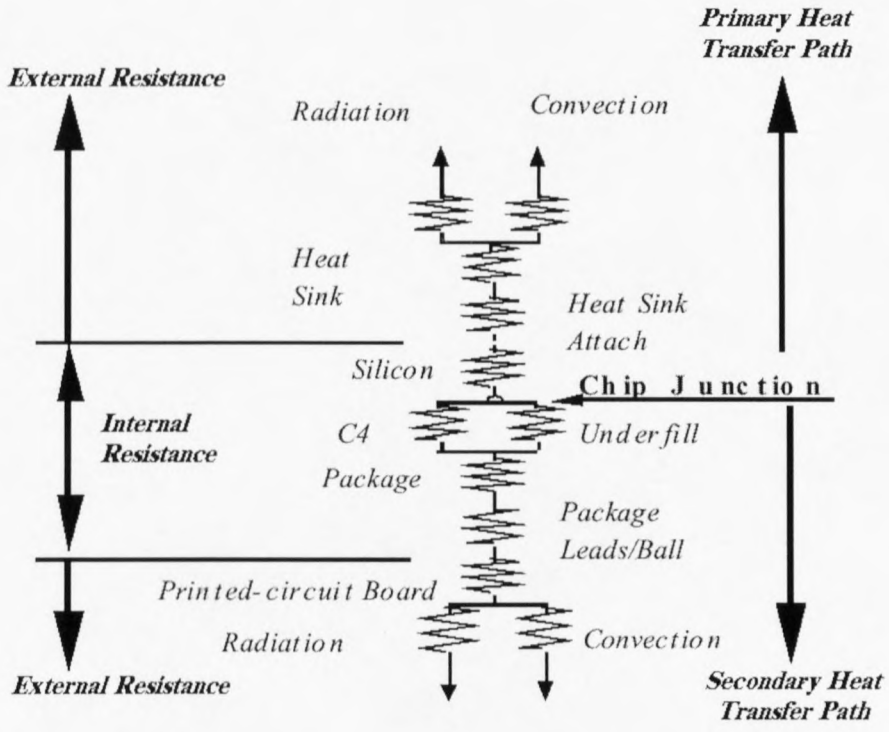


Figure 3-3. Simplified Thermal Network of a C4/CBGA Package mounted to a PCB

Since the evaluation board is not densely populated in the area surrounding the CPU, no heat sink was deemed necessary. Section 3.2.10. gives detail on the temperature sensor used on the evaluation board to aid in thermal management.

3.2.2. FPGA

The ACEX™ EP1K50 FPGA from Altera was used as the support FPGA. The ACEX family are SRAM based and combine lookup-table-logic with embedded array blocks. The EP1K50 features 50,000 gates and 40,960 bits of RAM.

Since the 603e has a 64-bit data bus, I/O pins on the FPGA was a major concern, especially because the address lines had to be connected to the FPGA as well for the EDAC system to function. The 484-pin FineLine BGA package was used to alleviate the matter, providing a maximum of 249 user-programmable I/O pins.

On a satellite OBC, radiation tolerant FPGAs will be used instead. These devices offer more reliability in the space radiation environment than their SRAM-based counterparts do. Manufacturers like ACTEL provide a vast selection of radiation tolerant ASICs and their RT54SX-S series FPGAs are specifically designed for space applications. Internal clocking speeds of up to 310MHz are achievable and they have an 8.9ns clock-to-out (pin-to-pin) delay. They compare well with the device used on the evaluation board and Actel offers equivalent devices for prototyping.

The support FPGA performs a number of functions; the primary being to control the memory space and provide an I/O interface for the PowerPC 603e™ processor. Other functions include an EDAC system for memory integrity, communications, and an interface to the onboard telemetry, e.g. temperature. The sections that follow elucidate these subsystems implemented in the FPGA.

3.2.2.1. Memory Controller

Motorola offer two PowerPC™ compliant PCI bridge/memory controller parts: the MPC106 and MPC107. Both are compatible with the PowerPC 603e™, PowerPC 740™ and PowerPC 750™ microprocessors. Both integrate secondary cache control and a memory controller that supports DRAM, SDRAM, EDO DRAM, ROM and FLASH ROM. Dual processor support and power management is also included. DMA support, I^2C, I_2O and clock drivers are added features in the MPC107. These devices offer much more than

is needed by an embedded system. Complexity can sometimes reduce performance; especially since cache coherency instructions can use valuable clock cycles.

The memory controller implemented in the FPGA is based upon the one used in the Motorola *Eximer* evaluation board. A block diagram of the original controller is shown in Figure 3-4. It provides the necessary controls for a block of RAM, ROM and access to I/O. It is not programmable by software and the memory access cycles are tuned to provide only the necessary signals.

The use of a FPGA for the memory controller also allows the designer to partition the memory space and control signals as desired. It can even be done dynamically and/or allow memory segments to overlap. A DMA controller can also be added to expedite block memory transfers and assist the CPU.

The following sections elaborate on the internal modules that form the memory controller.

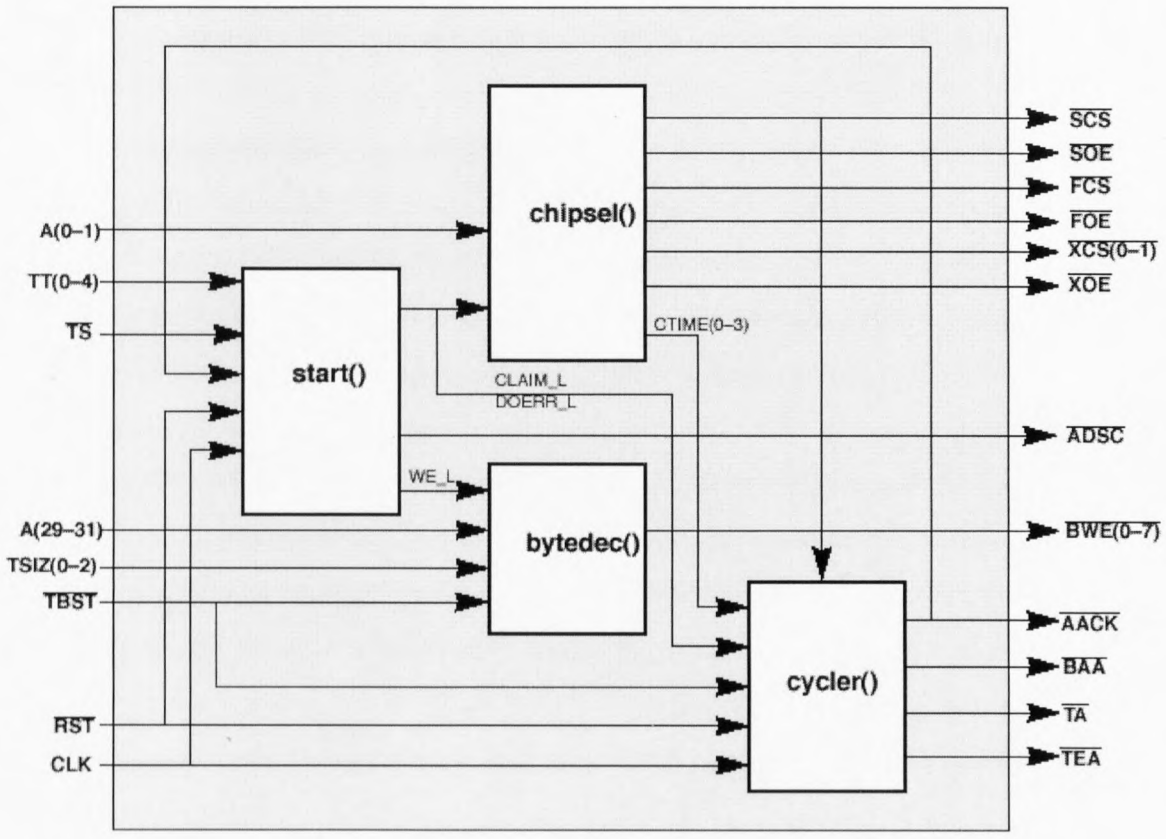


Figure 3-4. Memory Controller Architecture

3.2.2.1.1. Start Detection Module

The "start()" module provides the global signals used exclusively by other modules to detect whether a cycle is in progress or unclaimed and if its a write cycle or not. These signals remain valid until the memory cycle is completed by the memory controller by asserting the AACKn signal.

3.2.2.1.2. Byte Lane Decoder

The byte-lane write-enables, BWEn(0-7), are generated by the "bytedec()" module. The TSIZ(0-2) signals along with the lower address lines, A(29-31), are used to determine which byte lanes should be active. All lanes are enabled for burst transfers, while all other transfers enable only the necessary lanes based upon the start address and transfer size. The write status signal, decoded by the "start()" module, is examined to establish if the cycle is a write cycle or not.

3.2.2.1.3. Chip Select Unit

The higher address lines, A(0-3), are used by the "chipsel()" module to enable the corresponding memory space and to pass the proper time delay for the access to the cycler. The time delay is used to delay the assertion of TAn and thus the end of the transfer.

The evaluation board's memory space is shown in Table 3-1. The Flash ROM space was put at the top of the memory space because the PowerPC boots up by executing code located at 0xFFF0_0100, which is the exception vector address of the system reset interrupt. The SRAM is located at address 0x0000_0000 to enable the interrupt vector table to be relocated to RAM. This is accomplished by copying the table to RAM and clearing the IP bit in the MSR (Machine State Register) to change the exception prefix from 0xFFFx_xxx to 0x000x_xxx.

Table 3-1. Evaluation Board Memory Space

Start Address	End Address	R/W Size	Device
0x0000_0000	0x0FFF_FFFF	1, 2, 4 bytes	2Mb SRAM
0x1000_0000	0x1FFF_FFFF	1 byte	UART 1
0x2000_0000	0x2FFF_FFFF	1 byte	UART 2
0x3000_0000	0x3FFF_FFFF	1, 2 bytes	LVDS
0x4000_0000	0x4FFF_FFFF	1 byte	SCC Channel A
0x5000_0000	0x5FFF_FFFF	1 byte	SCC Channel B
0x6000_0000	0x6FFF_FFFF	1 byte	I^2C Controller
0x7000_0000	0x7FFF_FFFF	1, 2 bytes	System Management
0x8000_0000	0x8FFF_FFFF	1, 2 bytes	Debug port & LEDs
0xC000_0000	0xCFFF_FFFF	2 bytes	Interrupt Controller
0xF000_0000	0xFFFF_FFFF	1, 2, 4 bytes	4Mb Flash ROM

3.2.2.1.4. Cycler

The "cycler()" state machine has four flows. The first two handle SRAM single-beat and burst transfers that are typically the majority of code and data accesses. The third handles I/O and FLASH ROM transfers while the fourth is concerned with error transactions.

3.2.2.2. EDAC

For every 64-bit word, 16 extra bits are provided for the storage of EDAC checkbits. Since Hamming code require only eight checkbits for every 64-bit word protected, the 16 bits available allows for the evaluation of other EDAC schemes, e.g. Reed Solomon and BCH codes.

Hamming code provides a system with single error correction and double error detection. This might be sufficient for some aerospace applications, but when data integrity is vital for system operation and reliability, the study of other EDAC schemes are crucial. The BCH family are short block codes and have a greater distance than Hamming.

A quasi-cyclic (16,8) code capable of correcting 2 bit errors (T=2) was developed at the University of Surrey for their *SNAP* nano-satellite on-board computer.[9] The allocated 16-bits enables the development of similar codecs to improve on the single-error correction capability of Hamming code.

The address and data retry functions built into the PowerPC 603e are used to manage memory errors. When an error occurs while a read transaction is in progress, the memory controller can signal the CPU to invalidate the data read. This can be done for both the address and data buses independently and permits an effective way of interfacing with an EDAC unit. It allows for fast bus transactions and only slows the bus down in the event of an error. See Figure 3-5 for a timing diagram showing three different ways of delaying single-beat reads:

- The TAn signal can remain negated to insert wait states in clock cycles 3 and 4.
- For the second access, DBGn could have been asserted in clock cycle 6.

- In the third access, DRTRYn is asserted in clock cycle 11 to flush the previous data.

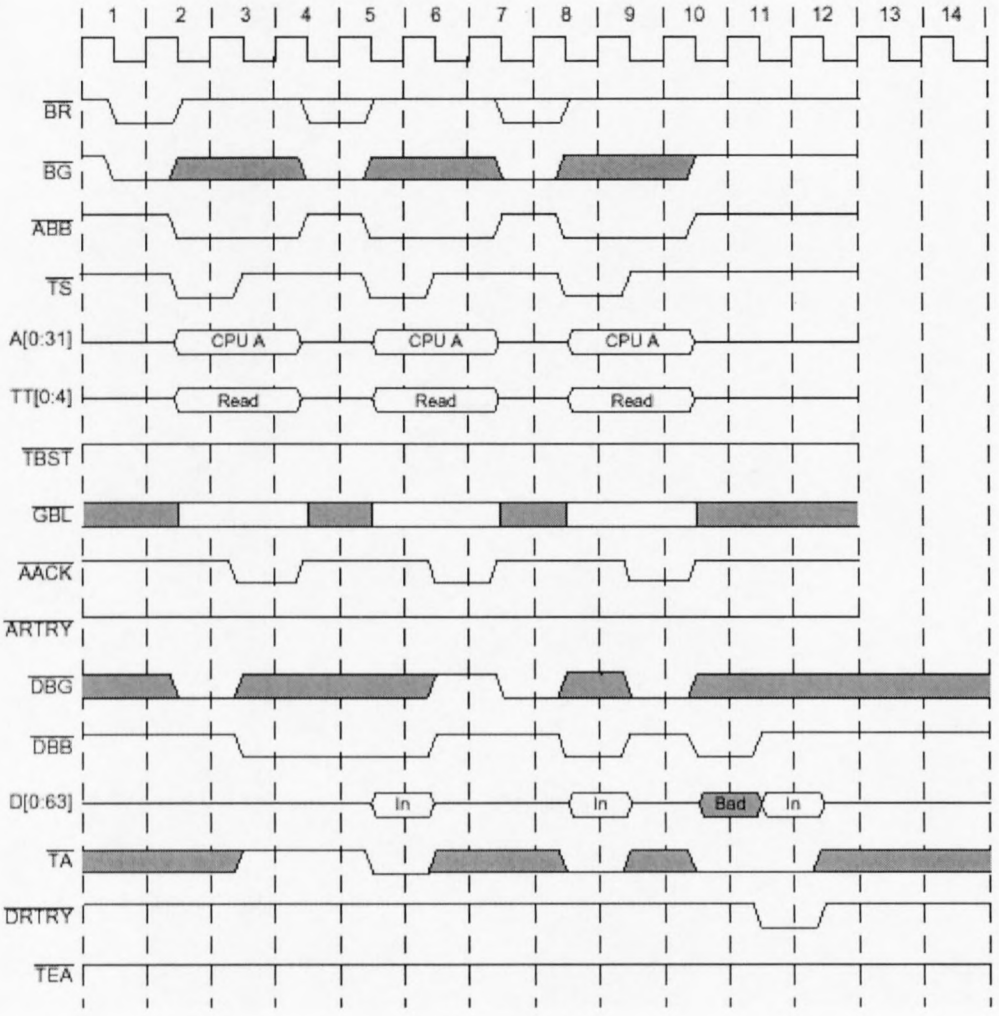


Figure 3-5. Single-Beat Reads Showing Data-Delay Controls

VHDL code was written, compiled and tested that performs Hamming code operations on a 64-bit bus using eight checkbits. The code was not implemented due to the complexity of the memory controller and the modifications necessary. Figure 3-6 shows the simulation results of the EDAC unit.

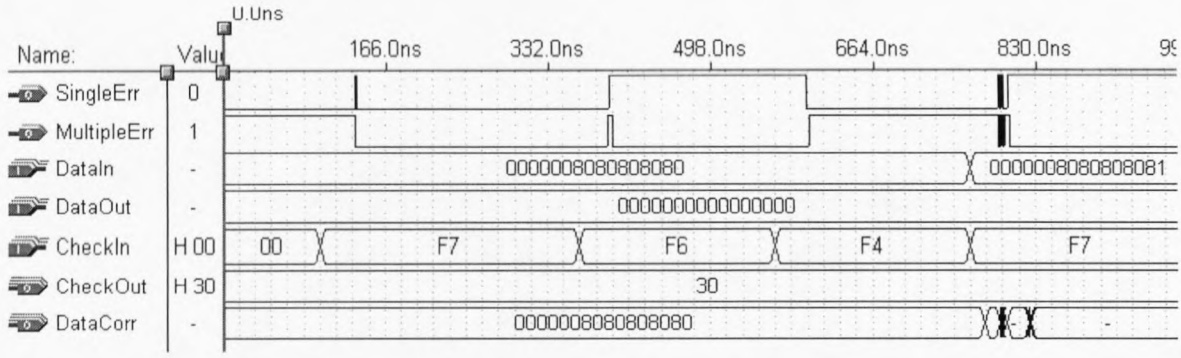


Figure 3-6. EDAC Unit Simulation Results

The maximum delay paths were obtained using the delay matrix feature of the MaxPlus II software. This maximum delay for checkbit generation (i.e. a write operation) was 26.2ns. Read operations were much slower with a maximum delay of 153.3ns.

3.2.2.3. Interrupt Controller

Implementing the interrupt controller inside the FPGA allows custom trigger settings to be programmed and thus minimises external components (glue logic).

The PowerPC 603e™ has three interrupt inputs - the MCPn, SMIn, and INTn signals. The machine check interrupt input signal, MCPn, is negative edge sensitive and initiates a machine check exception. The system management exception is initiated by the SMIn signal while the INTn signal generates an external interrupt. Both the SMIn and INTn signals are level-sensitive. The machine check exception has a higher priority than the soft reset exception as well as the other two interrupt signals and is generally used to handle bus errors, e.g. parity and/or EDAC errors. The system management interrupt and the external interrupt differ only in the signal asserted and the vector taken.

The machine check interrupt is initiated upon the detection of an EDAC error condition since it has the highest priority of the three interrupt signals. This will prevent the error from propagating to the system level. The system management interrupt is reserved for future use. The system management interrupt is used to trigger on the interrupt signal from the temperature sensor. This allows the system management software to control the tem-

perature of the processor by, for example, changing the bus clock frequency and/or changing the PLL settings or for bringing the processor out of sleep-mode.

The interrupt controller contains two registers. The one register is used to mask the individual interrupts while the other contains the current pending interrupts. The interrupt service routing (ISR) must read the latter register to determine the source of the interrupt. Both registers have one bit for each interrupt source and include bits for both UARTs, I^2C controller, and the interrupt signals from the external devices like the SCC, real-time clock, temperature sensor and LVDS (Low-Voltage Differential Signalling) channel. The switches are also routed to the interrupt controller to be used as interrupt sources.

3.2.2.4. UARTs

The two UARTs implemented in the FPGA are based on the Atmel AT90S8515 AVR RISC microcontroller's internal UARTs. They are fully independent and software programmable using the relevant registers. They support eight and nine data bits and are capable of full-duplex operation. Noise filtering is implemented in hardware by sampling the incoming channel three times and using a voting circuit to determine the valid state. They also feature overrun and framing error detection. Interrupts are generated independently on completion of a transmission, when the transmit register is empty, and upon receiving a character.

3.2.2.5. I^2C Controller

I^2C is a two-wire, bi-directional bus providing a simple and efficient method of communication over short distances between many devices. The controller core used was downloaded from the OpenCores website and is compatible with their WISHBONE architecture. This architecture is analogous to a microcomputer bus and offers a flexible integration solution that is easily tailored to a specific application.

Although the I^2C standard is a true multi-master bus, the peripherals connected on the bus are all slave devices. Since there is no need for multi-master capability, a single master controller, in which it is the master, was a good alternative. It saves space on the FPGA

and reduces the complexity of the controller itself, which, in turn, reduces the complexity of the software driver. Figure 3-7 shows the block diagram of the controller architecture.

Control of the bus is provided on both a byte and bit level using its registers. This allows the use of a single controller to interface with devices that are I^2C, SMBus, and 2-wire bus compatible. The controller generates a maskable interrupt upon completion of a task.

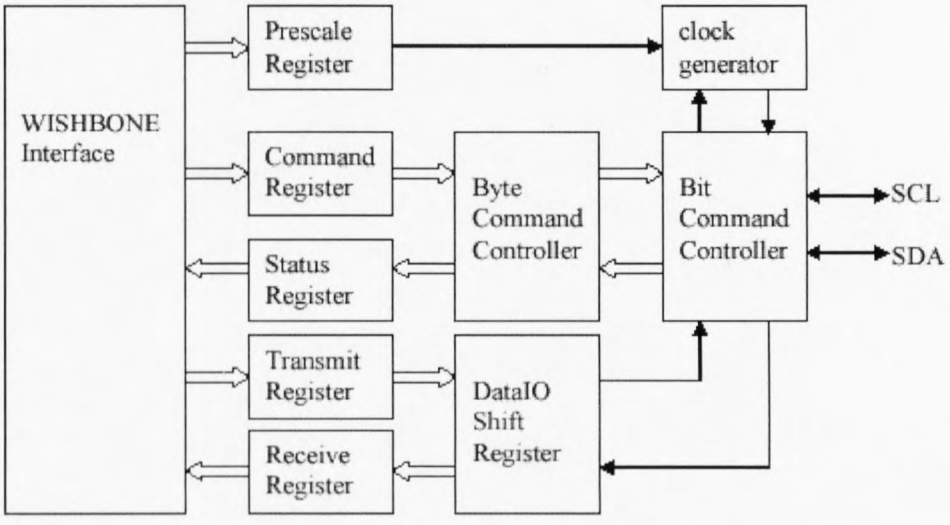


Figure 3-7. Internal Structure of I^2C Controller

3.2.2.6. Reset Controller

To ensure the correct reset of all internal circuitry, especially at power-on, a reset controller was added. It provides a reset pulse for the internal circuitry once the FPGA is configured. It also provides a reset pulse with a programmable width to the processor.

3.2.2.7. Debug & Expansion

To facilitate the addition of external devices and to aid in debugging a 20-pin connector was added. It provides 16 user configurable FPGA I/O pins, write enable line, system clock output and a ground connection. Four status LEDs and three debounced pushbutton switches are also provided.

3.2.3. FLASH ROM

FLASH ROM is used for the storage of program code, the operating system (OS) and boot-loader. It was decided to use a 64-bit wide FLASH ROM-space to allow the 603e to fetch two instructions at once and to simplify the memory controller design. The FLASH ROM used is 3.3V programmable and features a hardware sector protection mechanism where sectors can only be erased and programmed with a 12V supply.

Four Mbytes of FLASH ROM is provided by four 8-Mbit (512K x 16-bit) devices. The devices use the same software commands as E^2PROMs and have very low power requirements. They feature an automatic low power mode that is enabled when the address lines remain stable for 150ns. This feature is especially useful when the 603e is in a low power mode to ensure maximum power saving.

3.2.4. SRAM

With processor bus speeds of 66MHz the SRAM used had to be fast. Access times of 15ns are achievable with the parts used. They also only consume a maximum of 4mA each when in standby mode, i.e. when the device is disabled. Operating currents are dependant on cycle times and range from $190mA_{max}$ at 15ns to $140mA_{max}$ at 25ns (Manufacturer's maximum values). This is quite high compared to currently available 5V devices, but the speed advantage makes it well worth the extra power consumed.

Five 4-Mbit (256K x 16-bit) devices were used giving two Mbytes of RAM for use of program data storage and one 4-Mbit device for the storage of checkbits. As mentioned earlier, the memory controller is capable of implementing various EDAC schemes to for the detection and correction of errors caused by radiation.

3.2.5. Clock Generator

When the PLL is enabled, the PowerPC 603e™ can run with a bus clock frequency of between 25MHz and 75MHz. The PLL can be bypassed altogether enabling the 603e to be

run at frequencies below one MHz. This could further power reduction if the doze or sleep modes are not favourable.

To be able to dynamically change the bus clock frequency of the processor, a clock generator is used. To minimize pin usage on the FPGA, an I^2C-interface was preferred. The I^2C-bus is shared with other peripherals on the evaluation board, further reducing the amount of pins needed.
The MAX1077L 60MHz clock generator is used and it conforms to the above. It fits the clock jitter and duty cycle requirements of the 603e and features two independent synchronous outputs and a power-down mode. The power-down mode will however rarely be used since the clock generator also feeds the FPGA that controls it.

The change of the bus-clock frequency together with the ability to change the PLL configuration of the CPU allows the operating system to dynamically manage performance and power.

3.2.6. Real-Time Clock

Although the 603e's *Time Base* feature can be used to calculate time-of-day, a real-time clock was added for more flexibility and for calibration of the *Time Base.*

The DS1337U is I^2C-compatible and measures seconds up to years with leap-year compensation. It is fully register driven and features two time-of-day alarms. These can generate interrupts and can be used for general timing and/or watchdog purposes. The device uses very little power — $150\mu A_{max}$ active supply current and $2\mu A_{max}$ whilst in standby mode (Manufacturer's maximum values). The only external components needed are a 32.768kHz crystal and a pull-up resistor on the open-drain interrupt signal.

3.2.7. Reset Generator

To ensure the correct integration of reset sources and sound reset timing a reset generator was used. The critical power requirements of the PowerPC 603e™ led to the use of the

DS1831C Multi-Supply Monitor & Reset that monitors both the 3.3V and 2.5V power rails.

It monitors the 3.3V and 2.5V supplies with selectable tolerance levels and a temperature compensated voltage reference and sensor. Three external reset inputs are provided that provide manual pushbutton functionality. Reset delays are also user programmable.

A reset condition can be caused by the FPGA itself, pushbutton switch, reset circuit or the COP interface. The use of the DS1831C with its open-drain inputs/outputs greatly simplified the integration of these sources to provide a glitch-free reset pulse with an acceptable width. This is shown in Figure 3-8.

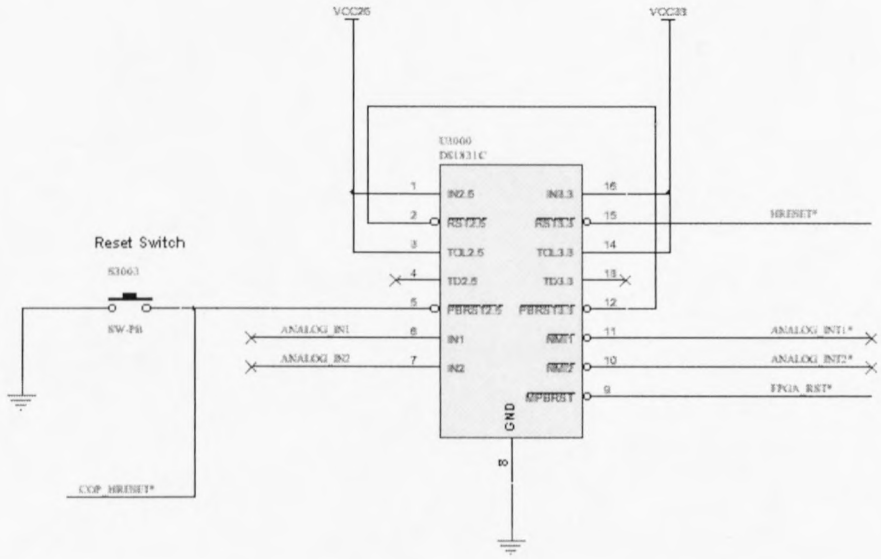


Figure 3-8: Integration of Reset Sources with Reset Controller

3.2.8. SCC

The Z85C30 from Zilog was used for its multi-protocol data communications capabilities. It handles asynchronous formats, synchronous byte-oriented protocols such as IBM Bi-sync, and synchronous bit-oriented protocols such as HDLC and IBM SDLC.

Features of the Z85C30 include two 10x19-bit status First In First Out (FIFO) buffers and DMA transfer support. It also generates and checks cyclic redundancy check (CRC) codes in any synchronous mode and can be programmed to check data integrity in various modes. Modem control signals are also provided that doubles as general-purpose I/O pins when unused.

The Z85C30 was used with great success on SUNSAT and provide a means for the development and test of communications software and modem drivers.

3.2.9. LVDS

LVDS is a low-power, low noise producing, noise tolerant technology providing a fast, failsafe data link. Putting the radiation issues of commercial LVDS transceivers aside, LVDS was found extremely suitable for aerospace applications. However, since the endorsement of the IEEE 1355 standard by ESA and major companies involved in the space industry, numerous space-qualifiable radiation-tolerant components have been manufactured.[7]

Bus LVDS (BLVDS) uses a differential transmission scheme similar to LVDS, but it is enhanced for multi-point applications. It eliminates the need for active termination devices commonly required by open-drain/collector technologies; thus simplifies backplane design and improves system reliability. Moreover, live-insertion of devices into an active bus is supported — the resulting glitch is seen as common mode noise and rejected by the receiver.

The DS92LV-series from National Semiconductor was used to exploit the abilities of BLVDS. The DS92LV1021 and DS92LV1212 are 16-40 MHz 10 bit Bus LVDS serialisers and deserialisers, respectively. The DS92LV1021 can drive a heavily loaded backplane or a point-to-point connection while the DS92LV1212 locks to the embedded clock to recover the serialised data.

3.2.10. Temperature Sensor

The PowerPC 603e™ is a powerful processor and as such can generate a substantial amount of heat when being used at its full potential. To monitor its temperature, and its temperature above the ambient temperature, a temperature sensor with local and remote sensing capability was needed.

The MAX6654 features an internal temperature sensor for the measurement of the local board temperature. It also supports an external remote temperature sensor that was used to measure the temperature of the CPU. Very little external components are needed and a low cost, easily mounted 2N3906 PNP transistor is used as the remote temperature sensor.

The MAX6654 interfaces with the FPGA via the I^2C bus and enables it to read temperature data and set alarm thresholds. Alarm threshold data is stored in four registers containing high and low temperature values for each channel. A diode fault alarm is also provided that detects open and short-circuit conditions on the remote sensor pins. All these alarms cause the ALARMn interrupt signal to be asserted and an exception to be taken by the CPU. The readout of temperature data can occur automatically or autonomously (one-shot).

These, together with the PLL-configuration and clock generator features, can be used by thermal management software to control the temperature of the CPU by either slowing it down or by forcing it to enter a low-power mode.

3.2.11. Current Sensor

The current sensor was added primarily for the dynamic detection of latchup conditions in the CPU, but also for measuring the CPU's power consumption.

The MAX471 uses an integral 35mΩ current-sense resistor and is capable of accurately measuring currents of up to ±3A. The measurement is output as a current and converted to a voltage by an external 2.0kΩ resistor giving a voltage-to-current ratio of 1:1. This voltage is then read by an analog-to-digital converter (ADC) (see section 2.2.12.) to acquire a

digital value pertaining the current drawn by the processor. The 1:1 ratio eliminates the need for conversions when measuring power $(P = V \times I)$ with an oscilloscope.

Operating current is less than 100µA over the entire temperature range and a shutdown mode, where the device draws only $18\mu A_{max}$, is also featured.

3.2.12. ADC

To convert the measurement voltage of the current sensor to a digital value usable by system management software, an ADC was needed.

Featuring 12-bit resolution, 8-channel multiplexer and an I^2C interface, the ADS7828 was elected for the task. Using the I^2C bus meant the device could be employed with no extra pin usage on the FPGA. The remaining seven channels can be used for other purposes.

Power requirements at the standard I2C interface clocking speed of 100kHz are $290\mu W_{typ}$ and $2nA_{typ}$ when in powerdown mode. To filter power-supply noise a 5Ω resistor is put in series with the 3.3V power rail and $+V_{DD}$ power pin. The filtered 3.3V supply is also fed into the external voltage reference input giving a least significant bit size of about 806µV. This should be sufficient for the detection of latchup conditions as well as the interfacing of various analog circuitry.

3.2.13. Power

The nominal power supply on a satellite is about 12-14V. This is the output voltage from the solar panels and batteries. Switch-mode regulators was used to convert this relatively high input voltage to the required low voltages because of their inherently low losses, especially when compared to that of linear regulators.

Two PT6303 switch-mode regulator modules were used for supplying the 3.3V and 2.5V rails. These parts are about 85% efficient when converting 14V to 3.3V at 1A, which was projected to be the maximum current consumption of the processor at full speed. Worst-

case ripple on the output voltage is 4% which is below the threshold of 5% for the PowerPC 603e™ processor. The 3.3V supply does not need a regulator able to supply 3A of current. However, to meet the required relation between 3.3V and 2.5V supply voltages at start-up, two similar regulators were used that feature similar start-up transient responses.

A step-up DC-DC converter was used to supply the 5V rail for the SCC. The MAX1797 part offers an excellent efficiency of 95% when converting 3.3V to 5V while supplying 100mA. Although these devices are capable of supplying currents of up to 1A, the SCC will draw currents of no more than 25mA, to which the efficiency drops to about 93%.

The PowerPC 603e™ processors' 3.3V supply must not exceed its 2.5V supply by more than 1.2V at any time, including power-on reset. Therefore, a pair of MUR420 bootstrap diodes was connected between the 3.3V supply and the 2.5V supply. These diodes have a forward voltage of 0.6V at 500mA to provide a 1.2V drop across the pair. The processor draws about 500mA when in reset.

3.3. PHYSICAL LAYOUT

The physical board layout is shown in Figure 3-9. The board was manufactured using standard FR-4 laminate.

3.4. CONCLUSION

Much consideration was given to the design of a power conservative system while also aspiring to inflict no performance penalties. The current sensor can be used by power management software to inflict a strong power/performance relationship. This can be done by regulating the processor speed or by putting the processor into a static power-down mode (doze, nap or sleep).

The peripherals were chosen for, amongst other reasons, their respective abilities to enter low-power states. By completely disabling the devices not being used, the board-level power consumption can be reduced even more. In addition, the peripherals implemented

inside the support FPGA can be disabled as well. This reduces clocking inside the FPGA and reduces its power consumption.

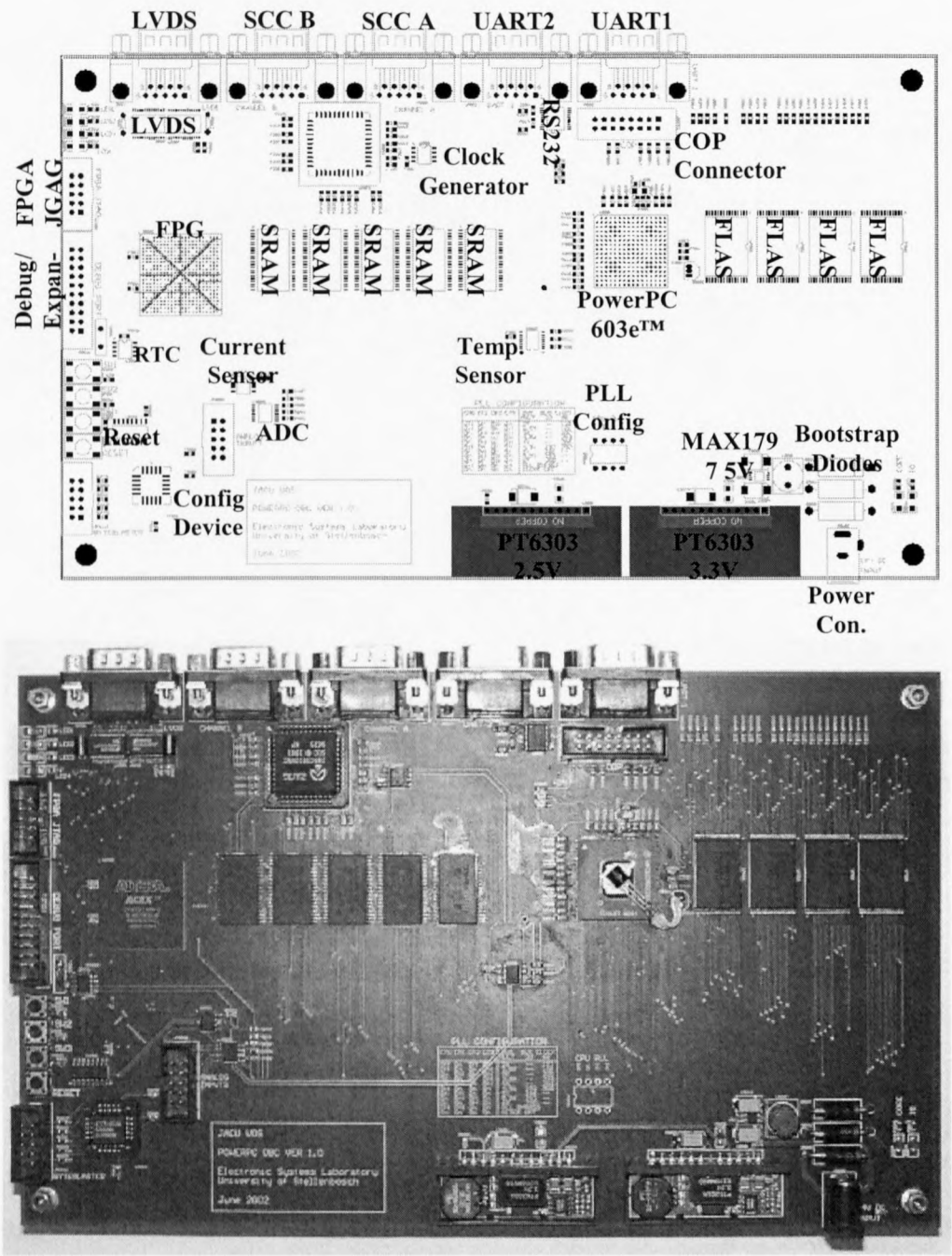


Figure 3-9. PowerPC 603e™ OBC Evaluation Board Layout

Interfacing the PowerPC processor can be made easy in an embedded system where caches are normally not used. This simplifies the memory controller since many of the control signals can be ignored or hard-wired to a desired state. The use of a custom memory controller facilitates the incorporation of mission dependant modules, like EDAC, without the use of separate devices. In this design, the memory controller performs many tasks that are all independent and easy to remove or replace. Because all the devices (peripherals) on-board (internal and external to the support FPGA) are register driven, interfacing these devices to the processor is relatively easy.

Some improvements need to be done to the memory controller module if performance is to be an issue for further development. The controller is optimised for a bus clock frequency of 66MHz and as a result, cause memory transfers to be slower than possible when used with a 30MHz clock. Registers can be included in the memory controller to make it dynamic, which is especially useful when interfacing with devices connected to the expansion port.

The next chapter contains specifics on the test software developed.

4

SOFTWARE DEVELOPMENT

Software was written to aid in the evaluation of the processor and its feasibility for use in aerospace applications. This software was also written to ease further software development on the evaluation board. The code was compiled in a Linux environment and conformed to the PowerPC EABI.

4.1. DEVELOPMENT ENVIRONMENT

Although an evaluation copy of Metrowerks' CodeWarrior Software was acquired, the GCC development environment was used for code compilation and assembly. Evaluation of the CodeWarrior software revealed that it requires an expensive debugger module to offer an eventful advantage over the GCC toolchain. The cross-compiler was compiled on a x86 platform using GCC version 2.95.3 and BINUTILS version 2.12.1. Source code for the toolchain is available on the supplementary CD.

To aid in the upload of new code, debugging and general-purpose serial I/O, a terminal was set up on a host PC. The use of a proprietary terminal program expedites software development and provides a secondary means of interaction with the evaluation board. The first being the debug and expansion port, status LEDs and switches.

4.2. THE POWERPC EABI

Code compiled for the evaluation board conformed to the PowerPC EABI.[13] This section briefly describes the EABI and its features.

The PowerPC Embedded Application Binary Interface (EABI) was created to meet the unique needs for PowerPC embedded applications, specifically minimizing memory usage while maintaining high performance. The EABI defines a set of conventions intended to afford interoperability between conforming software components (e.g., compilers, debuggers, assembly language code). These conventions are optimised for embedded applications, which typically differ from desktop applications in at least one of the following ways:

- ROM based,
- Real-time oriented,
- Memory constrained,
- Single purpose application.

The EABI is sufficiently close to the Unix System V Release 4 (SVR4) ABI for PowerPC that a single set of tools can support both ABI's and make it possible to use SVR4 functions in an EABI environment without recompilation.

Embedded programs that conform to the EABI gain efficiency in space and time by using the following features:

- Minimized stack usage,
- Relaxed alignment restrictions, optimising memory usage,
- Small data areas for RAM data, read-only data, and data around address zero. These reduce code size and improve data access time.

For running compiled programs, the EABI-specified register conventions must be followed. The EABI defines how the processor's registers are to be used by a conforming application. Table 4-1 lists the register conventions for the PowerPC EABI.

Table 4-1. PowerPC EABI Registers

REGISTER	CONTENTS
GPR1	Stack Frame Pointer
GPR2	_SDA2_BASE
GPR13	_SDA_BASE
GPR31	Local variables or environment pointer
GPR0	Volatile – may be modified during linkage
GPR3, GPR4	Volatile – used for parameter passing and return values
GPR5-GPR10	Used for parameter passing
GPR11-GPR12	Volatile – may be modified during linkage
GPR14-GPR30	Used for local variables
FPR0	Volatile register
FPR1	Volatile – used for parameter passing and return values
FPR2-FPR8	Volatile – used for parameter passing
FPR9-FPR13	Volatile registers
FPR14-FPR31	Used for local variables

The symbols _SDA_BASE and _SDA2_BASE are defined during linking. They specify the locations of the small data areas that contain part of the data of the executable. A program must load these values into GPR13 and GPR2, respectively, before accessing program data.

Before executing user code, the start-up code must also set up the stack pointer in GPR1. This pointer must be 8-byte aligned for the EABI and should point to the lowest allocated valid stack frame. The stack grows toward lower addresses, so its location should be selected so that is does not grow into data or bss areas.

4.3. PROCESSOR INITIALISATION

This section explains the boot code executed on the evaluation board. It describes the necessary code for setting up the processor to execute a user program.

At power-up, the PowerPC 603e™ processor is in a minimal state with most features and interrupts disabled. It boots up in big-endian mode and executes code beginning at 0xFFF0_000. This code, located at the system reset vector, must handle system initialisation. A typical initialisation sequence performs the necessary setup or hardware-specific initialisation, and then enables exceptions.

If memory management features, such as memory address translation, are required, the boot program will need to set up the MMU. For a minimal system like the evaluation board, it is sufficient to use block address translation (BAT) to perform a rudimentary mapping. For more complex systems, the segment registers and page tables need also be initialised. There are eight BAT array entries allowing the programmer to specify options such as whether the specified address range is valid for supervisor or user mode, the memory/cache access mode, and the protection bits for the block. Four of these map data regions (DBATs), while the remaining four entries specify instruction regions (IBATs). Each entry consists of two registers called the upper and lower BAT entries. The procedure for initialising a pair of BAT registers is as follows:

1. Disable the MMU.
2. If modifying a data BAT, execute an **isync** instruction.
3. Initialise the lower portion of the BAT array entry.
4. Initialise the upper portion of the BAT array entry.
5. Execute an **isync** instruction.
6. Re-enable the MMU when all setup is complete.

In a complete operating system, MMU setup continues with invalidating TLB entries, initialising the segment registers, and setting up the page table. Even if only BAT mappings are used for translation, it is possible that a user program will generate accesses to addresses that are invalid or not mapped by the BAT registers. In this case, the processor hardware attempts to look at the page table to resolve the reference. For this reason, the page table need to be initialised to prevent them from containing random data and cause unintended memory accesses.

When the MMU setup completes, the MMU may be enabled. At this point, address translation is active. The instruction and data caches can be enabled at this point, but since this is the evaluation of an on-board computer where caches are normally disabled due to the radiation environment, they will be kept disabled.

Section 3.1.1. briefly described the PowerPC EABI. Much of the required EABI register setup is accomplished through a call to __eabi(). The user does not call this function directly; a call to __eabi() is inserted at the beginning of main() in the user program by the compiler. Programs that lack a main() should call __eabi() before executing any user code.

Most compile environments provide an __eabi() function that is automatically linked with user programs. The __eabi() startup code is responsible for setting up the registers specified by the EABI. A simple __eabi() is used that initialises registers GPR2 and GPR13. GPR1 is initialised prior to the call to main() by the init sequence.

4.4. SOFTWARE INTERFACING

This section focuses on how software interfaces with the devices on the evaluation board. This is necessary for future development on the board and to test the interfacing capabilities of the processor as well as that of the support FPGA. See Appendix A for the bit-assignments of the registers implemented in the FPGA.

4.4.1. UARTs

The two identical UARTs are implemented inside the support FPGA with only the MAX3222E device located externally. They are register driven with interrupts generated upon completion of the transmission and/or reception of a byte and when the transmit register is empty. These registers are the following:

- The UART Data Register (UDR) is written with a new byte to be transmitted and is read to read the current received byte.
- The UART Status Register (USR) contains status information bits for the control of transmission and reception. Completion of transmission

and reception and if the UDR register is empty (for transmission) are used to control data flow and are the sources of the UART interrupts. Error status bits, including framing and parity errors, are also included in this register. This register is read-only.

- Enabling of the UARTs, interrupts and the selection of 8-or 9-bit character mode are done via the UART Control Register (UCR). It also contains the ninth bit to be transmitted and the ninth bit currently received. Both read and write operations can be performed on this register.
- The baud rate divisor is located in the UART Baud Rate Register (UBRR). This 8-bit value is computed using the following equation:

$$UBRR = \left(\frac{f_{sysclock}}{16 \times BAUD} \right) - 1$$

The factor 16 is used since the core uses a clock equal to 16 times the baud clock frequency.

The driver implemented did not make use of the interrupt capability of the UARTs. Instead, the status register, USR, was polled to establish completion of transactions. Table 4-2 lists the registers of UART1. UART1 is located at base memory location 0x1000_0000 and UART2 at 0x2000_0000. The register addresses for UART2 are the same as that for UART1 with the base address location moved to 0x2000_0000.

Table 4-2. UART Registers

NAME	ADDRESS	ACCESS	DESCRIPTION
UART1_UBRR	0x1000_0090	R/W	Baud Rate Divisor Register
UART1_UCR	0x1000_00A0	R/W	Control Register
UART1_USR	0x1000_00B0	R	Status Register
UART1_UDR	0x1000_00C0	W	Transmit Register
UART1_UDR	0x1000_00C0	R	Receive Register

4.4.2. I²C Controller

As mentioned in section 3.2.2.4. the I²C controller offers control on both a byte and bit level. This was necessary since the devices on the I²C bus did not use the same protocol and allowed the use of SM-bus (System Management Bus) compatible devices. Since the SM-bus protocol is inherently similar to I²C, they may be used on the same bus. Because of these differences between the devices on the I²C bus, separate drivers had to be written for each device.

The I²C controller is fully register driven and as such is easy to set up. The devices connected on the I²C bus can all operate at a frequency of 100kHz or less. This frequency is set by writing a clock divider value into one of its registers. The other registers contain command, status and control information.

The PRESCALE register is a 16-bit register and is accessed via two 8-bit registers: PRERhi and PRERlo. The prescale value is used to set the SCL clock frequency. Internally, the core uses clock equal to five times the frequency of the SCL clock. Thus, the prescale value is calculated as:

$$prescale = \left(\frac{f_{sysclock}}{5 \times f_{SCL}} \right) - 1$$

The control register is used to enable the core and its interrupt signal, while commands are issued via the control register. These commands include the output of start and stop conditions and the acknowledge bit to be sent.

The status information contained within the status register consists of the currently received acknowledge bit, whether the I²C bus is busy, and if there is currently a transfer operation in progress. The interrupt flag is also available in the status register.

The memory locations of the I2C controller registers are listed in Table 4-2.

Table 4-3. I2C Controller Registers

NAME	ADDRESS	ACCESS	DESCRIPTION
PRERlo	0x6000_0000	R/W	Clock Prescale Register lo-byte
PRERhi	0x6000_0010	R/W	Clock Prescale Register hi-byte
CTR	0x6000_0020	R/W	Control Register
TXR	0x6000_0030	W	Transmit Register
RXR	0x6000_0030	R	Receive Register
CR	0x6000_0040	W	Command Register
SR	0x6000_0040	R	Status Register

4.4.3. Real-Time Clock

To read data from the real-time clock (RTC), its register pointer has to be programmed with the address of the register to be read. Data is then read from that register and subsequent registers on following read operations. Since the time data, seconds up to years, are stored in subsequent registers, the register pointer was set up only once prior to reading. Consecutive reads are then performed to obtain the complete time data from the device.

Because the RTC is located on the I^2C bus, all read and write operations are made via the I^2C controller. The flow diagram of Figure 4-2 illustrates the procedure executed when reading the time information from the RTC. The writing of time information is done in similar fashion.

4.4.4. Temperature Sensor

Although the temperature sensor is also situated on the I^2C bus, the modus operandi is a bit dissimilar. To read temperature data from it, a command byte has to be sent first. This command byte can be the address of a register to read from or write to, or it can be a command to, for example, order a one-shot conversion. Thus, the register pointer of the temperature sensor is not automatically incremented as is the case with the RTC and has to be written to before any read or write operation can be performed.

High and low temperature alarm thresholds can be set for both temperature channels. These are controlled by four registers and can trigger an interrupt when either threshold is exceeded. The control register is written to enable the interrupt signal and initiate standby mode. The parasitic resistance cancellation feature, used to minimise the effect of diode resistance on temperature measurement accuracy, is also enabled via the control register. The status register shows whether a conversion is in progress and if the various temperature thresholds have been exceeded or not. It also shows if there is an open circuit condition in the external sense diode.

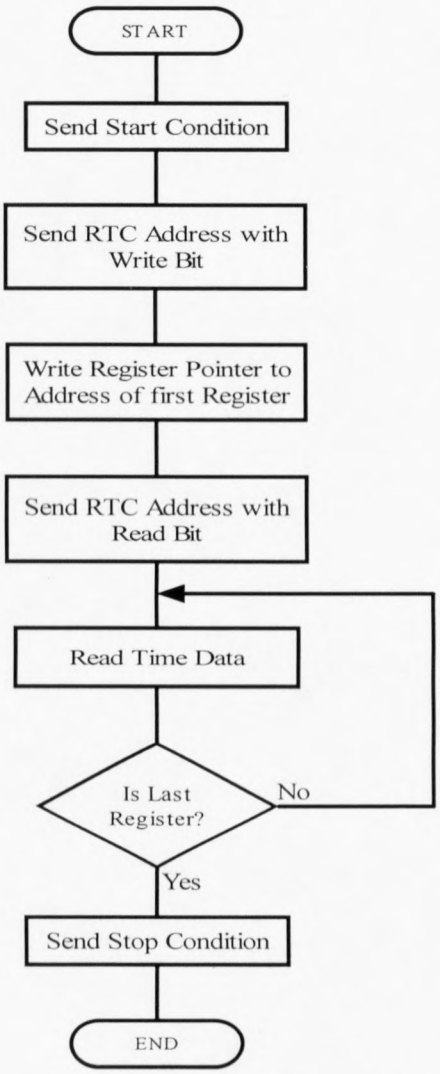


Figure 4-1. Flow Diagram of RTC Read Procedure

4.4.5. Clock Generator

The clock generator used allows the system to dynamically change the bus and/or LVDS clock frequency without clock jitter or glitches while still maintaining the processor's internal state. The LVDS and bus clocks are connected to the two independent outputs of the generator allowing fully independent operation. The LVDS clock can even by disabled while the system clock keeps running.

Changing the system clock frequency implies many complications, the biggest being the processor PLL. Since the PLL has to relock to the new system clock frequency, it is impossible to use the processor to alter its own system clock frequency. The processor has to be put in sleep mode or held in hard reset while changing the frequency (and or PLL configuration) and while the PLL locks to the new frequency. It is the responsibility of the support FPGA to change the bus clock frequency when told to do so after the system has entered a quiescent state.

The memory controller has to adjust its timing to the new frequency to optimise memory transfers. In addition, if software uses the time-base facility of the processor it has to consider the new clock frequency - especially if the Time Base is used to compute time-of-day. Further, all clock divisors, e.g. those of the UARTs and I^2C controller, have to be attuned to the new frequency too.

No software was written to change the system clock frequency. However, alterations to the PLL configuration were made dynamically while the processor was in sleep mode.

4.4.6. Interrupt Controller

The Interrupt controller's registers provide a means of masking and identifying interrupts. These two registers are shown in Table 4-4. INT_REG contains information on currently pending interrupts and identifies the source of the external interrupt exception. The other register, INT_MSK, is used to mask off interrupt signals not being used. Both these registers are 32 bits wide.

Table 4-4. Interrupt Controller Registers

NAME	ADDRESS	ACCESS	DESCRIPTION
INT_REG	0xC000_0000	R	Interrupts Pending Register
INT_MSK	0xC000_0010	R/W	Interrupt Mask Register

In INT_REG a logical "1" implies a pending interrupt whereas in INT_MSK it implies that the corresponding interrupt is enabled. All interrupts are enabled through the INT_MSK register, including the interrupt signals from the switches and external I/O port.

4.4.7. LVDS

The LVDS channel (transmit and receive) is memory mapped to location 0x3000_0000. This 16-bit memory space is connected to the 10-bit LVDS interface with the six most significant bits masked off. Enabling the transceivers is done by setting corresponding bits in the LVDS_CTRL register. This register is also used to activate synchronisation mode in the serialiser. The LVDS_STATUS register can be read to obtain the current lock status. The lock bit is set when the deserialiser has locked to the incoming data stream. This bit must be used by the synchronisation software to maintain the lock. Table 4-5 lists the registers used for LVDS transactions.

Table 4-5. LVDS Registers

NAME	ADDRESS	ACCESS	DESCRIPTION
TXR	0x3000_0000	W	Transmit Register
RXR	0x3000_0000	R	Receive Register
CTRL	0x3000_0010	W	Control Register
STATUS	0x3000_0020	R	Status Register

The LVDS clock is supplied directly from the clock generator's second output. Any change in this frequency is done by changing the clock divider in the clock generator. This frequency must be between 16MHz and 40MHz for the parts used.

4.4.8. Serial Communications Controller

The Zilog serial communications controller (SCC) used features two independent multi-protocol communications channels. These channels are mapped to two separate memory locations: channel "A" to 0x4000_0000 and channel "B" to 0x5000_0000. There are 16 registers programmed separately to configure the channels, and 10 registers providing status information. These registers, also called control registers, are accessed indirectly by first writing the address of the desired register into WR0 (Write Register 0). The following read or write operation is then performed on that register. The data registers (transmit and receive registers) for both channels can be accessed in similar fashion, or directly by pulling the D/C line high.

No software routines were written to interface with the SCC due to its complexity and its limited use in the evaluation of the processor.

4.4.9. System Management

Most of the components used on the evaluation board have the capability to enter a low-power state. The driver IC for the UARTs can, for example, have its transmitters and/or receivers disabled. It also features a power-down mode to minimise power consumption. These features together with the various power saving features of the other devices are activated via the system management register.

Table 4-6. System Management Registers

NAME	ADDRESS	ACCESS	DESCRIPTION
SMR	0x7000_0000	R/W	System Management Register

4.4.10. Debug & Expansion

The 16-pin expansion connector was discussed in section 3.2.2.6. These pins are connected to the FPGA and can be configured in a number of ways to suit almost any external interfacing requirement. For general I/O support the WEn signal is hard-wired to the port.

The port is memory mapped to address 0x8000_0000 via a read-write register. The pins of the port can also be used as interrupt sources by enabling the corresponding bits in the interrupt mask register. The direction of each pin, input or output, is set via the direction register. Also mapped to this address are a read-only register representing the status of the switches and a write-only register for status LED output. These registers are listed in Table 4-7.

Table 4-7. Debug/Expansion Registers

NAME	ADDRESS	ACCESS	DESCRIPTION
PORTREG	0x8000_0000	R/W	Port Register
DIRREG	0x8000_0010	R/W	Direction Register
LEDREG	0x8000_0020	R/W	Status LED Register
SWREG	0x8000_0030	R	Switch Status Register

4.5. CODE UPLOAD

This section will firstly explain the VHDL code written to upload code to the evaluation board. The source code is available on the supplementary CD.

Since the FLASH ROM used contain a bottom boot block located exactly where the processor starts executing from a hard reset condition, no removable FLASH devices were used. The bootloader can be loaded into the boot sector and transfer control to an uploaded user program. This program can reside in the remaining sectors of the FLASH devices, or in SRAM. By programming the interrupt vectors in the bottom area of the SRAM and clearing the IP bit in the MSR, the interrupt vector table is transferred from the FLASH ROM to SRAM, thus enabling the user code to execute custom interrupt handlers without modifying that of the boot software.

Motorola developed the S-Record file format to provide a humanly readable text representation of the binary equivalent 'hex'-file. This format proved to be lucrative in that it al-

lowed the use of proprietary software for the serial transmission of the file. It also made the inspection of memory dumps much easier.

The block diagram of the programmer is shown in Figure 4-2. The S-record is sent serially to the FPGA that sends the bytes to the S-record decoder to retrieve address and data values. The state machine resides in the top-level design file and orchestrates the programming of the SRAM by toggling the chip-select and write enable lines and enabling the bus drivers to output the address and data on the bus.

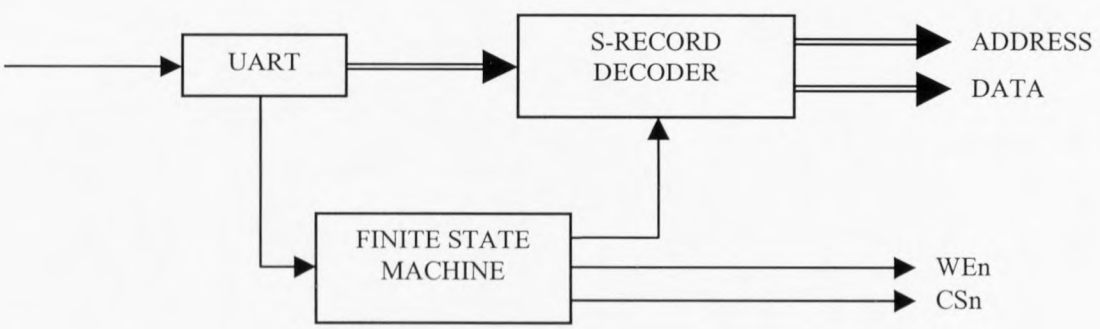


Figure 4-2. Block Diagram of S-Record Programmer

4.6. DEMO PROGRAM

This section discusses the demo program executed on the evaluation board that initialises the processor and some of the peripherals. Time, temperature and current consumption data are output on one of the serial ports. The source code is available on the supplementary CD.

The first part of the code initialises the processor as described in section 3.3. To gain access to the PC's serial port (and terminal program), UART1 is initialised. Then, the I^2C controller is set up for the retrieval of the temperature and time information. The interrupt controller is also initialised to allow the switches to trigger an external interrupt exception. Finally, the program enters an endless loop and continuously outputs the time and temperature data via the serial connection to the PC.

To achieve the chosen baud rate of 38.4kbps with the system clock frequency of 30 MHz the UBRR register of UART1 was loaded with the value 0x30H. Then, the UCR register was programmed to enable the transmitter and receiver of the UART. It was programmed with the value 0xC0H.

The I^2C is also initialised to gain access to the RTC, temperature sensor and ADC. The slowest device on the I^2C bus is the temperature sensor capable of running at a maximum SCL frequency of 100kHz. The I^2C bus frequency was programmed at 100kHz by programming the prescale register with the value 0x3C. The I^2C core is then enabled via the I2C_CTR register.

To enable the test of interrupts, the interrupt controller was initialised and programmed to enable interrupts from the three pushbutton switches.

The program now enters an endless loop where data is gathered from the RTC, the temperature sensor's two channels and the ADC to read the current sensor output. This data is output on UART1 by means of polling.

When a pushbutton switch is pressed, the interrupt routine is executed that reads the INTPEND register to get the source of the interrupt and outputs it on UART1.

4.7. SRAM DUMP

To aid in hardware debugging of the evaluation board VHDL code was written to dump the SRAM contents to UART1. It also allows the software designer to verify that a previous code upload procedure was executed successfully and to evaluate the data and stack segments. It can also be used to validate memory coherency while performing radiation tests or to validate the correct operation of the EDAC unit. The source code is available on the supplementary CD.

4.8. Conclusion

Software drivers were written and successfully executed for the UARTs, RTC, and temperature sensor. Software drivers still need to be written for the SCC and LVDS devices.

The LVDS driver will require some hardware assistance. A DMA controller will need to be added to the support FPGA and memory controller module to facilitate the fast data transfers. If a DMA controller is not favourable, some other means of interaction with the memory controller is unavoidable.

5

TEST AND MEASUREMENT

Power consumption and temperature are the main criteria when evaluating an OBC processor. Temperature measurements were made by fixing the external temperature sense diode on the processor die with heatsink paste. The temperature sensor has a tolerance of ±1°C. The on-board current sensor together with the ADC was used for the current measurements. All measurements were made with caches disabled, floating point unit enabled and dynamic power management enabled.

5.1. Processor Temperature

The following table shows the maximum temperatures measured at an ambient temperature of 21±1°C. The processor bus clock frequency was set at 30MHz while the PLL setting was changed for different processor clock frequencies. Measurements were made 30 minutes after power-up for satisfactory results, i.e. when the temperatures have stabilised.

Table 5-1. Temperature Measurements

Speed	Max CPU Temp. [°C][1]	Max Board Temp. [°C][1]
200MHz	49	29
150MHz	43	28
30MHz	25	23

1. Allow tolerance of ±1°C

5.2. POWER CONSUMPTION

Current drawn by the processor core on the 2.5V supply ranged from about 470mA at 200MHz to 400mA at 150MHz and dropped to about 90mA at 30MHz. The latter measurement was made by bypassing the PLL and running the processor at the bus clock frequency. When internal clocking of the processor was turned off (PLL disabled) the current drawn was about 30mA (75mW). This is also the case when the processor is in sleep mode and the PLL and internal clocking is turned off. The 3.3V I/O current was not measured, as it is implementation dependant, i.e. different boards will have different characteristics.

5.3. POWER-UP CURVES

The following three figures show the processor and board temperatures and processor current curves from power-up. Exponential approximations for the temperature curves and linear averages for the current curves are also shown. Allow a tolerance of ±1°C for all graphs.

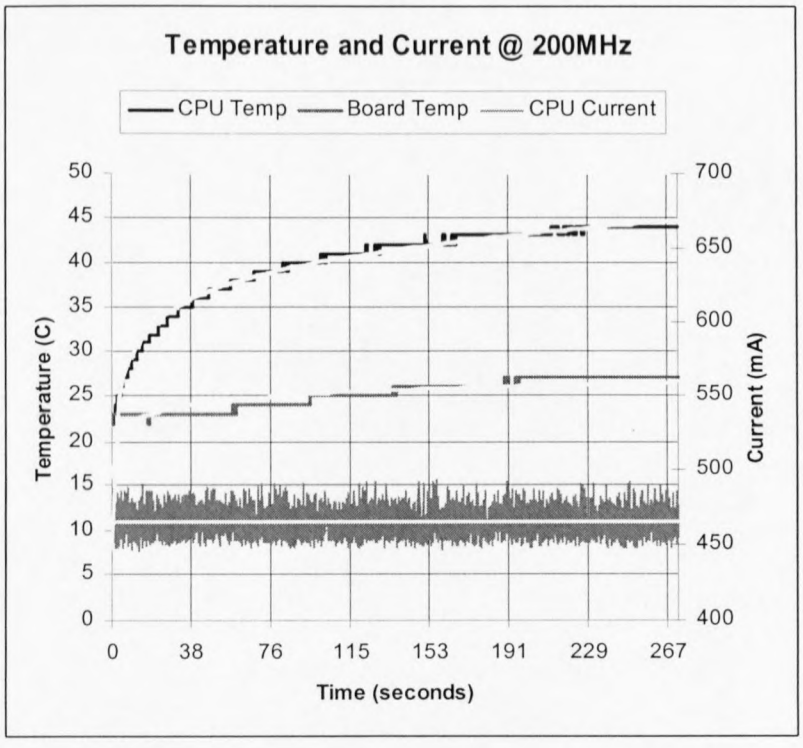


Figure 5-1. Temperature and Current Curve at 200MHz

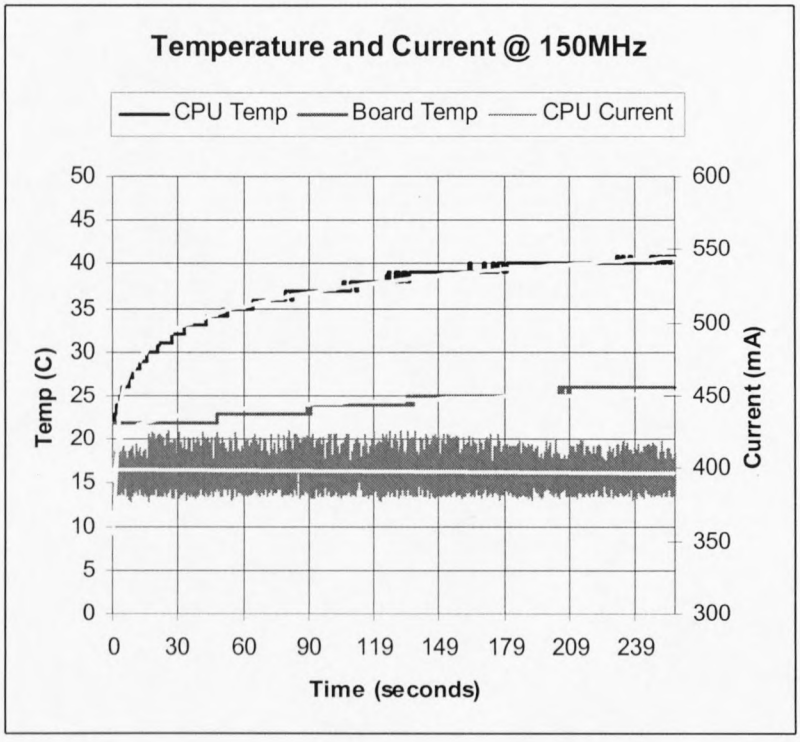


Figure 5-2. Temperature and Current Curve at 150MHz

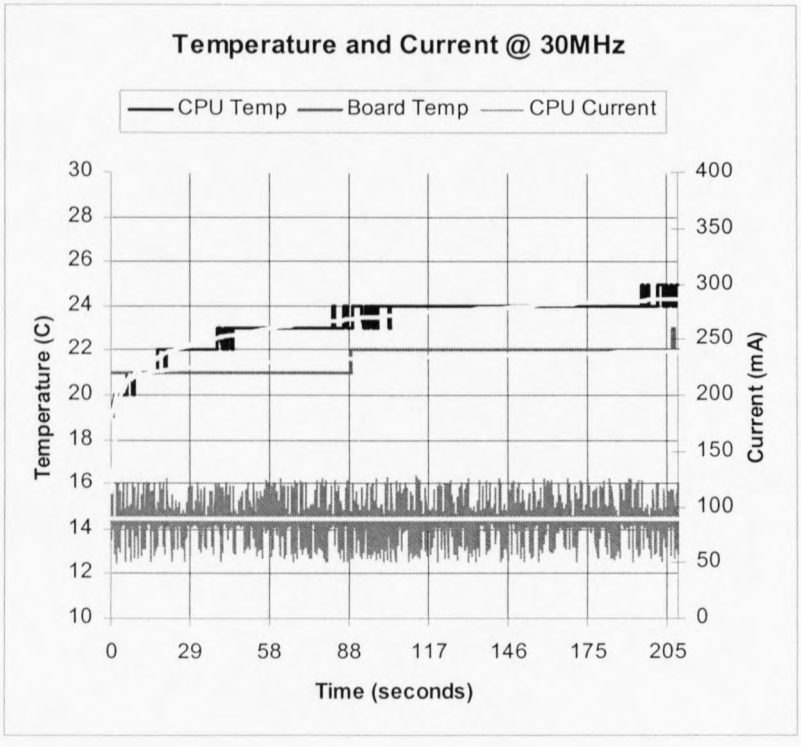


Figure 5-3. Temperature and Current Curve at 30MHz

From the figures, it is clear that the processor temperature is a function of its internal clocking frequency. It is also clear that a substantial amount of heat is transferred to the PCB which means that thermal layers will provide an effective means of heat transfer to the outside of the satellite.

5.4. SRAM Test

To aid in hardware debugging of the evaluation board VHDL code was written to perform a series of write and read operations on the SRAM. A test pattern is written to an incrementing address and subsequently read back for verification. When the cycle is completed, the test pattern is inverted and the process is repeated. The system is capable of testing (read and write) SRAM at speeds of up to 89Mb/s with a bus clock frequency of 66MHz. The source code is available on the supplementary CD.

5.5. Conclusion

At 40°C the processor's temperature stayed well within the specified maximum die-junction of 105°C and did not require a heat sink. The caches of the PowerPC 603e™ takes up just over half of the total transistor count of the processor. If they are enabled, current consumption will definitely increase and more heat will be generated. Heat sinks or any other heat transfer path, e.g. heat pipes, other than that of the PCB and/or thermal layers may then be needed. The temperature curves show that much of the heat is transferred to the PCB. This implies that thermal layers will be an effective means of heat dissapation.

The processor consumes a fair amount of power that can be reduced by reducing the clocking speed. The power consumption is further reduced dramatically by putting the processor in sleep mode and disabling the PLL.

6

CONCLUSIONS AND RECOMMENDATIONS

The COTS PowerPC device offers a clear path for performance upgrades because its commercial vendors leverage their investment in improved designs by maintaining backward compatibility for many generations.[2] For example, the PowerPC 750™ series is fully bus compatible with the PowerPC 603™ series. This implies easy implementation and integration of new PowerPC processors into existing designs as they are introduced. It greatly reduces development time and cuts down on development costs.

The PowerPC family of processors are inherently immune to SEL with estimated life spans of 15 years in a low-earth orbit. With expected "hang" rates of once every 3 days (with caches enabled) their SEU sensitivity may limit their use to commercial and/or LEO applications.

The mechanical reliability of the high density CBGA package can be improved by investigating new PCB laminate materials that more closely match the TCE of these high thermal mass devices. It is recommended that Thermount® be used for high-reliability CBGA applications. The use of BGA-to-PGA adapters also increases the thermal cyclability of these devices.

For future development it is recommended that the internal modules of the support FPGA be clocked by the LVDS clock. The LVDS devices are able to operate at 16MHz to

40MHz. Clocking the support FPGA at a fixed frequency of, for example, 33MHz allow the timing of internal modules to be independent of the bus clock frequency. The processor bus clock frequency can then be changed without any adverse effect on the functioning of, for example, the UART baud rates.

The EDAC system still needs to be implemented in hardware. To make memory transfers as fast as possible the effect of the EDAC delay in read operations need to be reduced. The PowerPC architecture provides the data retry feature and can be utilised as follows:

1. When the processor requires data from memory the uncorrected data is fed directly from the memory to the processor while the EDAC unit is busy checking its integrity.
2. In the following cycle, if a correctable error was detected, the DRTRYn signal is asserted and the corrected data presented to both the processor and memory. If no error was detected the processor uses the data previously acquired.
3. Upon the event of an uncorrectable error, the MCPn signal should be asserted to trigger a machine check exception. The exception routine should determine the recoverability of the error and generate a system reset via the support FPGA if the error is fatal.

The data retry feature allows effective error correction without compromising performance.

Memory bandwith can be increased radically with the use of synchronous SRAM. The processor and memory controller both support burst transfers, but only when caches are enabled. It must therefore be decided on a system level whether burst transfers, and caches, are required or not. Caches increase performance, but dramatically decrease the SEE performance of the processor.

Among currently available commercial processors, the PowerPC family ranks high in MIPS per watt, but its suitability for space applications outside low-earth orbits may be limited by the radiation environment, particularly single event effects.

REFERENCES

1. Poivey, C., "Use of commercial Components in Space Systems - Radiation Aspects", Technical Event, 4th European Conference on Radiation and their Effects on Components and Systems (RADECS 97), Cannes, France, September 15, 1997.
2. Lathi, D., "ISC (Integrated Spacecraft Computer) Case Study of a Proven, Viable Aproach to Using COTS in Spaceborne Computer Systems", 14th Annual/USU Conference on Small Satellites, 2000.
3. McDonald, P.T., "PC603E 32-bit RISC µP Radiation Effects Study", Innovative Concepts Inc., White Paper, 1999.
4. Irom, F., "Single Event Upset in Commercial Silicon-on-Insulator PowerPC Microprocessors", Jet Propulsion Laboratory/California Institure of Technology, July 16, 2002.
5. Swift, G.M., "Backside Device Irradiation for Single Event Upset Tests of Advanced Devices", Jet Propulsion Laboratory/California Institute of Technology, SEE Symposium, Session E: Testing and Facility, April 25, 2002.
6. Ledebuhr, A.G., "Autonomous, Agile, Micro_Satellites and Supporting Technolo-gies for Use in Low-Earth Orbit Missions", 12th AIAA/USU Conference on Small Satellites, 1998.
7. Gary, S., "The PowerPC™ 603 Microprocessor: A Low-Power Design for Portable Applications",
8. Dr. Parks, S.M., "High-Speed, Low-Power, Excellent EMC: LVDS for On-Board Data Handling", Applied Computing, University of Dundee, Dundee, Scotland, UK.

9. Wicker, S. and Bhargava, V., "Reed-Solomon codes and their applications", IEEE Press, 1994.
10. Hodgart, M.S. and Tiggeler, H.A.B., "A (16,8) Error Correcting Code (T=2) For Critical Memory Applications", Surrey Space Centre, University of Surrey, UK.
11. Kromann, G.B., "Motorola's PowerPC 603™ and PowerPC 604™ RISC Microprocessor: the C4/Ceramic-ball-grid Array Interconnect Technology", Motorola Advanced Packaging Technology, Semiconductor Products Sector, Austin, Texas.
12. Gerke, R.D., "The Effect of Solder-Joint Temperature Rise on Ceramic-Ball-Grid Array to Board Interconnection Reliability: The Motorola PowerPC 603™ and PowerPC 604™ Microprocessors and MPC105 Bridge/Memory Controller", Motorola Advanced Packaging Technology, Semiconductor Products Sector, Austin, Texas.
13. Kromann, G.B., "Thermal Management of a C4/Ceramic-Ball-Grid Array: The Motorola PowerPC 603™ and PowerPC 604™ RISC Microprocessors", Motorola Advanced Packaging Technology, Semiconductor Products Sector, Austin, Texas.
14. Parry, J., "The Development of Component-Level Thermal Compact Models of a C4/CBGA Interconnect Technology: The Motorola PowerPC 603™ and PowerPC 604™ RISC Microprocessors", Flomerics Group plc, Hampton Court, England.
15. Sobek, S., "PowerPC Embedded Application Binary Interface (EABI): 32-Bit Implementation", Microcontroller Technologies Group, Motorola, Austin, Texas.
16. "MPC603e RISC Microprocessor User's Manual Rev. 2.0", Motorola Literature Distribution, Denver, Colorado, August 2001.
17. "Errata to MPC603e RISC Microprocessor User's Manual Rev. 2.0", Motorola Literature Distribution, Denver, Colorado, January 2002.
18. "PowerPC™ Microprocessor Family: The Programming Environments For 32-Bit Microprocessors Rev. 1.0", Motorola Inc., January 1997.

APPENDIX A : REGISTER BIT ASSIGNMENTS

There are several registers implemented in the support FPGA that are used for control, status and data transactions. Their bit-assignments follow.[i]

A1. UART REGISTERS

Each UART contains four registers of which two are bit-assigned. They are the Control and Status registers.

Table A1-1. UART Control Register

BIT	ACESS	DESCRIPTION
0	R/W	RXCIE, RX Complete Interrupt Enable Cause an interrupt when RXC is set also.
1	R/W	TXCIE, TX Complete Interrupt Enable Cause an interrupt when TXC is set also.
2	R/W	UDRIE, UDR Empty Interrupt Enable Cause an interrupt when UDRE is set also.
3	R/W	RXEN, Receiver Enable '1' = Receiver Enabled '0' = Receiver Disabled
4	R/W	TXEN, Transmitter Enable '1' = Transmitter Enabled '0' = Transmitter Disabled

i. All registers are big-endian

BIT	ACESS	DESCRIPTION
5	R/W	CHR9, 9-bit Characters When set transmitted characters are 9 bits long plus start and stop bits. 9th bit is read and written by using RXB8 and TXB8 in UCR register, respectively.
6	R	RXB8, Received Data Bit 8 Contains the 9th data bit of the received character.
7	W	TXB8, Transmit Data Bit 8 Contains the 9th bit in the character to be transmitted.

Reset Value: 0x02

Table A1-2. UART Status Register

BIT	ACESS	DESCRIPTION
0	R	RXC, Receive Complete Bit Is set upon completion of reception of a byte. Cause an interrupt if RXCIE is set.
1	R	TXC, Transmit Complete Bit Is set upon completion of transmission of a byte. Cause an interrupt if TXCIE is set.
2	R	UDRE, Data Register Empty Bit Is set when data is shifted from data register to shift register for transmission. Cause an interrupt if UDRIE is set.
3	R	FE, Frame Error Is set when the stop bit is received as '0'.
4	R	OR, Overrun Error Is set when the byte in the data register is not read before a new byte has been shifted in.
5-7	R	Reserved Will read as '0's

Reset Value: 0x20

A2. I²C CONTROLLER REGISTERS

The I²C controller has three bit-assigned registers, the control, command and status registers.

Table A2-1. I²C Control Register

BIT	ACESS	DESCRIPTION
0	R/W	EN, Core Enable Bit When set to '1', core is enabled When set to '0', core is disabled
1	R/W	IEN, Interrupt Enable When set to '1', interrupt is enabled When set to '0', interrupt is disabled
2-7	R	Reserved Will read as '0's

Reset Value: 0x00

Table A2-2. I²C Command Register

BIT	ACESS	DESCRIPTION
0	R/W	STA, Generate (repeated) start condition when set to '1'.
1	R/W	STO, Generate stop condition when set to '1'.
2	R/W	RD, Read from Slave Device
3	R/W	WR, Write to Slave Device
4	R/W	ACK, Acknowledge to be sent ACK = '1' NACK = '0'
5-6	R	Reserved Will read as '0's
7	R/W	IACK, Interrupt Acknowledge Clears pending interrupt when set

Reset Value: 0x00

Table A2-3. I²C Status Register

BIT	ACESS	DESCRIPTION
0	R	RxACK, Acknowledge received from Slave Device '1' = No Acknowledge Received '0' = Acknowledge Received
1	R	Busy Bit '1' After Start Condition Detected '0' After Stop Condition Detected
2-5	R	Reserved Will read as '0's
6	R	TIP, Transfer-In-Progress '1' when Transferring Data '0' when Transfer Complete
7	R	IF, Interrupt Flag This bit is set when an interrupt is pending, i.e. when a single byte transfer has completed. Will cause an interrupt if IEN is set.

Reset Value: 0x00

A3. INTERRUPT CONTROLLER REGISTERS

Interrupts are enabled via the interrupt mask register while pending interrupts are shown in the interrupt pending register.

Table A3-1. Interrupt Mask Register

BIT	ACESS	DESCRIPTION
0	R/W	GIE, Global Interrupt Enable Bit Set to enable the interrupt signal to processor.
1	R	Reserved Will read as '0'
2-17	R/W	D/EIE Set to Enable interrupt sourcing from the respective signals of the Debug/Expansion port.
18	R/W	TempIE Enables the Temperature Sensor Interrupt when set.
19	R/W	LVDSIE Enables the LVDS Interrupt when set.

BIT	ACESS	DESCRIPTION
20	R/W	RTCIE Enables the RTC Interrupt when set.
21	R/W	SCCIE Enables the SCC Interrupt when set.
22	R/W	UART1TxIE Enables UART1 Transmit Complete Interrupt when set.
23	R/W	UART1RxIE Enables UART1 Receive Complete Interrupt when set.
24	R/W	UART1UDRIE Enables UART1 UDRE Interrupt when set.
25	R/W	UART2TxIE Enables UART2 Transmit Complete Interrupt when set.
26	R/W	UART2RxIE Enables UART2 Receive Complete Interrupt when set.
27	R/W	UART2UDRIE Enables UART2 UDRE Interrupt when set.
28	R/W	I^2CIE Enables the I^2C Controller Interrupt when set.
29	R/W	Sw0IE Switch 0 Interrupt enabled when set.
30	R/W	Sw1IE Switch 1 Interrupt enabled when set.
31	R/W	Sw2IE Switch 2 Interrupt enabled when set.

Reset Value: 0x0000_0000

Table A3-2. Interrupt Pending Register

BIT	ACESS	DESCRIPTION
0	R	Unused Will read as '0'
1	R	Reserved Will read as '0'
2-17	R	D/EIE Set if the corresponding Debug/Expansion interrupt is pending.

BIT	ACESS	DESCRIPTION
18	R	TempIE Set if the Temperature Sensor Interrupt is pending.
19	R	LVDSIE Set if the LVDS Interrupt is pending.
20	R	RTCIE Set if the RTC Interrupt is pending.
21	R	SCCIE Set if the SCC Interrupt is pending.
22	R	UART1TxIE Set if the UART1 Transmit Complete Interrupt is pending.
23	R	UART1RxIE Set if the UART1 Receive Complete Interrupt is pending.
24	R	UART1UDRIE Set if the UART1 UDRE Interrupt is pending.
25	R	UART2TxIE Enables UART2 Transmit Complete Interrupt when set.
26	R	UART2RxIE Set if the UART2 Receive Complete Interrupt is pending.
27	R	UART2UDRIE Set if the UART2 UDRE Interrupt is pending.
28	R	I^2CIE Set if the I^2C Controller Interrupt is pending.
29	R	Sw0IE Switch 0 Interrupt pending when set.
30	R	Sw1IE Switch 1 Interrupt pending when set.
31	R	Sw2IE Switch 2 Interrupt pending when set.

Reset Value: 0x0000_0000

A4. LVDS REGISTERS

The two bit-assigned LVDS registers provide control functionality and status information.

Table A4-1. LVDS Control Register

BIT	ACESS	DESCRIPTION
0	R/W	Tx Enable Bit When set to '1', transmission is enabled When set to '0', transmission is disabled
1	R/W	Rx Enable Bit When set to '1', reception is enabled When set to '0', reception is disabled
2-6	R	Reserved Will read as '0's
7	W	Synchronisation Bit Activate synchronisation by setting to '1'. When set to '0', synchronisation disabled

Reset Value: 0x00

Table A4-2. LVDS Status Register

BIT	ACESS	DESCRIPTION
0-6	R	Reserved Will read as '0's
7	R	Lock Bit When set to '1', PLL is locked When set to '0', Lock lost/not obtained

Reset Value: 0x00

A5. SYSTEM MANAGEMENT REGISTER

The system management register offers power-down functions to the various peripherals that support it. Its bit-assignment is shown in Table A5-1. All peripherals are active at power-up.

Table A5-1. System Management Register

BIT	ACESS	DESCRIPTION
0-8	R	Reserved Will read as '0's

BIT	ACESS	DESCRIPTION
9	R/W	UARTEN, UART Enable '1' = UART Receivers Enabled '0' = UART Receivers Disabled
10	R/W	UARTShdn, UART Shutdown '1' = UARTs Shut Down '0' = UARTs Active
11	R/W	LVDSDEN, LVDS Drivers Enable '1' = LVDS Bus Drivers Active '0' = LVDS Bus Tri-Stated
12	R/W	LVDSREN, LVDS Receiver Enable '1' = LVDS Receiver Active '0' = LVDS Receiver Disabled
13	R/W	LVDSPwrdn, LVDS Powerdown '1' = LVDS Devices Powered Down '0' = LVDS Devices Operational
14	R/W	TempStby, Temperature Sensor Standby '1' = Sensor in Standby Mode '0' = Sensor Active
15	R/W	CurShdn, Current Sensor Shutdown '1' = Sensor Shut Down '0' = Sensor Active

Reset Value: 0x0058

A6. DEBUG/EXPANSION REGISTERS

The four debug/expansion registers provide access to the debug/expansion port. They are the port, direction, led, and switch status registers and their bit-assignments are shown in Table A5-1.

Table A6-1. Port Register

BIT	ACESS	DESCRIPTION
0-15	R/W	The port pins reflect the value of these bits when the directions of the corresponding pins are set as outputs. When set as inputs, these bits are ignored.

Reset Value: 0x0000

Table A6-2. Direction Register

BIT	ACESS	DESCRIPTION
0-15	R/W	The direction of the corresponding pins of the port are set to inputs when '1' and to outputs when '0'. The pins are tri-stated when set as inputs and driven with the values of the corresponding bits in the Port Register when set as outputs.

Reset Value: 0xFFFF

Table A6-3. LED Register

BIT	ACESS	DESCRIPTION
0-11	R/W	Unused Will read as '0's
12	R/W	Status LED 1 '1' – LED 1 off '0' – LED 1 on
13	R/W	Status LED 2 '1' – LED 2 off '0' – LED 2 on
14	R/W	Status LED 3 '1' – LED 3 off '0' – LED 3 on
15	R/W	Status LED 3 '1' – LED 3 off '0' – LED 3 on

Reset Value: 0x000F

Table A6-4. Switch Status Register

BIT	ACESS	DESCRIPTION
0-12	R	Unused Will read as '0's
13	R	Switch 1 Status '1' – Switch 1 open '0' – Switch 1 pressed
14	R	Switch 2 Status '1' – Switch 2 open '0' – Switch 2 pressed

BIT	ACESS	DESCRIPTION
15	R	Switch 3 Status '1' – Switch 3 open '0' – Switch 3 pressed

Reset Value: 0x0007

APPENDIX B : SUPPLEMENTARY CD