# A Software and Hardware Redundancy Architecture for Using Raspberry Pi Modules as Command & Data Handling Systems for the DESCENT Mission

Latheepan Murugathasan[1], Udai Bindra[2], Chonggang Du[3], Zheng H. Zhu[4] and Franz T. Newland[5]
York University, Toronto, Ontario, M3J 1P3, Canada

## INTRODUCTION

CubeSats, which are a special class of nanosatellites, often impose very strict form-factor, schedule and cost constraints on their development. University projects may also impose limitations on human resources, in terms of team size, turnover and level of training. Low-cost commercial off the shelf (COTS) components are well suited to low-budget training activities and, would be ideal as replacements for more costly space-grade hardware. Their lack of space heritage demands rigorous testing and more complex design to ensure sufficient reliability for mission success, however. Traditionally, such assessment has focused on radiation and environmental performance [3-6], however other aspects of design and testing are also important to ensure a low-cost, robust and easily implementable system design. The command and data handling unit (C&DH) is of particular interest because there are many COTS alternatives available. With the advent of the System on a Chip (SoC), the cost of C&DH modules can be reduced by orders of magnitude while providing comparable if not superior computational performance and power consumption as well as reductions in volume.

In this paper, we provide details of one of the C&DH units for the upcoming DESCENT (2U CubeSat) mission. This C&DH unit uses two redundant Raspberry Pi Zeros in a novel system architecture. The paper details the design of the PCB hosting the two Pi Zeros that can be directly interfaced with the

[1] PhD Student, Department of Earth and Space Science, Email: lat123@yorku.ca
[2] PhD Student, Department of Earth and Space Science, Email: Bindra.udai@gmail.com
[3] PhD Student, Department of Earth and Space Science, Email: dudu1990@yorku.ca
[4] Professor, Department of Mechanical Engineering, Email: gzhu@yorku.ca
[5] Assistant Lecturer, Department of Earth and Space Science, Email: franz.newland@lassonde.yorku.ca

standard PC104 stack, typically found on CubeSats, as well as an arbitration logic to determine which Raspberry Pi should assume control of the CubeSat bus. The arbitration logic and memory archiving approach have been designed to increase the reliability of the low-cost Pi to a point where it can provide mission support for the short-duration DESCENT mission. The paper also discusses the synergies of this architecture with more traditional OBC architectures and the resulting development and training benefits of the low-cost Pi architecture for the DESCENT mission team.

The Deorbiting Spacecraft using Electrodynamic Tether (DESCENT) mission is a 2U university CubeSat project that aims verify the effectiveness of space tethers to deorbit spacecrafts at the end-of-life (EOL). The figure below depicts the space segment of the mission which consists of two 1U CubeSats attached by a 100m long bare tape tether. The purpose of the tether is to collect electrons from the ambient plasma. A Spindt Field Emitter Array (Spindt Array) is placed inside the Daughter cube (1U cube orbiting at a lower altitude with the respect to the other 1U cube (Mother)) which is used to emit the electrons collected by the tether back into the ambient plasma thus forming a current loop between the tether system and the space plasma [7-9]. The current flowing through the tether interacts with the Earth's magnetic field to produce a Lorentz force which can be used generate a delta-V and perform orbital maneuvers such as deorbits.

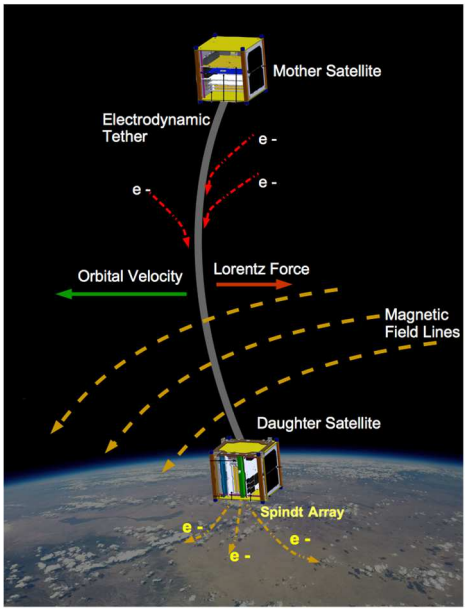


**Figure 1. DESCENT Mission Space Segment**

The primary mission objectives are to deploy and generate a current through the 100m tape tether [1]. The generation of current can only be verified via a measurement on the Daughter satellite and as a result, the system was designed to achieve the primary objectives through successful operation of this cube alone. A consequence of this decision is the relatively large allocation of the financial budget to procure space qualified components with extensive space heritage to ensure mission success. The residual was then only sufficient to procure a space grade power supply, battery and solar panels for the Mother satellite. As such, COTS components were considered to fulfill the secondary objectives and remaining goals of the mission.

## POTENTIAL COTS ALTERNATIVES FOR C&DH

There is an extensive list of commercially available microcontrollers and SoCs on the market for a substantially reduced cost relative to the space qualified counterparts. For the DESCENT mission, SoCs

were chosen for ease of development however, a similar study can be conducted for microcontrollers which are typically more appropriate for space systems.

The Raspberry Pi Zero and BeagleBone Black are two popular “single board computers” in the open-source community which are essentially break-out boards for their respective SoCs. Table 1 provides a comparison of the two outlining the features, performance and other relevant information [10,11]. In terms of computational resources and available interfaces the two are very similar and have comparable price points.

| Feature | Raspberry Pi Zero | BeagleBone Black |
|---|---|---|
| Cost | $5 | $55 |
| SoC | Broadcom BCM2835 | TI AM3358 |
| CPU | 1 GHz ARM11 | 1 GHz Cortex-A8 |
| Memory | 512MB | 512MB |
| Storage | MicroSD | 4GB eMMC (On-board) |
| Peripherals | GPIO, UART, I2C, SPI, PWM | GPIO, UART, I2C, SPI, PWM, LCD, CAN bus, Timers, LCD, GPMC, MMC1 |
| Power Consumption | 1.75 W (Max.)<br>0.5 W (Restricted) | 2.3 W (Max.)<br>0.8 W (Restricted) |
| Size | 65 mm x 30 mm x 5 mm | 86.4 mm x 53.3 mm |
| Weight | 9 g | 39.68 g |

**Table 1. Comparison of COTS Alternatives**

The Pi Zero was ultimately chosen as the C&DH for the Mother satellite of the DESCENT mission because of its reduced power consumption when underclocked and disabled peripherals. Only unnecessary peripherals were disabled such as HDMI and ethernet, but they were sufficient to provide a significant reduction in power consumption. Also, the Pi Zero’s form factor was very appealing as the PC104 standard can accommodate two of them allowing for cold redundancy and a more resilient architecture.

## EXISTING C&DH SYSTEMS

A preliminary study was conducted to compare some existing on-board computers in the market as shown in Table 2 [10,12-14]. The COTS alternative can provide a significant reduction in cost and improvements in terms of computational resources but, lacks space heritage and is not optimized in terms of power consumption. Rigorous testing needs to be conducted to evaluate performance of the COTS alternative under thermal fluctuations and continuous exposure to radiation, but the benefits appear promising. Finally, the current choice SoC does not provide real-time capabilities "out-of-the box" but can be augmented with a microcontroller such as a PIC to perform real-time duties. It is also important to note that ESA has an Astro Pi competition in which high school students compete to have their code executed on a raspberry pi that is inside the International Space Station (ISS) [2]. Although it is not exposed to the harsh space environment directly, it is a positive sign as it was able to survive the launch conditions.

| Feature | Proposed C&DH | ClydeSpace OBC | GOMSpace OBC | Innovative Solutions In Space (ISIS) OBC |
|---|---|---|---|---|
| Cost (USD) | $5 | Unavailable | Unavailable | ~5400-12000 |
| SoC | Broadcom BCM2835 | Smart Fusion 2 | N/A | N/A |
| CPU/MCU | 1 GHz ARM11 32-bit | 50 MHz Cortex-M3 32-bit | 64 MHz AVR32 | 400 MHz 32-bit ARM9 |
| Storage | 512MB RAM + MicroSD | 4GB Flash + 8MB MRAM | 512KB Flash + 128MB NOR Flash + 32KB FRAM + 32 MB SDRAM | 64MB RAM + 4GB SD + 256KB FRAM |
| Peripherals | GPIO, UART, I2C, SPI, PWM | GPIO, UART, I2C, SPI, RTC, CAN-Bus, JTAG, 3-axis magnetometer & gyro | GPIO, UART, I2C, SPI, RTC, CAN-Bus, ADC, 3-axis magnetometer & gyro, PWM | GPIO, UART, I2C, SPI, RTC, JTAG, ADC, PWM, USB, Image Sensor Interface (CMOS) |

| | | | | |
|---|---|---|---|---|
| Power Consumption | 1.75 W (Max.)<br>0.5 W (Restricted) | 1 W (Max.)<br>0.4 W (Typ.) | 0.9 W (Max.)<br>0.17 W (Typ.) | 0.55 W (Max.)<br>0.4 W (Typ.) |
| Size | 65 mm x 30 mm x 5 mm | 95.89 mm x 90.17 mm (PC104)<br>12 mm Thickness (6 mm from center) | 65 mm x 40 mm x 7.1 mm | 96 mm x 90 mm (PC104) x 12.4 mm |
| Weight | 9 g | 61.9 g | 24 g | 94 g |
| Operating Temp. | -40°C to 85°C (SoC) | -40°C to 80°C | -30°C to 85°C | -25°C to 65°C |
| Flight Heritage | No | Yes | Yes | Yes |
| Protection Circuits | Unavailable | EDAC (Memory and peripheral FIFOs), Autonomous single event latch-up protection, Integrated Cache and Memory Protection Unit, Hardware Watchdog | Unavailable | Redundant SD card storage with FailSafe FAT journaling file system, Redundant RTC, External hardware watchdog |

**Table 2. Comparison of Existing C&DH Modules**

## HARDWARE DESIGN

PC104 has become the standard form factor for CubeSat PCBs. We have chosen to place two Raspberry Pi Zero's in cold redundancy on a single PCB to increase the reliability of the C&DH subsystem. Cold redundancy was chosen for this mission due to constraints on the power budget. During mission analysis, it was estimated that the Mother satellite can only generate approximately 1 W of power. Given that each Pi consumes roughly 0.5 W, warm and hot redundancy is just not feasible.

The Pi Zero's GPIO header was used to break out the pins to the PC104 Stack header which gives the Pi Zero's access to the CubeSat bus. Cold redundancy results in every line from the PC104 Stack (CubeSat bus) be routed to both Pi Zero's, except for the power supply input. The power supply input for each Pi Zero is connected to isolated switchable lines which provides control of the boot priority and

sequence. The decision to assign priority and alter sequence will be computed on-orbit through arbitration logic which is discussed in detail in the Software Design section.

One of the main concerns with the cold redundant architecture is potential conflicts on the communication buses. For the Mother satellite, there are three main communication buses, I2C, SPI and UART. The Raspberry Pi Zero has only one I2C and UART lines available and two chip selects for the SPI line. Additional lines may be made available by reconfiguring some of the unused GPIO pins and creating software versions of these buses (via bit banging) however, the reliability, error and data rates may be compromised. Both I2C and SPI work in Master-Slave configurations and unfortunately, the Raspberry Pi Zero does not support multi-master configurations of either bus. This raises a major concern since the power supply unit is commanded through the I2C line and expects a periodic watchdog command. If both Pi Zero's simultaneously attempt to communicate on this bus, there is a high probability/risk of failure. A hardware solution to this problem can be achieved through a multiplexer but, for this mission we have chosen a simpler approach in software which is outlined in the next section. UART lines in general are not considered to be multidrop which implies that only one device can communicate on the line. Therefore, the only alternative is to implement a software equivalent and depending on the application, may be a feasible solution.

Arbitration logic has been implemented on the Pi Zeros to determine the boot priority and sequence. The logic is achieved by having each Pi Zero compute and exchange respective performance functions. Details of this function is outlined in the next section but due to the limitations in available hardware lines, we have opted to have the two Pi Zeros communicate over TCP/IP by reconfiguring the USB port as an Ethernet controller. There are two main advantages of using this route, the first is that TCP/IP is a widely used protocol, and there are many open-source libraries available to allow communication. The second advantage is the ability to send SSH commands, specifically the Secure Copy

Protocol (SCP) command which is an easy and convenient method for transferring large amounts of data. One of the challenges however, is that pre-flight, the USB bus (reconfigured to Ethernet) is the only channel available to upload new software and for communication during ground testing. But, like UART, the USB bus is not in general, compatible with multidrop. For this application, a software alternative is not available, and we have decided to use USB multiplexers. Each Pi Zero has a multiplexer that switches between the other Pi Zero and its respective external connector.

The figures below depict the schematic and layout of the custom designed PCB. There are a few additional connectors on the sides which interface with the external connectors of the Mother satellite. There are also two camera connectors to the side of each Pi Zero that are used to reduce complexity and routing of cabling. In figure 2, green copper traces can be seen connecting the two Pi Zeros together and the red copper traces connect the Pi Zeros to the PC104 Stack. The green and red traces are on separate copper layers and are connected by vias (yellow).

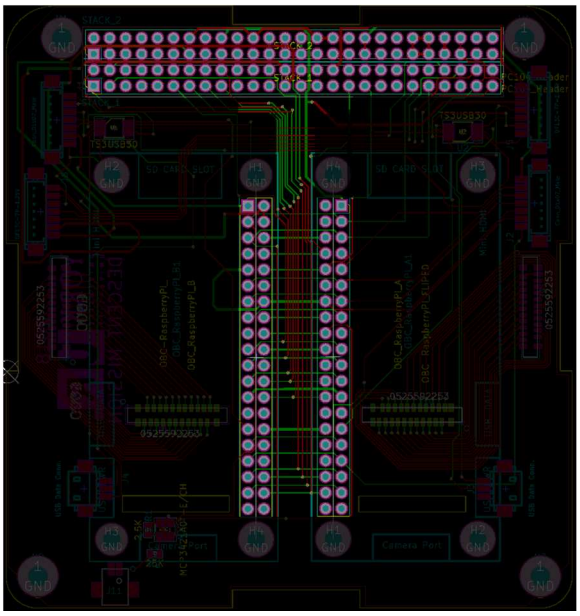

**Figure 2. Connection between Pi Zeros**

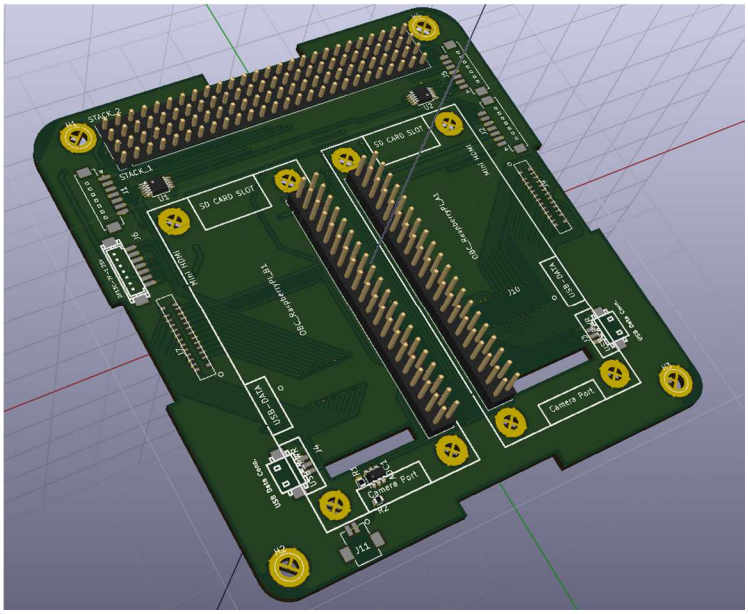


**Figure 3. 3D View of C&DH Unit**

## SOFTWARE DESIGN

Arbitration amongst the two Pi Zeros was included to reduce the risk of potential conflicts on the communication buses and, ensure a more reliable and smooth transition from one Pi Zero to the next. This provides the basis of a more intelligent and robust cold redundant architecture that can be easily adapted and improved. The premise here is that after system boot, each Pi Zero computes a performance function, which is an indication of its health, and upon comparison with its counterpart, will be given authority to command the spacecraft bus. The function is computed as follows,

$$P = Operational\ Capability + Operational\ Priority$$

Operational capability is a metric used to quantify the health of the communication buses. It is designed such that each parameter defined within this metric has a binary value and the parameters are ordered in terms of relative priority thus forming an n-bit integer. For the DESCENT mission we have chosen 10 whose priority is shown in Table 3 and described in detail in Table 4. Operational priority is designed to

give preference to the Pi Zero that has accumulated more flight time. This would ensure continuity of mission operations and hinder frequent/multiple switching between Pi Zeros. The parameters of operational priority are also described in Table 4. The Pi Zero with a greater *P*, will then assume control and command the EPS to switch off the supply to its counterpart to prevent inadvertent power consumption. There is a time delay associated with the EPS command to allow the other Pi Zero to safely shutdown.

| Bit | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| Value | I2C Status | I2C Error Check | SPI 1 Status | SPI 1 Error Check | Software UART Status | Software UART Error Check | UART Status | UART Error Check | SPI 2 Status | SPI 2 Error Check |

**Table 4. Priority of Operational Capability Parameters**

| Group | Parameter | Value/Range | Explanation/Justification |
|---|---|---|---|
| Operational Capability | I2C Status | Binary | Indicates whether I2C is operational or defective. Highest priority because power supply communicates on I2C and expects periodic watchdog commands. Power supply will reset if watchdog command is not received |
| | I2C Error Check | | Indicates if the command sent/received has errors. Justification for priority is similar to above |
| | SPI 1 Status | | Operational status of the wireless communication module required to communicate with Daughter satellite |
| | SPI 1 Error Check | | Checks for errors on this communication bus |
| | Software UART Status | | Operational status of the wired communication link with the Daughter satellite |

| | | | |
|---|---|---|---|
| | Software UART Error Check | | Verifies checksum of received data |
| | UART Status | | Indicates successful communication with the GPS module |
| | UART Error Check | | Checks for appropriate response to commands from GPS module |
| | SPI 2 Status | | Indicates operational or defective secondary payload |
| | SPI 2 Error Check | | Verifies checksum of received data |
| Operational Priority | Master Index Count | [0,1] | Count of number of arbitration wins. Normalized as follows $\frac{n-1}{n}$. Where n is the count. |
| | Total Time In Control | | Counter which is incremented periodically when in control. Normalized to the total expected lifetime of the mission. |
| | Data Stored | | Computes size of Telemetry and Payload Data. Normalized to the maximum available storage onboard. |

**Table 3. Performance Function Parameters**

To circumvent conflicts on the communication buses, a software solution was chosen for simplicity. The Linux operating system is supported by the developers of the Pi Zero as well as the open-source community. One of the advantages of this operating system is the ability to dynamically load/unload kernel modules (device drivers) without a system reboot. Thus, we can temporarily disable the I2C and SPI lines of each Pi dynamically to coordinate communication on each line respectively. An important note is that although the device drivers have been disabled, the physical pins do not get affected. I2C and SPI use “open-drain” and “push-pull” drivers and disabling the kernel module does not affect this configuration. Preliminary tests have not shown this to be an issue/concern however, the BCM2835 (SoC of the Pi Zero) can assign alternative configurations for each GPIO pin including I2C and

SPI lines dynamically without reboot. Future work could consider analyzing the alternate configurations and determine any advantages.

To minimize the impact of alternating Pi Zeros on mission operations, telemetry and payload data is periodically synced between the two. This ensures that if one of the Pi Zeros were to fail, the other Pi Zero would be able continue relatively seamlessly. As mentioned earlier, the network connection between the two Pi Zeros provides a convenient method to copy large amount of data effectively and efficiently. For the DESCENT mission, we have scheduled data backup once every day to minimize power consumption as having both Pi Zeros powered on simultaneously depletes a significant portion of our power budget.

## CONCLUSION

A cost-effective alternative to traditional space-grade C&DH units is proposed. An intelligent cold redundant architecture is implemented to increase the reliability and compensate to some extent for the lack of space heritage. Comparisons between existing on-board computers and the proposed architecture is shown and appears to have clear advantages in terms of processing power and form factor. Future work may consider rigorous environmental testing along with the inclusion of some protection circuitry such as latching current limiters and memory redundancy.

## REFERENCES


[1] U. Bindra, L. Murugathasan, V. Jain, G. Li, J. Kang, C. Du, Z. H. Zhu, F. T. Newland, M. Alger, O. Shonibare and A. de Ruiter, "DESCENT: Mission Architecture and Design Overview," in *AIAA Space and Astronautics Forum and Exposition*, Orlando, 2017.

[2] D. Honess and O. Quinlan, "Astro Pi: Running your code aboard the International Space Station," *Acta Astronautica,* vol. 138, pp. 43-52, 2017.

[3] M. Rousselet, P. C. Adell, D. J. Sheldon, J. Boch, H. Schone and F. Saigne, "Use and benefits of COTS board level testing for radiation hardness assurance," in *Proceedings of the European Conference on Radiation and its Effects on Components and Systems*, Bremen, 2017.

[4] P. Villa, E. Bezerra, R. Goerl, L. Poehls, F. Vargas, N. Medina, N. Added, V. De Aguiar, E. MacChione, F. Aguirre and M. Da Silveira, "Analysis of COTS FPGA SEU-sensitivity to combined effects of conducted-EMI and TID," in *Proceedings of the 2017 11th International Workshop on the Electromagnetic Compatibility of Integrated Circuits*, St. Petersburg, 2017.

[5] B. Buergler, B. Faure, D. Latif, L. Diblik, P. Vasina, V. Gineste and M. Simcak, "Towards Supercapacitors in Space Applications," in *11th European Space Power Conference*, Porto PalaceThessaloniki, 2017.

[6] J. Carron, A. Philippon, L. S. How, A. Delbergue, S. Hassanzadeh, D. Cillierre, P. Danto and M. Boutillier, "Cryogenic characterization of LEDs for space application," in *16th International Conference on Solid State Lighting and LED-Based Illumination Systems*, San Diego, 2017.

[7] J. R. Sanmartin, E. C. Lorenzini and M. Martinez-Sanchez, "Electrodynamic Tether Applications and Constraints," *Journal of Spacecraft Rockets,* vol. 47, no. 3, pp. 442-456, 2010.

[8] H. Wen, D. P. Jin and H. Y. Hu, "Advances in dynamics and control of tethered satellite systems," *Acta Mech Sin,* vol. 46, pp. 229-241, 2008.

[9] K. Xie, C. C. Farnell and J. D. Williams, "The plasma properties and electron emission characteristics of near-zero differential resistance of hollow cathode-based plasma contactors with a discharge chamber," *Physics of Plasmas,* vol. 21, no. 8, 2014.

[10] "RASPBERRY PI ZERO," Raspberry Pi, [Online]. Available: https://www.raspberrypi.org/products/raspberry-pi-zero/. [Accessed 20 4 2018].

[11] "BeagleBone Black," BeagleBoard, [Online]. Available: https://beagleboard.org/black. [Accessed 20 04 2018].

[12] "Nanosatellite On Board Computer," Clyde Space, [Online]. Available: https://www.clyde.space/products/3-nanosatellite-on-board-computer. [Accessed 20 04 2018].

[13] "VERSATILE ON-BOARD COMPUTER FOR CUBE, NANO AND MICROSAT MISSIONS," GOMSpace, [Online]. Available: https://gomspace.com/Shop/subsystems/computers/nanomind-a3200.aspx. [Accessed 20 04 2018].

[14] "ISIS On Board Computer," Innovative Solutions In Space, [Online]. Available: https://www.isispace.nl/product/on-board-computer/. [Accessed 20 04 2018].