An FPGA Based Fault Tolerant PPU for PNSS-1

Salim Ullah, Laiq Hasan, Asif Ali Khan, Ramsha Narmeen, Muhammad Jaleed Khan, Abdullah, Asim Shahzad

***Abstract*— The payloads being the principle performance drivers for a satellite, must work properly throughout the lifetime of a satellite to ensure space mission success. An onboard Payload Processing Unit (PPU) is required for payload data storage and its processing. Different design and implementation techniques are employed to achieve a reliable and computationally efficient PPU. Due to the high computational capabilities of modern FPGAs at relatively low power and cost, they can be used in the design and implementation of PPUs. Their support for soft-core processors and real time operating systems helps in increasing the efficiency of PPU in terms of performance and flexibility. This paper presents a fault tolerant PPU using commercial off-the-shelf (COTS) components for Pakistan National Student Satellite -1 (PNSS-1). The proposed PPU implements a LEON3 processor on a Xilinx Virtex 4 FPGA using Triple Modular Redundancy (TMR) and employing RTEMS for real time operations of the PPU software.**

Keywords*— *Microsatellites, PPU, FPGAs, COTS, PNSS-1, Reconfigurable Architecture, Payloads

I. Introduction

Research and development in space science and technology is a fundamental tool for the progress and development of a country in global environment. Unfortunately, due to lack of information sharing, students and early stage researchers of developing countries, like Pakistan, lack interest in space exploration. Pakistan National Student Satellite Program (PNSSP) is a right step taken in this direction by SUPARCO to overcome this barrier. The purpose of this program is to develop awareness and use the expertise of talented students and young researchers in space science development in Pakistan. The first satellite under this project, PNSS-1 is planned to be launched in the sun-synchronous Low Earth Orbit (LEO) at an altitude of 450-700km, in 2016.

The payload subsystem of PNSS-1 contains several payloads such as narrow view camera (NVC), wide view camera (WVC), experimental payload (EXP) and the payload transmitter (PLT). An on-board PPU is required to control and receive data streams produced by these payloads. The PPU stores the received information temporarily and performs telemetry and telecommand (TC) operations on it.

The reliability of a payload processing unit is critical for a satellite to complete its mission successfully. Besides providing performance efficiency and functional correctness, a PPU must withstand the harsh radiations and temperature variations in upper space. The noticeable effects faced by satellites in LEO are Single Event effects (SEE), Single Event Upset (SEE) and Total Ionization Dose (TID) effects. Radiation hardened ASICs have been used to mitigate these radiation effects and for the design and implementation of various components of satellites for decades. However, these radiation hardened ASICs are inefficient in terms of cost and computational power and they do not offer fault tolerance and reconfiguration of hardware to meet the changing mission requirements [1].

For the past few years, FPGA based systems have gained significant importance in designing hardware modules for space applications [2]. Modern space grade FPGAs support high data rate applications with flexibility, fault tolerance and dynamic partial reconfiguration at low power and cost. The dynamic partial reconfiguration enables FPGA based systems to be changed or adapted at run time according to unanticipated mission requirements [3-5].

A fault tolerant PPU for PNSS-1, using Xilinx Virtex 4 FPGA is presented in this paper. TMR has been proposed to minimize the harsh space environment generated ionization effects on memories.

Rest of the paper is organized as follows: section II describes the literature review. PNSS-1's requirements for PPU and common challenges faced by microsatellites in space are presented in section III. Section IV explains the trade-off analysis and proposed design in detail. The implementation and results are explained in Section V.

Manuscript received April 7, 2015. This work was supported in part by SUPARCO under Pakistan National Student Satellite Program.

Salim Ullah is with the Department of Computer Systems Engineering, University of Engineering & Technology Peshawar, Pakistan. (Phone: +92-347-7613040; e-mail: saleemullah@uetpeshawar.edu.pk).

Laiq Hasan is with the Department of Computer Systems Engineering, University of Engineering & Technology Peshawar, Pakistan. (e-mail: laiqhasan@uetpeshawar.edu.pk).

Asif Ali Khan is with the Department of Computer Systems Engineering, University of Engineering & Technology Peshawar, Pakistan. (e-mail: asif.ali@uetpeshawar.edu.pk).

Ramsha Narmeen is a student at the Department of Computer Systems Engineering, University of Engineering & Technology Peshawar, Pakistan. (e-mail: ramsha.narmeen@yahoo.com).

Muhammad Jaleed Khan is a student at the Department of Computer Systems Engineering, University of Engineering & Technology Peshawar, Pakistan. (e-mail: mjaleedkhan@outlook.com).

Abdullah Khan is a student at the Department of Computer Systems Engineering, University of Engineering & Technology Peshawar, Pakistan. (e-mail: abdullah.khan00185@yahoo.com).

Asim Shahzad is with the Satellite Research & Development Center, SUPARCO, Lahore Pakistan.

Finally section VI concludes the paper.

II. Literature Review

The design and development of fault tolerant and efficient payload processing units for satellites have remained an area of active research since the beginning of artificial satellites. Different kinds of development platforms, microprocessors and microcontrollers have been used for the development of PPUs for satellites. The prototype of a scalable, fault tolerant and dynamically reconfigurable payload processing unit using RAPTOR-X64 architecture is presented in [1]. The implemented processing unit supports multiple types of partial reconfiguration such as self-reconfiguration and external reconfiguration through external interfaces. The modular design approach of the presented PPU allows the processing unit to be adapted according to new architectures.

The authors of [6] have presented a Virtex 5 FPGA based payload processing unit for the real-time data processing operations of a Multiangle Spectro-polarimetric Imager (MSPI). The developed system is responsible for real time processing of data from nine cameras and other payloads. An Atmel microcontroller based processor performs the telemetry and telecommand operations of the satellite.

A microsatellite processing system has been presented in [7]. The presented system uses an SRAM based FPGA which performs all the mathematical operations, error detection and correction checks and housekeeping operations using three Nios-2 cores and a mathematical coprocessor. TMR and reconfiguration is used for mitigation and correction of errors caused by radiation effects. A Watch Dog timer allows a microcontroller to monitor the FPGA periodically using heart beat signals and reconfigure it in case of failures.

The authors of [8] have presented a low cost and fault tolerant on board computer (OBC) for microsatellites which uses an ARM7 based processor and two anti-fuse FPGAs to protect memories of OBC from upper space radiations. The integrity of OBC's Linux based software is ensured by using modular and hierarchical approach.

A partially reconfigurable single chip processing unit using Xilinx Virtex 4 FPGA for satellites has been presented in [9]. A SPARC V8 based softcore processor is responsible for initial configuration of the FPGA, scrubbing and decompression of the uploaded compressed partial reconfiguration bit stream file.

Due to increasing complexity of satellites' payloads, the onboard data storage and its processing requirements are becoming major space mission challenges. The authors of [10] have presented an on-board payload data processing system optimized for storage and processing of large amount of data. Various techniques, such as data filtering and data compression, have been proposed to fulfil the increasing data storage requirement.

PROBA-1, the first small satellite technology development and demonstration precursor mission within the European Space Agency's (ESA) General Support Technology Program (GSTP) series. The satellite has a payload processing unit for the processing of image data [11]. The PPU of PROBA-1 consists of a solid-state recorder and a data signal processor (DSP) for the processing of payload data. A memory management unit (MMU) has been employed by the PPU for the storage of resulting images. PPU also provides several additional interfaces such as interfaces to solid state gyroscopes (SSG), an interface to an extra star tracker (PASS), and an interface to a House-keeping bus. In PROBA-2, the second small satellite, the conventional subsystems, such as power conditioning system (PCS), power distribution unit (PDU), data handling system (DHS), mass memory unit (MMU) and PPU are merged into one system called the advanced data and power management system (ADPMS) [12]. It results in the resource optimization with a volume reduction of more than 30%. ADMPS has been implemented in two cold redundant nominal TC-decoders. Its processor board has been designed for 100 MHZ operations with tunable clock frequency, 64 MB SDRAM, 4 MB SRAM 4 MB Flash and 256 KB PROM.

Payload Data Handling Unit (PDHU) for Gaia space observatory, presented in [13], is a controller board with a storage capacity of 960Gb that uses 240 SDRAMs each with a capacity of 4 Gb .Among other functions, PDHU act as a 'hard-disk' of Gaia. It sorts and stores thousands of compressed images per second from payload cameras. The low priority data is deleted if data rates or data volumes exceeds the storage capacity of the system.

The Payload Controller, Processor and Memory Unit (PCPMU) presented in [14] challenges the conventional approach to Payload Electronics (PE) by creating a modular architecture, utilizing highly multi-purpose, integrated components (based on reconfigurable and state-of-the-art FPGA technology) and different standardized interfaces. The PCPMU is a modular, reconfigurable PE subsystem and it consists of digital and analog cards/modules. PCPMU consists of a controller/processor section and a data storage unit. The processor section is mission specific and it compresses image data for optical missions and processes on-board data (modulation and demodulation). The data storage unit is common for all configurations and it is a highly scalable unit that provides control and data management function.

The adaptive instrument module (AIM) for FedSat [15] contains an SRAM-based FPGA providing the capability to evolve and adapt to changing mission requirements. It uses 16-bit microcontroller based on Intel 80C196 architecture for processing of data.

III. Challenges & Requirements

The presence of radiations and thermal variations in upper space can cause undesirable changes to electronic devices. These effects range from degradation of performance to functional failures. As a result, satellites may experience partial or complete degradation of the equipment, shortened lifetimes and major mission failures.

The major challenges faced by satellites in low earth orbits are single event upset (SEU), single event latchup (SEL) and total ionizing dose (TID) effects. Charges deposition due to radiations induced ionization can cause SEU, i.e. a "bit flip" in a memory element or a logic circuit that can result in short-term unanticipated behavior of the circuit. The ionization tracks can cause parasitic components, allowing unwanted currents flow between components in a circuit, which can permanently damage an electronic circuit. Moreover, the long-term accumulation of radiations induced ions, termed as TID, can degrade the overall electronic component.

Besides withstanding the aforementioned radiation hazards, the proposed PPU is required to fulfill a number of requirements as posed by PNSS-1. The proposed PPU, presented in Figure 1, is required to perform telemetry and telecommand operations of all payloads (NVC, WVC, EXP, and PLT) and store output data of payloads and Data Handling Unit (DHU) till ground contact. PPU is responsible for acquiring, packetizing, encoding and storing payload data from all payloads and housekeeping telemetry. DHU, which is connected to PPU through CAN bus and a dedicated RS-422 backup interface, is responsible for providing packetizing services of telemetry and telecommands according to the Consultative Committee for Space Data Systems (CCSDS) standard. The stored telemetry and payload data packets are required to be downlinked through PLT on ground contact.

PPU after receiving telecommands from DHU, is required to execute its own telecommands and distribute the telecommands for all payloads. In case of failure of master CAN controller, PPU will act as a slave CAN controller. PPU, being the critical unit in payload subsystem, must be physical redundant and fault tolerant. The interface and functional requirements of PPU are demonstrated in Figure 1.

PPU is also required to be physically redundant with only one copy of PPU operating at a time. Moreover, the maximum initial usage limit on processor and memory is 50%. The data rate of PPU can be selected by TC in a range of 64-512Kbps. For data acquisition, the memory is divided into three parts, i.e. 10% for telemetry, 10% for EXP Data and 80% for payload data.

IV. Trade-off Analysis & Solutions To Challenges

The selection of development platform, processing core and RTOS for PPU were the major design choices to meet the requirements of PNSS-1 and withstand the harsh space environment. These choices have been discussed below and demonstrated in Figure 2.

A. Development Platform

Due to high computational power, hardware re-programmability, reconfiguration, parallelism, and software up-gradation capabilities, FPGAs have been used in a variety of space missions.

Space grade FPGAs such as Xilinx Virtex-5 FPGA are radiation hardened and thus they are less vulnerable to the radiations and thermal effects caused by the harsh space environment. They provide immunity against SEU, SEL and TID effects, and have large system integration capacity. A number of satellites have used space grade FPGAs for implementation of different subsystems of satellites.

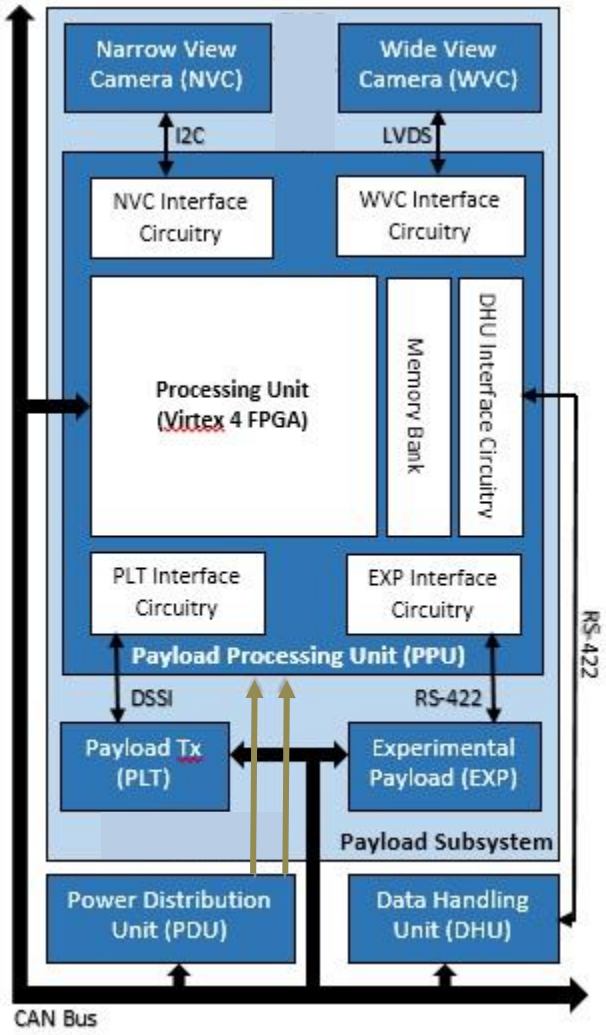

Figure 1. Conceptual Level Block Diagram of PPU

Non space grade FPGAs such as Virtex-4 are preferred for cost-constrained space missions. The radiation-hardening in these FPGAs is achieved through coating and shielding techniques. Xilinx Virtex-4 FX family provides high-performance and full-featured solutions for Embedded platform applications.

Space grade FPGAs, due to their high cost are not suitable for microsatellites such as PNSS-1. The proposed design presented in this paper uses Virtex-4 FX FPGA with TMR, coating and shielding for radiation-hardening.

B. Processing Unit

The processing unit of a PPU comprises of a microprocessor, non-volatile and volatile memories and the

interfaces that connect the microprocessor to the on-board equipment.

ARM7 hard-core processor based on ARMv4T architecture is feasible for nanosatellites due to their low cost, low power consumption and high computational power.

Soft-core processors such as LEON processors can be implemented on programmable devices such as FPGAs and ASICs. Contrary to hard-core processors, soft-core processors can be optimized for specific requirements. In many space applications, high performance is compromised to achieve flexibility. The proposed PPU uses the flexibility of LEON 3 for the design and implementation of its processing unit.

The LEON3 is a synthesizable VHDL model of a 32-bit processor compliant with the IEEE-1754 (SPARC V8) architecture. The LEON3 features a 7-stage pipeline and supports both asymmetric and symmetric multiprocessing (AMP/SMP). Up to 16 processors can be used in a multiprocessing configuration. The LEON3 model is highly configurable, and suitable for system-on-chip (SoC) designs. Some features of the vast variety of functionalities provided by LEON3, which helped in its selection for PPU, are listed below;

- SPARC V8 instruction set with V8e extensions
- Advanced 7-stage pipeline
- Hardware multiply, divide and MAC units
- High-performance, fully pipelined IEEE-754 FPU
- Separate instruction and data cache (Harvard architecture) with snooping
- Configurable caches 1 - 4 ways, 1 - 256 Kbytes/way. Random, LRR or LRU replacement
- Local instruction and data scratch pad RAM, 1 - 512 Kbytes
- SPARC Reference MMU (SRMMU) with configurable TLB
- AMBA-2.0 AHB bus interface
- Advanced on-chip debug support with instruction and data trace buffer
- Symmetric Multi-processor support (SMP)
- Power-down mode and clock gating
- Robust and fully synchronous single-edge clock design
- Up to 125 MHz in FPGA and 400 MHz on 0.13 um ASIC technologies
- Extensively configurable
- Large range of software tools compilers, kernels, simulators and debug monitors
- High Performance 1.4 DMIPS/MHz, 1.8 Core Mark/MHz

C. Real Time Operating Systems (RTOS)

A real time operating system ensures that interrupts and other time critical tasks are processed when required. Implementation of an RTOS can provide priority levels to various critical tasks. For PNSS-1 PPU, multiple options were considered, such as VxWorks and Real Time Executive for Multiprocessor System (RTEMS). VxWorks is a high performance, scalable RTOS, developed by Wind River Systems, that supports a variety of processors. After careful consideration, the open source RTOS RTEMS was selected for the proposed PPU. RTEMS fulfils the specific requirements of PPU by having real-time capabilities, full availability of the source code, functional development environment, cross-platform support and community support. It is a fully featured RTOS that supports variety of open standard application programming interfaces (APIs) and interface standards like POSIX. The key features of RTEMS are listed below;

- Multitasking capabilities
- Homogeneous and heterogeneous multi-processor systems
- Event-driven, priority based, preemptive scheduling
- Optional rating monotonic scheduling
- High level of user configurability
- Dynamic memory allocation
- Inter-task communication and synchronization
- Responsive memory management

It is designed for embedded systems such as robotic controllers and on-board satellite instruments. RTEMS supports multi-processor systems with a dozen of CPU architectures and over 150 specific system boards. It has been ported to various processors architectures i.e. ARM, Atmel AVR, Black fin, free scale Cold-Fire, Intel, MIPS, LEON, SPARC V9, Pentium, Lattice Mico32 etc. RTEMS provide directives which can be used to dynamically create delete and manipulate a predefined object types. The object oriented nature of RTEMS allows the creation of modular applications using reusable “building block” routine.

RTEMS is a commonly used space application software with its development resources easily accessible via Git repository.

The proposed PPU, as shown in Figure 1 and Figure 2, uses Xilinx Virtex 4 FPGA to implement the PPU’s processing unit using the LEON3 soft-core processor along with RTEMS to run the application software of PPU.

V. IMPLEMENTATION & RESULTS

Using Windows platform, the Gaisler Research IP-cores Library (GRLIB) configuration has been performed to implement the proposed design. The GRLIB provides re-usable IP cores for commercial and space applications development. Design templates for SPARC architecture based processors along with supporting development tools such as GRTools are also available in GRLIB [16]. Since the GRLIB tools are based on Linux make-file, therefore cygwin has been used to imitate the functionality of make and other important utilities such as tcl, tk. The SoC based PPU has been implemented using an integrated set of GRLIB’s IP-cores and development tools; the simulation results are obtained using ModelSim and the synthesis results are obtained using Xilinx Integrated Synthesis Environment (ISE).

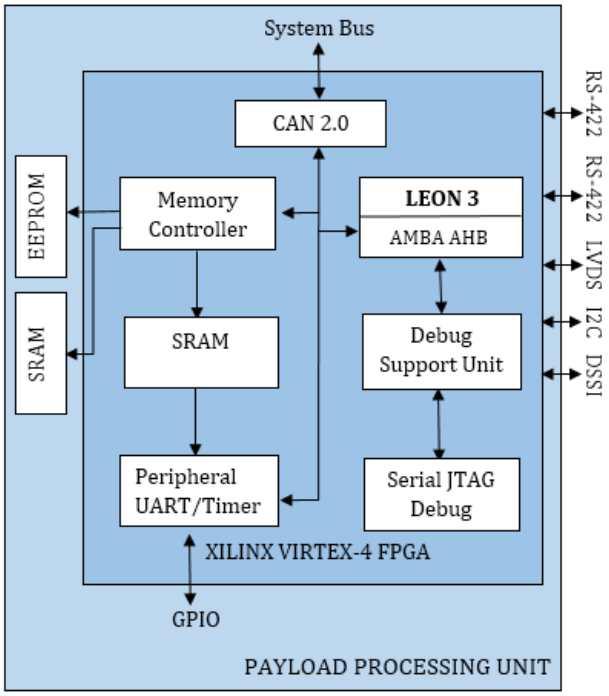

Figure 2. Proposed Design of PPU's Processing Unit

The distinctive plug & play method of GRLIB has been used to configure and connect the IP cores, without any modification in its global resources. In this approach the IP cores are fixed on a common on-chip bus. The AMBA-2.0 AHB/APB bus has been used to connect the IP-cores and obtain a bus-centric design. The LEON3 template design for xilinx-xc4vlx100-10ff1513 board has been used for implementation. The coherent method has been used for the simulation and synthesis of the design.

A. Configuration of design

GRLIB's command line GUI tool, i.e. xconfig allows adding, removing, and configuration of IP cores (processor cores, memory controllers, etc.) in the design. The designed model was configured using xconfig according to the proposed design of PPU. AMBA bus IP core has been considered as the system bus, connecting all other IP cores, such as. LEON3 processor core, memory controller and peripherals. For monitoring the registers' status and dumbing logs in the simulator, debug support IP core having two interfaces i.e. a JTAG and a Serial RS422 is also included in the implementation. Moreover, GPIO ports are used for communication between PPU and other units of its subsystem. The DHU is connected to PPU through the CAN bus interface provided by Gaisler. An SRAM has been inferred on to the FPGA which performs the program space for LEON3 along with loading the RTOS and other PPU software from the EEPROM on PPU startup.

B. Design Model Simulations and Design Synthesis

ModelSim, a simulator for simulating Verilog and VHDL designs, has been used for simulation of the configured design. The design includes a test-bench for simulating a PROM which is used for execution of a test program. It consists of two parts, i.e., a simple PROM boot loader (prom.S) and the test program itself (systest.c). The simple PROM boot loader (prom.S) initializes the processor, memory controller and other peripherals. After successful simulations, the complete design has been synthesized using Xilinx ISE. After mapping the whole design, ISE is launched from Cygwin using make commands. Table 1 and Figure 3 summarizes the synthesis results.

Logic Utilization	Used	Available	Utilization
No. of Slices	22998	49152	46%
No. of Slice Flip Flops	13737	98304	13%
No. of 4 input LUTs	42851	98304	43%
No. of bonded IOBs	265	960	27%
No. of FIFO16/RAMB16s	96	240	40%
No. of GCLKs	5	32	15%
No. of DCM_ADVs	2	12	16%
No. of DSP48s	3	96	3%

Table 1. Device Utilization Summary (estimated values)

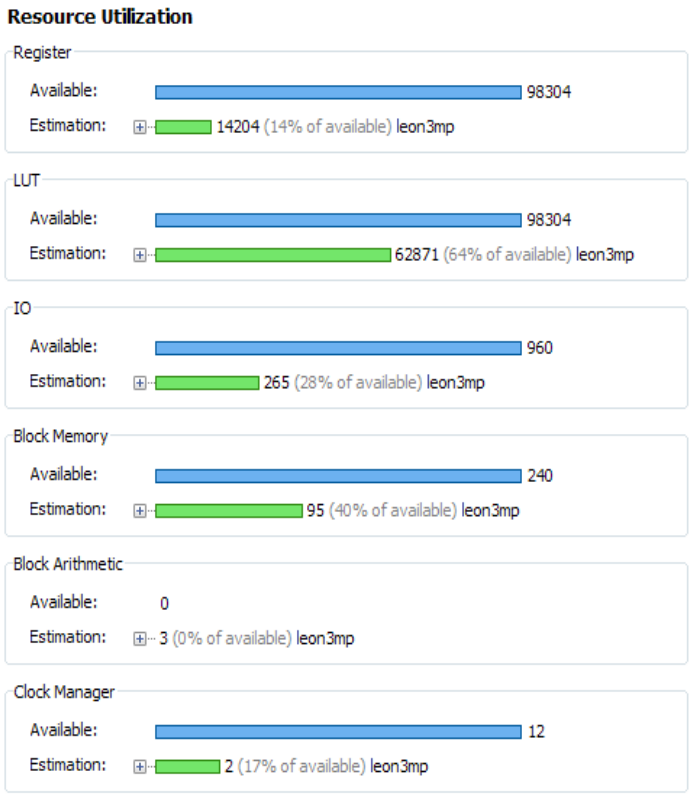

Figure 3. PPU's Processing Unit Resource Utilization

As shown by Table 1 and Figure 3 that the proposed design of PPU's processing unit utilizes only one third of the slices available on Virtex 4 FPGA, therefore TMR

scheme can be easily implemented for on board SRAM. Satellite systems often use TMR scheme instead of hamming code for the protection of vulnerable memories against SEUs, SELs and TIDs. The presented design leaves enough FPGA slices for TMR implementation of SRAM.

CONCLUSION

This paper presents the design of a fault tolerant payload processing unit for PNSS-1. The proposed design uses COTS components such as Virtex 4 FPGA and LEON3 processor. RTEMS is used for real time operation of PPU Software. TMR is also discussed for providing increased fault tolerance for memories. The proposed design meets all the requirements and constraints of PNSS-1 satisfactorily.

ACKNOWLEDGMENT

We are thankful to SUPARCO for their assistance. We are also grateful to Dr. Hamad Alizai for their constant support and encouragement throughout this project.

REFERENCES

[1] J. Hagemeyer, A. Hilgenstein, D. Jungewelter, D., andJ. Ilstad, "A scalable platform for run-time reconfigurable satellite payload processing," in *Proc. Adaptive Hardware and Systems (AHS), 2012 NASA/ESA Conference on* , vol., no., pp.9,16, 25-28 June 2012 doi: 10.1109/AHS.2012.6268642

[2] Ferguson, R.C.; Tate, R., "Use of field programmable gate array technology in future space avionics," Digital Avionics Systems Conference, 2005. DASC 2005. The 24th , vol.2, no., pp.11 pp. Vol. 2,, 30 Oct.-3 Nov. 2005.

[3] Steiner, N.; Athanas, P., "Hardware autonomy and space systems," Aerospace conference, 2009 IEEE , vol., no., pp.1,13, 7-14 March 2009.

[4] Koester, M.; Luk, W.; Hagemeyer, J.; Porrmann, M.; Ruckert, U., "Design Optimizations for Tiled Partially Reconfigurable Systems," Very Large Scale Integration (VLSI) Systems, IEEE Transactions on , vol.19, no.6, pp.1048,1061, June 2011.

[5] Guodong Xu; Song Wang; Jie Lin; Yuan Liu, "A novel reconfigurable on-board computer based on soft core CPU," Mechatronics and Automation, 2009. ICMA 2009. International Conference on, vol., no., pp.2467,2471, 9-12 Aug. 2009

[6] Bekker, D.L.; Werne, T.A.; Wilson, T.O.; Pingree, P.J.;Dontchev, K.; Heywood, M.; Ramos, R.; Freyberg, B.; Saca, F.; Gilchrist, B.; Gallimore, A.; Cutler, J., "A CubeSat design to validate the Virtex-5 FPGA for spaceborne image processing," Aerospace Conference, 2010 IEEE , vol., no., pp.1,9, 6-13 March 2010

[7] Guodong Xu; Song Wang; Jie Lin; Yuan Liu, "A novel reconfigurable on-board computer based on soft core CPU," Mechatronics and Automation, 2009. ICMA 2009. International Conference on, vol., no., pp.2467,2471, 9-12 Aug. 2009

[8] Shiqiang Tian; Zuobiao Yin; Jian Yan; Xuming Liu, "Design and implementation of a low-cost fault-tolerant on-board computer for micro-satellite," Communications and Networking in China (CHINACOM), 2012 7th International ICST Conference on , vol., no., pp.129,134, 8-10 Aug. 2012

[9] Daixun Zheng, Dr. Tanya Vladimirova, Hans Tiggeler, Prof. Martin Sweeting, "Reconfigurable single-ship on board computer for a small satellite," IAF-01-U3.09.

[10] Armbruster Philippe, ESA/ESTEC/TOS-ETD.;Gasti Wahida, TERMA & ESA/ESTEC/TOS-ETD.,"On-board payload data processing systems, on-board networks for future missions" EUSIPCO'2002, Volume II pp 577-580.

[11] Satellite Mission PROBA-1 (Project for On-Board Autonomy - 1), Launched by ISRO in 2001, "https://directory.eoportal.org/web/eoportal/satellite-missions/p/proba-1".

[12] Satellite Missions PROBA-2 (Project for On-Board Autonomy-2), Launched by ISRO on November 9, 2009, "https://directory.eoportal.org/web/eoportal/satellite-missions/p/proba-2".

[13] ESA, Gaia was launched on 19 December 2013 by Arianespace using a Soyuz ST-B/Fregat-MT rocket flying from Kourou in French Guiana.

[14] Sasa T.; Trajkovic.; George Tyc.; Kenneth V. James,; Daniel J. Schulten; Peter Allan; Ed Ahad; Richard Allen; Simon Ladouceur; Martin Cote, "PCPMU: A Modular, Multi-Use Payload Electronics Architecture for Affordable, Responsive Missions" 7th Responsive Space Conference, April 27-30, 2009, Los Angeles, CA

[15] FedSat (Federation Satellite) Australian satellite, launched from Tanegashima Space Center, Japan by a NASDA H-IIA launch vehicle in December 2002 (NASDA is now merged with JAXA)

[16] Aeroflex Gaisler "www.gaisler.com"