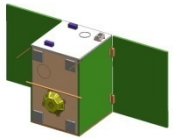
Reliability Prediction of a Fault Tolerant COTS Parallel Computing Payload

Sharon Lim Siok Lin[1] **Mok Yin Liong[2]**

1) Satellite Engineering Centre, Nanyang Technological University, Singapore

2) DSO National Laboratories, Singapore



Fault Tolerant, High Performance Parallel Processing Unit (PPU) onboard XSat

CREST

A 120 kg Polar and Sun-synchronous LEO micro-satellite developed by CREST

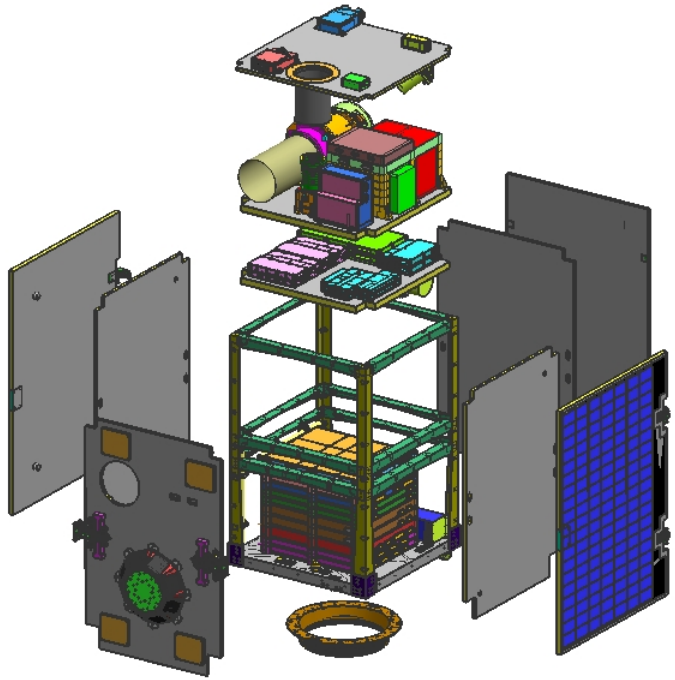
Collaboration between Nanyang Technological University of Singapore and DSO National Laboratories, Singapore

Remote-sensing mission
Launch date: 2009
Piggy-back launch with PSLV at Sriharikota (SHAR)

Primary Mission Payload:
IRIS Multispectral Camera (10m resolution)

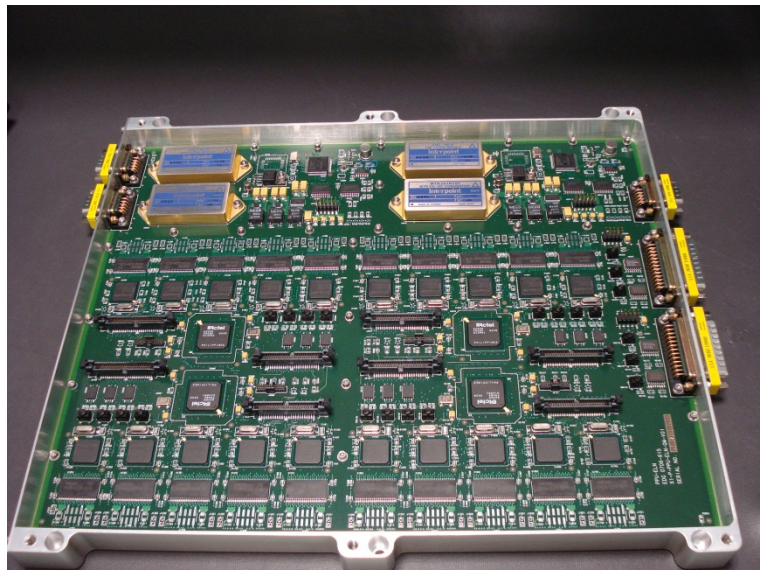
Secondary Mission Payloads:
Parallel Processing Unit (PPU)
DLR GPS Module

XSat



PPU Overview

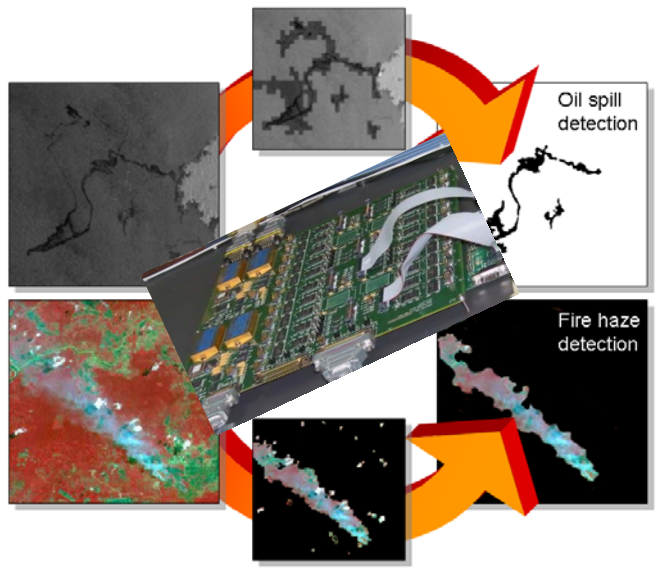
Secondary Mission Payload:
Parallel Processing Unit (PPU)

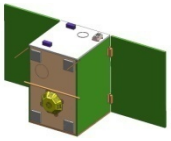


Objective:
To demonstrate a low-cost, fault tolerant and high performance payload for a micro-satellite such as XSat.

Research Rationale:
To research on enabling technology for low cost small satellites to have a high performance and reliable parallel processing capability onboard.

To enable a new class of applications formally not performed on microsatellites.

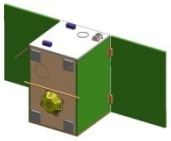




PPU Overview

Design Philosophy:

- Based on the usage of COTS components to take advantage of their
  - Lower cost
  - Ease in parts procurement and ITAR restrictions
  - Higher computation power (e.g. COTS processors are in general a few magnitudes higher in computation power than their space-grade counterparts).
  - Reduced power consumption, mass and volume than their space-grade counterparts.
- Uses parallelism to increase computation power as well as for fault tolerance through hardware redundancy (mixed redundancy).
- Ensures reliability through the incorporation of hardware redundancy and software/hardware fault-masking techniques

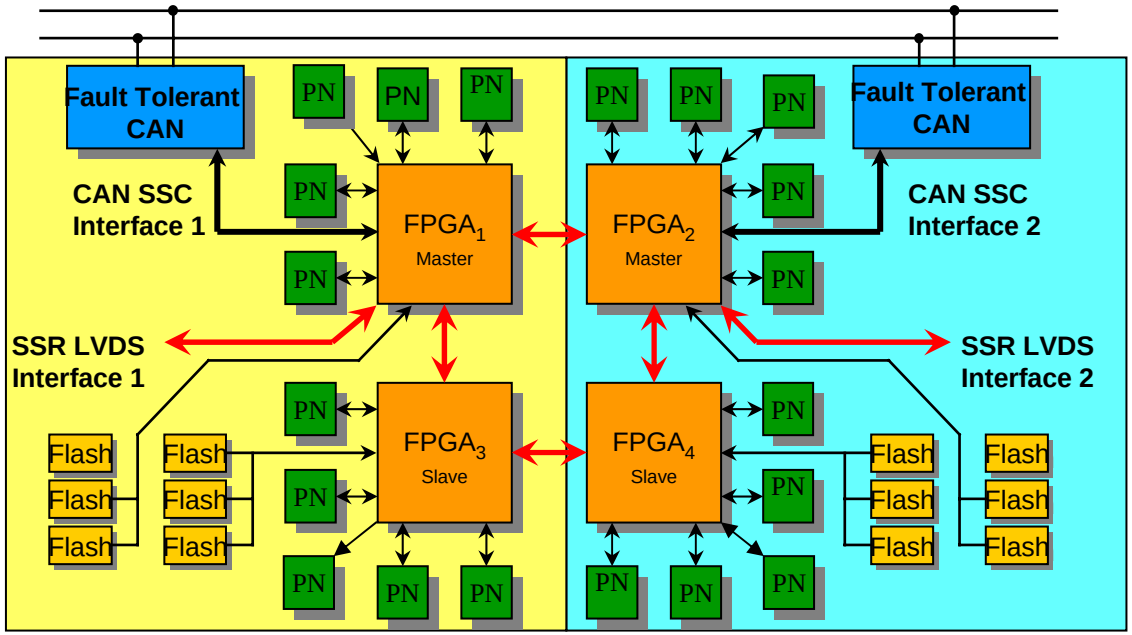


Project Goals

- A computing payload designed strictly within constraints of mass, volume, power and costs of XSat Project, which are typical of a micro-satellite remote sensing mission.
- Performance/power ratio of at about 200 MIPS/watt, scalable up to 40 watts.
- A demonstrator of a fault tolerance methodology that can achieve a hardware reliability of at least 0.9 for 1GIPS of computation power.
- Provision of a familiar application environment for users (e.g. Beowulf Linux Cluster in space)
- A software reconfigurable payload that function as a test bed for new applications and various software fault tolerant techniques.
- A platform to enable new class of applications for low cost small satellites (e.g. continuous surveillance or disaster monitoring).

PPU System Architecture

1. Hardware Architecture



Linux Beowulf Distributed Parallel cluster of COTS processors

4 FPGA Network Elements (AX1000-fg484)
- Four Actel antifuse FPGAs (AX1000-fg484)
- Each FPGA has a 25 wire interface with two neighbouring FPGAs in a square matrix configuration

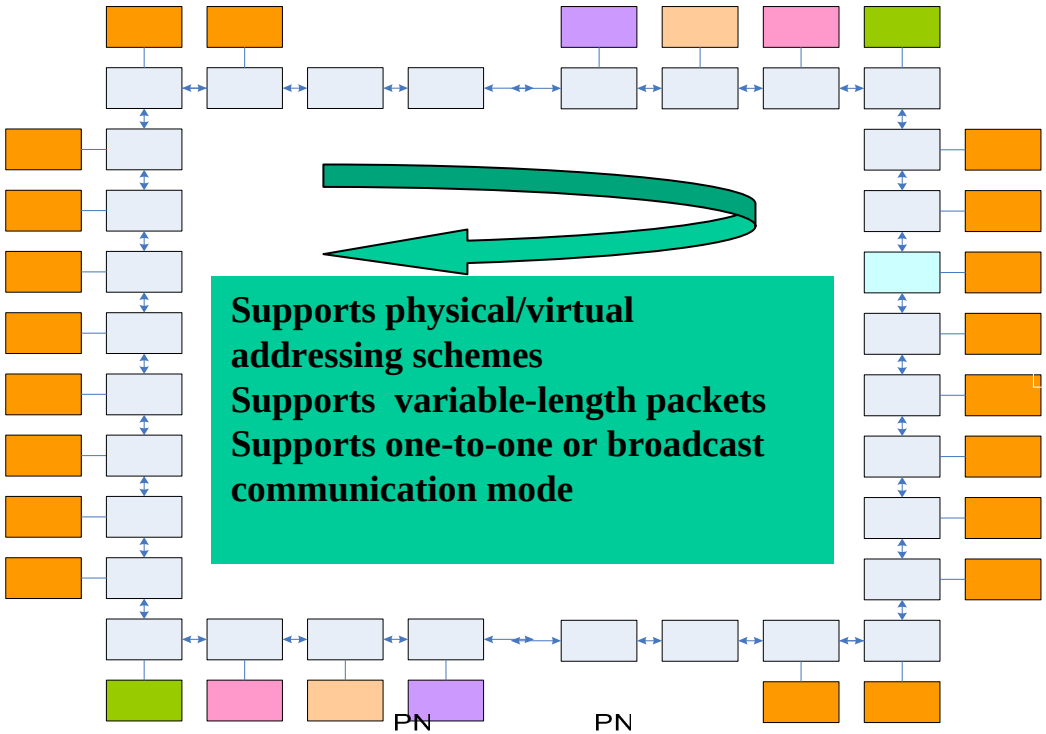
Connected to each FPGA
- Five Intel StrongArm SA1110 .
- Three Serial Flash Chips (Atmel AT45DB642D-CNU)

Connected to each Processor Node (SA1110)
- Two 32 MB SDRAM (Samsung K4S561632H)
- Two power-on transistor switches for its 1.8V core voltage and 3.3V IO voltage.

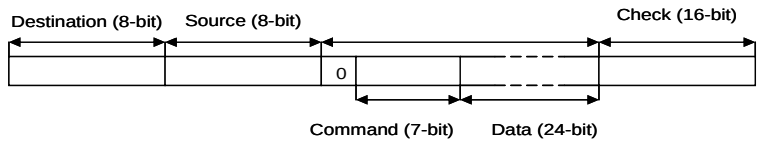
PPU System Architecture

2. Communication Network Architecture

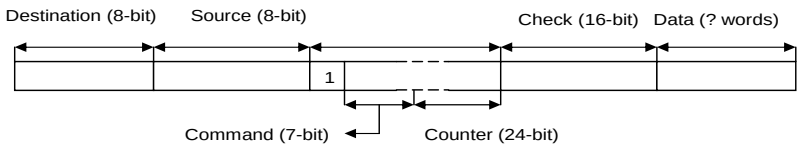
The FPGA provides a simple one-way ring-type bus henceforth referred to as the Variable Time-slotted Global Backplane (VTGB). This Packet Network spans across four FPGA clusters and has a one-way ring network topology.



VTGB Short Message Format



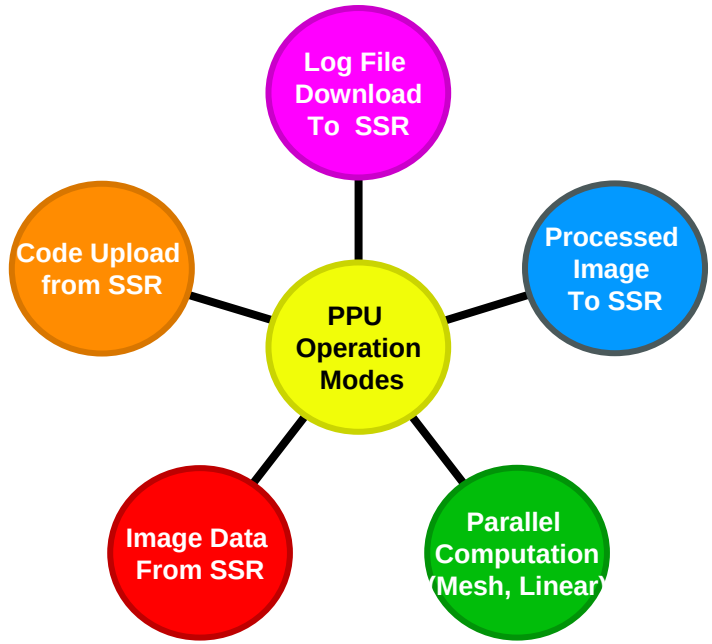
VTGB Long Message Format



PPU System Architecture

3. Software Platform

- COTS Open-source software suite (blob boot-loader, open-source Linux Kernel Version 2.6.4)
- Current design does not limit migration to use of other software suite, like VxWorks operating system …etc.
- A set of triple-redundant serial flash in each cluster stores all the SA1110 bootload code, Kernel and ramdisk images that are used to program the processor.
- The codes in the serial flash can be dynamically uploaded in orbit.



Component Selection

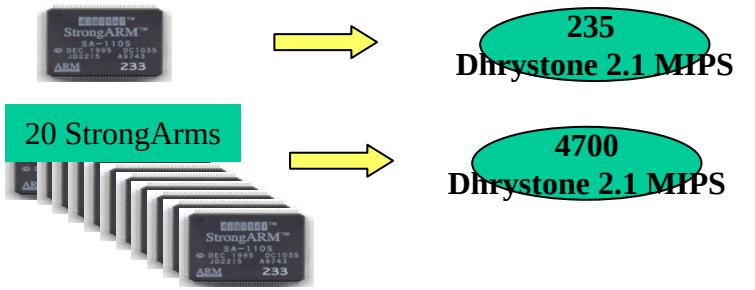
Parts Selection

- Preference of Components with Space Heritage.
- Preference for components with Radiation Data
- Select components which provide savings in power, space, cost and mass.
- Wide usage of COTS BGA components
- Usage of processors with high MIPS/watt and MIPS/cost

Role	Part	Heritage	Availability of Radiation Data
PN	StrongArm SA1110	Yes	No
NE	Actel AX1000S	No	Yes
Volatile Memory	Samsung K4S561632H SDRAM	No	Yes
Non Volatile Memory	Atmel Serial Flash	No	No
DC-DC converter	Interpoint MTR283R3SF	Yes	No

Component Selection

- PN selection: Why SA1110



High Performance Per Watt and High Performance/Cost

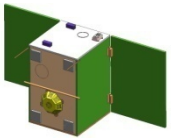
Predecessor SA1100 used onboard SNAP1 nanosatellite

- Network Element selection: Why Actel AX1000

Radiation Specs	RTAX1000S [4]	AX1000S	X-Sat Requirement
SEU LET_{TH} ($MeVcm^2/mg$) [5]	63	1.4 (for SRAM) 2.89 (for registers) 6.73 (for clock & control logic	37 (satellite industry benchmark)
SEU Rate (errors/flip flop/ day)	6.7×10^{-9}	8.9×10^{-5}	
SEL LET_{TH} ($MeVcm^2/mg$) [5]	117	120	40
Single-Event Transient (SET)	No anomalies up to 150MHz	NA	NA
Total Ionizing Dose	300 krad (Si)	200krad (Si)	10 krad (Si)

- More resilient towards single-event-upsets (SEUs) as its internal architecture will not be affected by SEUs (cells are programmed by blowing physical fuses).

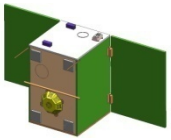
- However there will be SEUs in the registers and especially in the internal SRAM (which has very low SEU threshold)



Samsung K4S561632H SDRAM

CREST

- Based on “IEEE 2006 Radiation Workshop” published results on radiation testing for its predecessor, K4S561632E-TL75 chip.
- Samsung K4S561632H SDRAM 256Mb is estimated to have a MTBF of 27 days and to be latch-up immune.
- Based on (255,223)RS EDAC and assuming a probability of an uncorrectable upset per day < 10^{-6}, the memory scrubbing interval is computed to be greater than the operational period of the payload.
- Hence scrubbing is not required for PPU operation for this level of reliability.



PPU Fault Tolerant Methodology

CREST

PPU fault tolerance methodology lies in adopting five main strategies

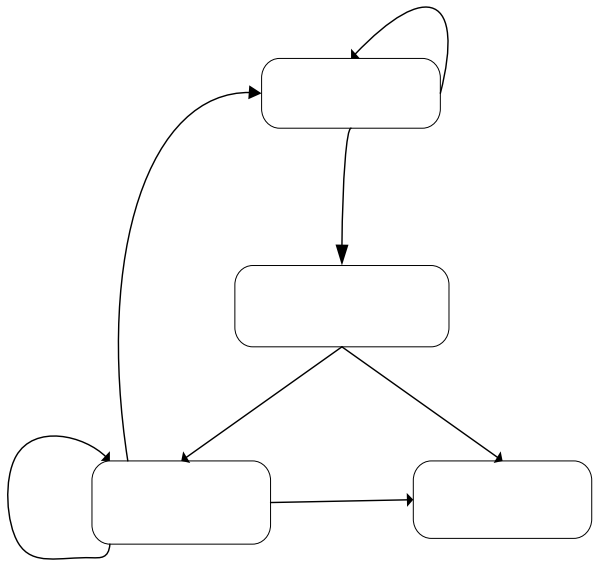
1. Having no single point of failure through hardware redundancy for all modules (either 1 for 1 redundancy or N for K redundancy where N > K).

   20 Redundant processing nodes
   4 Redundant network elements
   Dual Redundant Command Bus Link
   Dual redundant LVDS Data Bus Link
   Dual redundant power supply module
   4 sets of triple-voting serial flash chips
   20 distributed memory banks

2. Software Adaptability for the deployment of spare processors for replication of code execution or adopting a N-voting computing paradigm.
   - PPU is totally software reconfigurable and hence adaptive to different mission needs on reliability goals
   - Spare processors can be used to replace faulty processors during faults.
   - Spare processors can also be deployed to run the same program with the primary processor in a triple-voting for higher order voting fashion for more important mission tasks

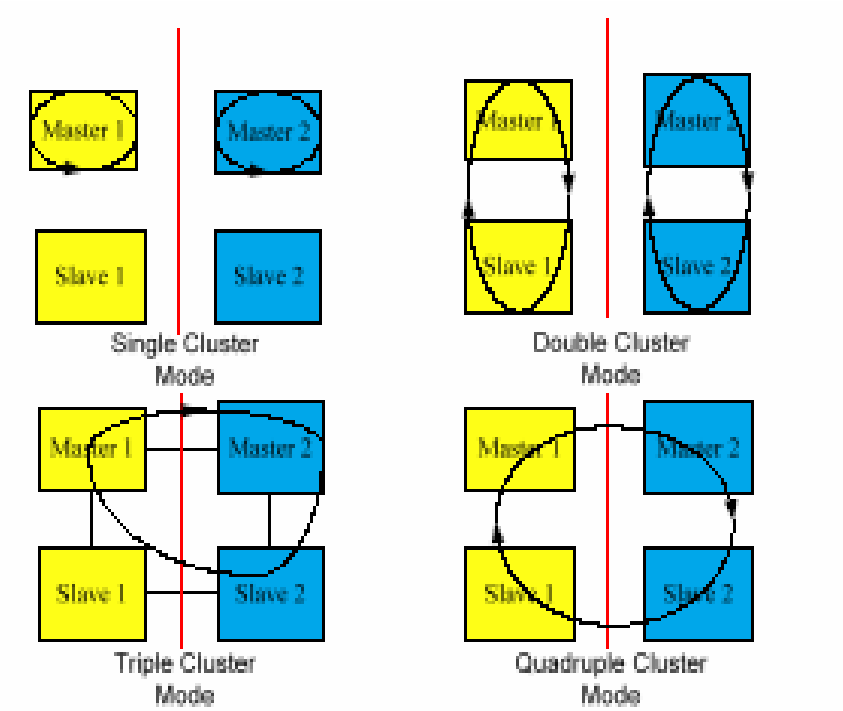
PPU Fault Tolerant Methodology

3. Autonomous Fault detection and Processor Recovery Routine
   - The FPGA network element contains the logic to monitor processor health and to control their on/off or reset states
   - Each network element maintains a watch-dog timer for all the processors in its cluster.
   - Processors have to write to FPGA health register at regular intervals to indicate that it is "alive"
   - Each FPGA network element has the ability to reset or power-cycle processors that are deemed faulty.



PPU Fault Tolerant Methodology

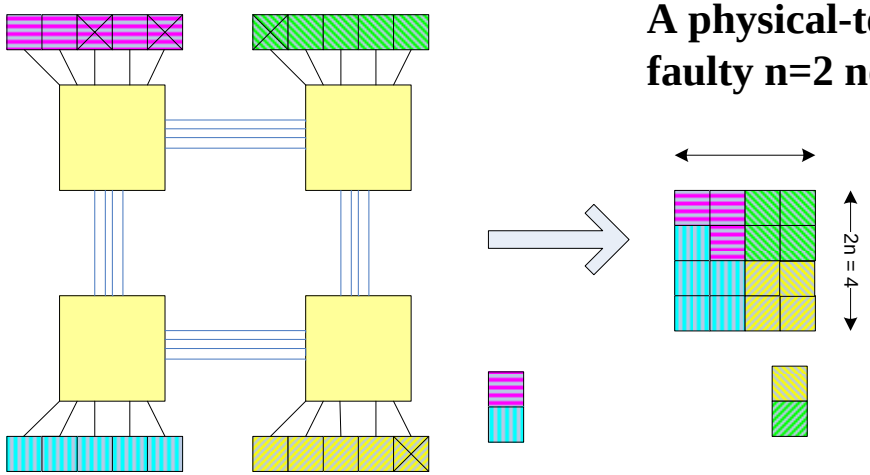
4. Fault Masking Techniques
   - Software EDAC schemes (Hamming/Reed-Solomon etc)
   - Software multiple reboot tries
   - Triple-voting read of the serial flash
   - Dynamic VTGB network reconfiguration capability



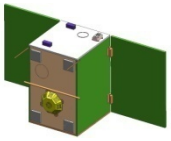
Various FPGA VTGB network configuration modes

PPU Fault Tolerant Methodology

5. Virtual Resource Management
   - Mapping of virtual requested resource to healthy resources dynamically in runtime
   - External interfaces communicates to entities using virtual addresses
   - Parallel applications uses different virtual addressing schemes to adapt to different parallel processing structures, e.g. logical positions in a mesh logical processing arrays. (see figure below).
   - A specific virtual Address can identify a node with a specific functional role in a Master/Slave programming structure such as the concept of a “System Node”.



A physical-to-logical processor mapping in a faulty n=2 network



Reliability Prediction

CREST

Reliability prediction of PPU over a period of 3 years mission operation under Xsat space environment will be assessed in quantitative terms:

Part 1:

Derive the failure rates of the various components in the payload using on the methods below (Lamda Predict Software from Reliasoft is used for the failure rates computation)

- Manufacturers' High Temperature Operating Life (HTOL) test data
- Estimated using Empirical reliability models from *MIL-HDBK-217F Notice 2, Reliability Prediction of Electronic Equipment.*

Part 2:

Compute the overall system reliability of payload by way of constructing a Reliability Block Diagram.

- Reliability modelling breaks a system into its functional blocks and model them as series, parallel or complex configurations, depending on their interdependencies.
- Each functional block has a reliability figure calculated by summing up the failure rates of the components that makes up the block.

Reliability Block Diagram

RBD drawn to reflect all possible scenario paths in the PPU operation.

There are two nodes in the PPU RBD

Upon reaching the first node, it means that there are at least one FPGA, one bank of serial flash, one CAN and one LVDS link operational.

Each of the 20 Paths to the second Node, corresponds to a successful scenario of a StrongARM processor being able to boot up healthily, and with access to at least one CAN or LVDS links.

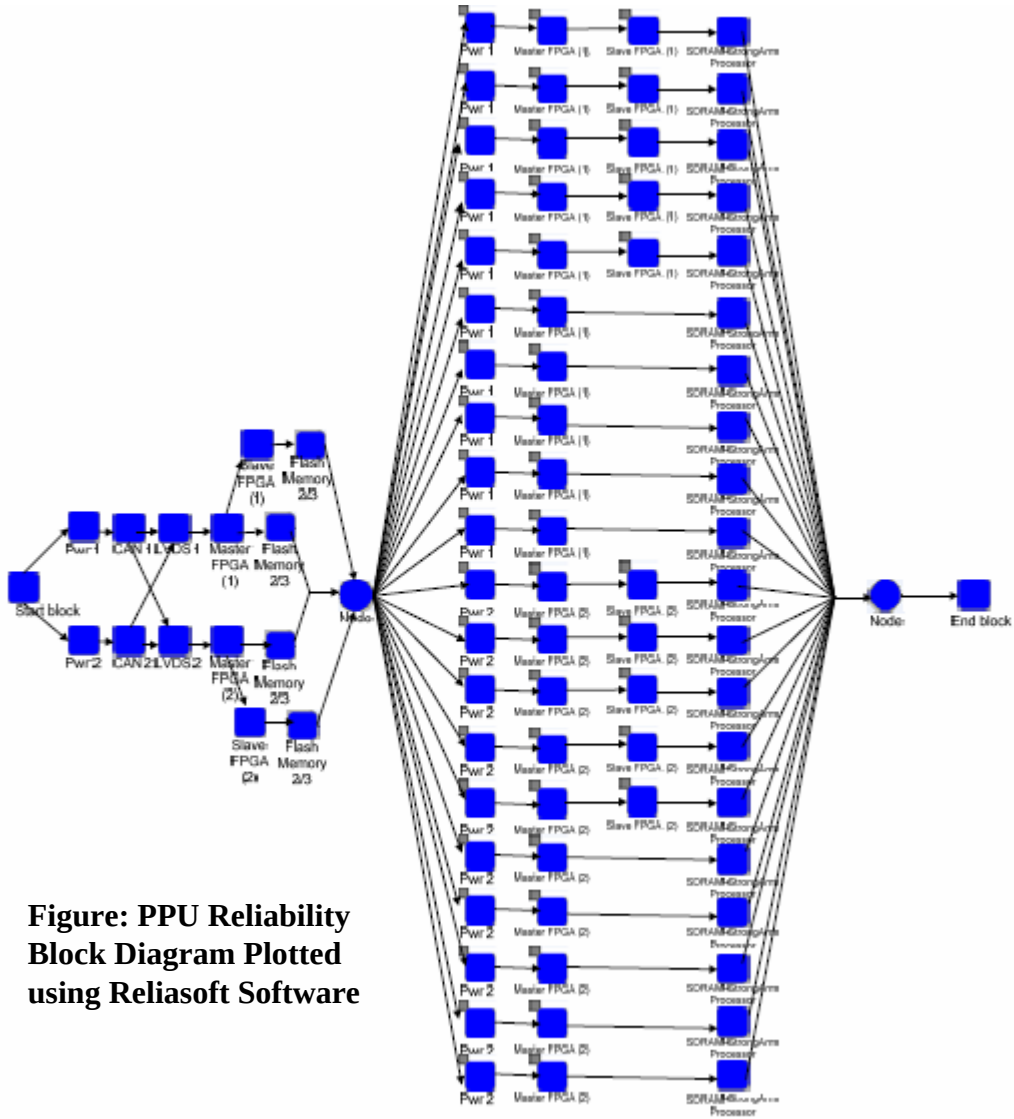
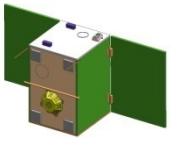


Figure: PPU Reliability Block Diagram Plotted using Reliasoft Software



Reliability Block Diagram

CREST

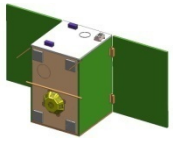
By plotting the probability of N successful paths to the node, the probability of PPU operating with N healthy StrongArm processors can be obtained

Table 1: Probability of N number of SA1110s in Operation in PPU (duty cycle of 10%)

	Failure Rate				Reliability
N Number of Healthy Operating SA1110s	Active	Dormant	Duty Cycle	Effective Failure Rate	Current Design
1	**0.9517**	**0.0952**	**10**	**0.1808**	**0.9953**
2	0.9750	0.0975	10	0.1853	0.9951
3	0.9790	0.0979	10	0.1860	0.9951
4	**0.9907**	**0.0991**	**10**	**0.1882**	**0.9951**
5	1.0260	0.1026	10	0.1949	0.9949
6	1.1980	0.1198	10	0.2276	0.9940
7	1.4030	0.1403	10	0.2666	0.9930
8	2.3990	0.2399	10	0.4558	0.9881
9	4.6487	0.4649	10	0.8833	0.9771
10	8.0418	0.8042	10	1.5279	0.9606
11	10.139	1.0139	10	1.9264	0.9506
12	10.283	1.0283	10	1.9538	0.9500
13	10.674	1.0674	10	2.0281	0.9481
14	11.330	1.1330	10	2.1527	0.9450
15	12.979	1.2979	10	2.4660	0.9372
16	**16.994**	**1.6994**	**10**	**3.2289**	**0.9186**
17	25.540	2.5540	10	4.8526	0.8803
18	42.198	4.2198	10	8.0176	0.8100
19	72.494	7.2494	10	13.7739	0.6963
20	128.80	12.8800	10	24.4720	0.5256

Table 2: Probability of N number of SA1110s in Operation in PPU (duty cycle of 100%)

	Failure Rate				Reliability
N Number of Healthy SA1110s	Active	Dormant	Duty Cycle	Effective Failure Rate	Current Design
1	**0.9517**	**0.0952**	**100**	**0.9517**	**0.9753**
2	0.9750	0.0975	100	0.9750	0.9747
3	0.9790	0.0979	100	0.9790	0.9746
4	**0.9907**	**0.0991**	**100**	**0.9907**	**0.9743**
5	1.0260	0.1026	100	1.0260	0.9734
6	1.1980	0.1198	100	1.1980	0.9690
7	1.4030	0.1403	100	1.4030	0.9638
8	2.3990	0.2399	100	2.3990	0.9389
9	4.6487	0.4649	100	4.6487	0.8850
10	8.0418	0.8042	100	8.0418	0.8095
11	10.139	1.0139	100	10.1390	0.7661
12	10.283	1.0283	100	10.2830	0.7632
13	10.674	1.0674	100	10.6740	0.7554
14	11.330	1.1330	100	11.3300	0.7425
15	12.979	1.2979	100	12.9790	0.7110
16	**16.994**	**1.6994**	**100**	**16.9940**	**0.6398**
17	25.540	2.5540	100	25.5400	0.5111
18	42.198	4.2198	100	42.1980	0.3299
19	72.494	7.2494	100	72.4940	0.1488
20	128.80	12.8800	100	128.8000	0.0339



X-SAT

PPU Reliability Figure Comparison CREST

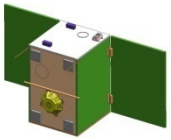
Based on a duty cycle of 10% as would be the case for PPU Mission Operations, reliability figures are as shown.

- Survival Probability = 0. 9953
- Nominal Performance Probability = 0.9951
- Peak Performance Probability = 0.9185

However based on a duty cycle of 100%, there is a drastic drop in reliability figure for the Peak Performance State

- Survival Probability = 0. 9753
  (This is comparable to the probability of Xsat Onboard Data Handler, a dual redundant ERC32 OBC and its I/O interfaces modules of 0.976)
- Nominal Performance Probability = 0.9743
- Peak Performance Probability = 0.6398

- Several design improvements can be made to increase reliability figure.



Conclusion

- PPU is a COTS Parallel computing payload that has an electronic hardware reliability figure > 0.9 for 3700 Dhrystone 2.1 MIPS of computation power for Peak Performance State (10% duty cycle).

- It is a software reconfigurable payload, capable of uploading new applications on the fly, and for testing the effectiveness of different software fault tolerant schemes.

- It adds value to the XSat mission by providing it with the means to run computation-intensive image processing algorithms to maximise useful image data throughput from the satellite.