CubeSat Power Distribution Module User Manual

Document No.: CN-USM-5025-SWT

Issue: A

Date: 14/03/2011

	Name	Date	Signed
Author CS	Kevin Worrall	14/03/2011	
Approved	Stephen Charlick	14/03/2011	

Clyde Space Ltd.
The Helix Building, Kelvin Campus
West of Scotland Science Park,
Glasgow G20 0SP, UK.
t: + 44 (0) 141 946 4440

e: enquiries@clyde-space.com
w: www.clyde-space.com

Registered in Scotland No. SC285287
at 123 St Vincent Street Glasgow G2 5EA

Document Control

Issue	Date	Section	Description of Change	Reason for Change
A	14/03/2011	All	First Draft	

1 Contents

Document Control 2
1 Contents 3
Acronyms and Abbreviations 6
1 Introduction 8
1.1 Additional Information Available Online 8
1.2 Continuous Improvement 8
1.3 Document Revisions 8
2 Overview 9
3 Maximum Ratings 10
4 Electrical Characteristics 11
5 Handling & Storage 12
5.1 Electrostatic discharge 12
5.2 General handling 12
5.3 Shipping and storage 12
6 Materials & Processes 13
6.1 Materials used 13
6.2 Processes and procedures 13
7 System description 14
7.1 Block diagram 14
7.1.1 PDU and telemetry 15
7.1.2 Serial interfaces 16
7.1.3 I²C Buffer 16
7.2 Pin descriptions 17
7.3 Mass and mechanical configuration 21
8 Switches and analogue channels 22
8.1 Basic operation 22
8.2 Basic command overview 23
8.3 Command description 25
8.3.1 Return analogue channel 1-8 reading 26
8.3.2 Return analogue channel 9-40 reading 26
8.3.3 Return switch 1-24 current reading 26
8.3.4 Turn switch 1-24 ON 26
8.3.5 Turn switch 1-24 OFF 26
8.3.6 Set all switches 26
8.3.7 Return all switch status 27
8.3.8 Set initial switch state 27
8.3.9 Switch all switches to initial switch state 27

8.3.10 Status of switch and telemetry node 27
8.3.11 Clear watchdog reset status bit 28
8.3.12 Force watchdog reset of switch and telemetry node 28
9 Serial interfaces 29
9.1 Basic operation 29
9.2 Basic command overview 29
9.3 Command descriptions 30
9.3.1 Clear watchdog reset status bit 30
9.3.2 Clear serial interfaces serial buffer 30
9.3.3 Clear serial interfaces I²C buffer 31
9.3.4 XOR checksum of data in I²C buffer 31
9.3.5 Current number of bytes in I²C buffer 31
9.3.6 Current number of bytes in serial buffer 31
9.3.7 Retrieve I²C buffer 31
9.3.8 Clear overflow indication bits 31
9.3.9 Return status information from this serial interface 31
9.3.10 Send I²C buffer to serial port 32
9.3.11 Retrieve serial buffer 32
9.3.12 Send I²C data to buffer 32
9.3.13 Force watchdog reset of serial interface 32
9.3.14 Set baud rate for serial port 32
9.3.15 Return serial interfaces serial port setup 33
9.3.16 Set parity bit for serial port 34
9.4 Using the serial interface 34
9.4.1 Transmission to an attached serial device 34
9.4.2 Retrieving data from an attached serial device 35
10 Test 37
10.1 Short Check 37
10.2 Test set-up and equipment 37
10.3 Switches 39
10.3.1 Switch operation 40
10.3.2 Set all switches 41
10.3.3 Initial switch state 41
10.3.4 Switch current and trip setting test 41
10.3.5 Analogue channel testing 42
10.3.6 Other PDU related tests 43
10.4 Serial port testing 43
10.4.1 RS232 set-up 44
10.4.2 RS422 set-up 44
10.4.3 3V3 and 5V set-up 45
10.4.4 Serial device testing 45

11 Developer AIT 50
ID 51
Requirement 51
Procedure 51
Result (X) 51
Success criteria 51
Compliance 51
(pass / fail) 51
SYS-0030 51
12 Compatible systems 52
Appendix A 53
A.1 Board Specification 53
A.2 Board I2C Addresses 53
A.3 Switch Trip Points 54
A.4 Switch Current Telemetry Equations 54
A.5 Analogue Channels 1 to 8 Equations 55

Acronyms and Abbreviations

The following acronyms and abbreviations are used within this document.

PDM	Power Distribution Module
BCR	Battery Charge Regulator
PCM	Power Conditioning Module
PDM	Power Distribution Module
MPPT	Maximum Power Point Tracker
USB	Universal Serial Bus
ESD	Electro Static Discharge
TLM	Telemetry
EPS	Electrical Power System
EoC	End of Charge
AMUX	Analogue Multiplexer
ADC	Analogue to Digital Converter
AIT	Assembly, Integration and Testing
3U	3 Unit
DEPS	Deployed Electrical Power System
rh	Relative Humidity
DoD	Depth of Discharge
$Kbits^{-1}$	Kilobits per second
Voc	Open Circuit Voltage
Isc	Short Circuit Current

All units in this document are in SI format unless otherwise stated.

#	Warning	Risk
1	Ensure headers H1 and H2 are correctly aligned before mating boards	If misaligned, battery positive can short to ground, causing failure of the battery and PDM
2	Ensure switching configuration is implemented correctly before applying power to PDM	If power is applied with incorrect switch configuration a failure of the PDM may occur
3	Observe ESD precautions at all times	The PDM is a static sensitive system. Failure to observe ESD precautions can result in failure of the PDM.
4	Ensure not to exceed the maximum stated limits	Exceeding any of the stated maximum limits can result in failure of the PDM
5	Ensure batteries are fully isolated during storage	If not fully isolated (by switch configuration or separation) the battery may over-discharge, resulting in failure of the battery
6	No connection should be made to H2.35-36	These pins are used to connect the battery to the PDM. Any connections to the unregulated battery bus should be made to pins H2.43-44
7	H1 and H2 pins should not be shorted at any time	These headers have exposed live pins which should not be shorted at any time. Particular care should be taken regarding the surfaces these are placed on.

1 Introduction

This document provides information on the features, operation, handling and storage of the Clyde Space Power Distribution Module (PDM). The PDM is designed to allow the user to control the distribution of power throughout a platform and transfer various serial data streams onto the main platform I^2C bus. A basic block diagram representation of the system is shown in Figure 1-1.

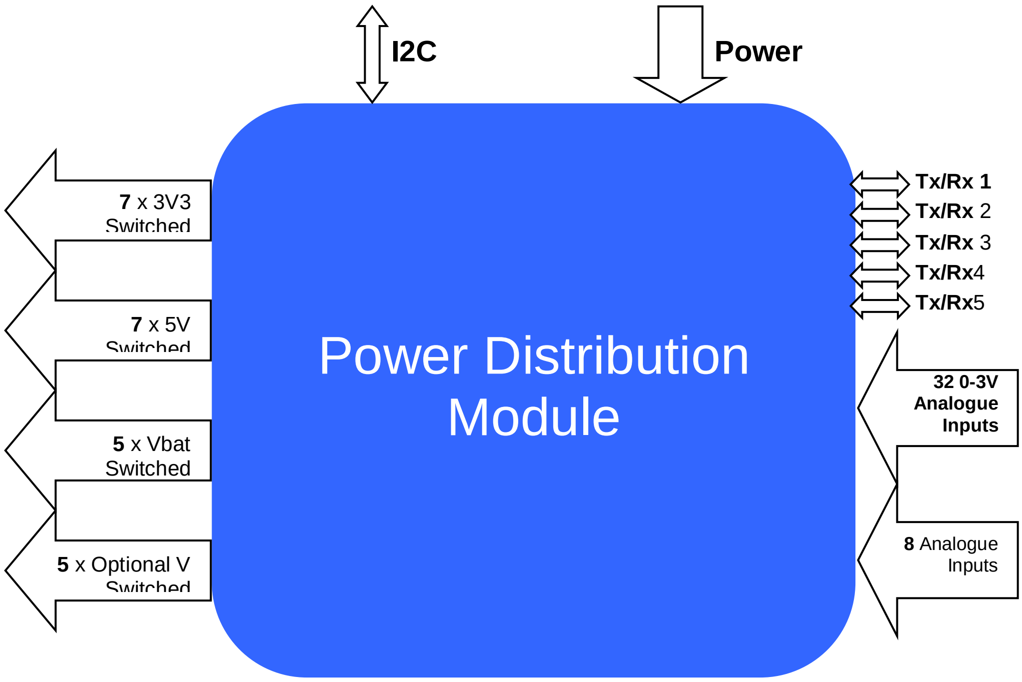

Figure 1-1 System Diagram

1.1 Additional Information Available Online

Additional information on CubeSats and Clyde Space Systems can be found at www.clyde-space.com. You will need to login to our website to access certain documents.

1.2 Continuous Improvement

Clyde Space is continuously improving its processes and products. We aim to provide full visibility of changes and updates, and this information can be accessed by logging in to www.clyde-space.com.

1.3 Document Revisions

In addition to hardware and software updates, we also make regular updates to our documentation and online information. Notes of updates to documents can also be found at www.clyde-space.com.

2 Overview

This is the second generation of the Clyde Space Power Distribution Module, developed by our team of highly experienced Spacecraft Power Systems and Electronic Engineers.

Since introducing the first generation in 2008 the continuing experience of the engineers has allowed us to increase the resources on the board and provide a more user configurable board.

Clyde Space is the World leading supplier of power system components for CubeSats. We have been designing, manufacturing, testing and supplying batteries, power system electronics and solar panels for space programmes since 2006. Our customers range from universities running student led missions, to major space companies and government organisations.

3 Maximum Ratings

4

OVER OPERATING TEMPERATURE RANGE (UNLESS OTHERWISE STATED)			
		Value	**Unit**
Input Voltage	Battery (Pins H2.45/46)	8.3	V
	5V Bus (Pins H2.25/26)	5.05	V
	3.3V Bus (Pins H2.27/28)	3.33	V
	Optional (Pins H2.51/52)	Dependent on Board	V
		Voltages (V)	**Current (A)**
Output Voltages/Currents Switches	Switches 1-3	3.33	0.5
	Switches 4-6	3.33	1
	Switches 7	3.33	4
	Switches 8-10	5.05	0.5
	Switches 11-13	5.05	1
	Switches 14	5.05	4
	Switches 15-16	8.3	0.5
	Switches 17-18	8.3	1
	Switches 19	8.3	4
	Switches 20-21	Optional + 2.5%	0.25
	Switches 22-23	Optional + 2.5%	0.5
	Switches 24	Optional + 2.5%	1
		Voltage Levels	**Units**
Serial Links (RX)	LVTTL	3.33	V
	TTL	5.05	V
	RS232	Rx: -12 to 12	V
	RS422	Rx: -6 to +10	V
		Voltage Levels	**Units**
Analogue Channels	Channels 1-8	Dependent on Board	V
	Channels 9-40	3.00	V
Operating Temperature		-40 to 85	°C
Storage Temperature		-50 to 100	°C
Vacuum		10^{-5}	torr
Radiation Tolerance		TBC	kRad
Shock		TBC	
Vibration		TBC	

Table 3-1 Performance Characteristics of the PDM

(1) Stresses beyond those listed under maximum ratings may cause permanent damage to the PDM. These are the stress ratings only. Operation of the PDM at conditions beyond those indicated is not recommended. Exposure to absolute maximum ratings for extended periods may affect PDM reliability

4 Electrical Characteristics

Description	Conditions	Min	Typical	Max	Unit
Switches					
Voltage Output		Rated-5%	Rated	Rated+1%	V
Current				Rated	A
Analogue Channels					
Channels 1 to 8			Rated		V
Channels 9 to 40		0	-	3	V
Conversion			10		Bit
Conversion Error			1		LSB
3.3V Serial Link					
Rx: Low/High			0/3.3		V
Tx: Low/High			0/3.3		V
Speed		300	-	115200	Baud
5V Serial Link					
Rx: Low/High			0/3.3		V
Tx: Low/High			0/3.3		V
Speed		300	--	115200	Baud
RS232					
Rx: threshold Low/High		0/1.5	0.6/ --	1.2/2.4	V
Tx Outputs		±5	±5.4	--	V
Speed		300	--	115200	Baud
RS422					
Speed		300	--	115200	Baud
Differential: TX/RX		2000/-200	-- /-125	3300/-50	mV
I^2C Communications					
Protocol		--	I^2C	--	
Transmission speed		--	100		KBps
Bus voltage		3.26V	3.3V	3.33V	
Node address range		--	0x50-0x55	--	Hex
Address scheme		--	7bit	--	
Node operating frequency		--	10MHz	--	
Quiescent Operation					
Power Draw		--	0.16	--	W

Table 4-1 Performance Characteristics of the 3U EPS

5 Handling & Storage

The PDM requires specific guidelines to be observed for handling, transportation and storage. These are stated below. Failure to follow these guidelines may result in damage to the units or degradation in performance.

5.1 Electrostatic discharge

The PDM incorporates static sensitive devices and care should be taken during handling. Do not touch the PDM without proper electrostatic protection in place. All work carried out on the system should be done in a static dissipative environment.

5.2 General handling

The PDM is robust and designed to withstand flight conditions. However, care must be taken when handling the device. Do not drop the device as this can damage the PDM. There are live connections between the battery systems and the PDM on the CubeSat Kit headers. All metal objects (including probes) should be kept clear of these headers.

5.3 Shipping and storage

The devices are shipped in anti-static, vacuum-sealed packaging, enclosed in a hard protective case. This case should be used for storage. All hardware should be stored in anti-static containers at temperatures between 20°C and 40°C and in a humidity-controlled environment of 40-60%rh.

The shelf-life of this product is estimated at 5 years when stored appropriately.

6 Materials & Processes

6.1 Materials used

	Material	Manufacturer	%TML	%CVCM	%WVR	Application
1	Araldite 2014 Epoxy	Huntsman	0.97	0.05	0.33	Adhesive fixing
2	1B31 Acrylic	Humiseal	3.89	0.11	0.09	Conformal Coating
3	DC 6-1104	Dow Corning	0.17	0.02	0.06	Adhesive fixing on modifications
4	Stycast 4952	Emerson & Cuming	0.42	0.17	0.01	Thermally Conductive RTV
5	PCB material	FR4	0.62	0	0.1	Note: worst case on NASA out-gassing list
6	Solder Resist	CARAPACE EMP110 or XV501T-4	0.95 or 0.995	0.02 Or 0.001	0.31	-
7	Solder	Sn62 or Sn63 (Tin/Lead)	-	-	-	-
8	Flux	Alpha Rosin Flux, RF800, ROL 0	-	-	-	ESA Recommended

Table 6-1 Materials used

Part Used	Manufacturer	Contact	Insulator	Type	Use
ESQ-126-39-G-D	Samtec	Gold Plated	Black Glass Filled Polyester	PTH	CubeSat Kit Compatible Headers
DF13-10P-125DSA	Hirose	N/A	Polyamide	Crimp Housing	Analogue channel input

Table 6-2 Connector headers

6.2 Processes and procedures

All PCB assembly is carried out and inspected to ESA Workmanship Standards; ECSS-Q-ST-70-08C and ECSS-Q-ST-70-38C.

7 System description

The Clyde Space PDM is designed to enable the user to control the power distribution within their satellite. The board is also designed to interface with a wide range of Serial devices and transfer the data from these devices onto a single communication bus, without modification this is the Cubesat I^2C bus. Various applications exist for the PDM including power distribution, magnetorquer driving, current protection of vital devices and interfacing various off the shelf serial devices and serial peripherals to a single I^2C bus. The features of the PDM are:

- CubeSat and CubeSat Kit Compatible
- 7 x 3.3V Current Limited Switched Lines
- 7 x 5V Current Limited Switched Lines
- 5 x Raw V Current Limited Switched Lines
- 5 x Optional V Current Limited Switched Lines
    - 3.3V/5V/battery or other voltage. Other Voltage can be applied to pins H2.51-52. Contact Clyde Space to ensure optional voltage is compatible with the switches
- Current Feedback on all Switched lines
- All Switches lines can be User Commanded On and Off
- Status of all Switch lines available
- Telemetry channels for up to 32 3V Analogue signals and 8 Selected Voltage Analogue Signals
- 3 x 3.3V/5V Logic Level serial bus. Set at board build
- 1 x 3.3V/5V/RS422 Set at board build
- 1 x 3.3V/RS232 Set at board build
- All Serial Devices have user selectable baud rate
    - 300,1200,2400,4800,9600,19200,38400,57600,115200
- 8 bit data with user selectable Parity: None/Odd/Even
- Each Serial Device has an 80 Byte input buffer and an 80 Byte output buffer

This is the standard configuration of the board. User specific configurations can be found in Appendix A.

7.1 Block diagram

Figure 7.1 shows a block diagram representation of the system. Each of the sections highlighted in the block diagram will be introduced and there basic operation presented. The PDM can be split into two distinct sections; Power Distribution Unit (PDU) and telemetry section and serial interfaces section. The PDU and telemetry part of the board handles the 24 switched lines on the board and the 40 analogue channels that are on the board. The serial interface part is designed to allow the user to communicate with multiple external serial devices.

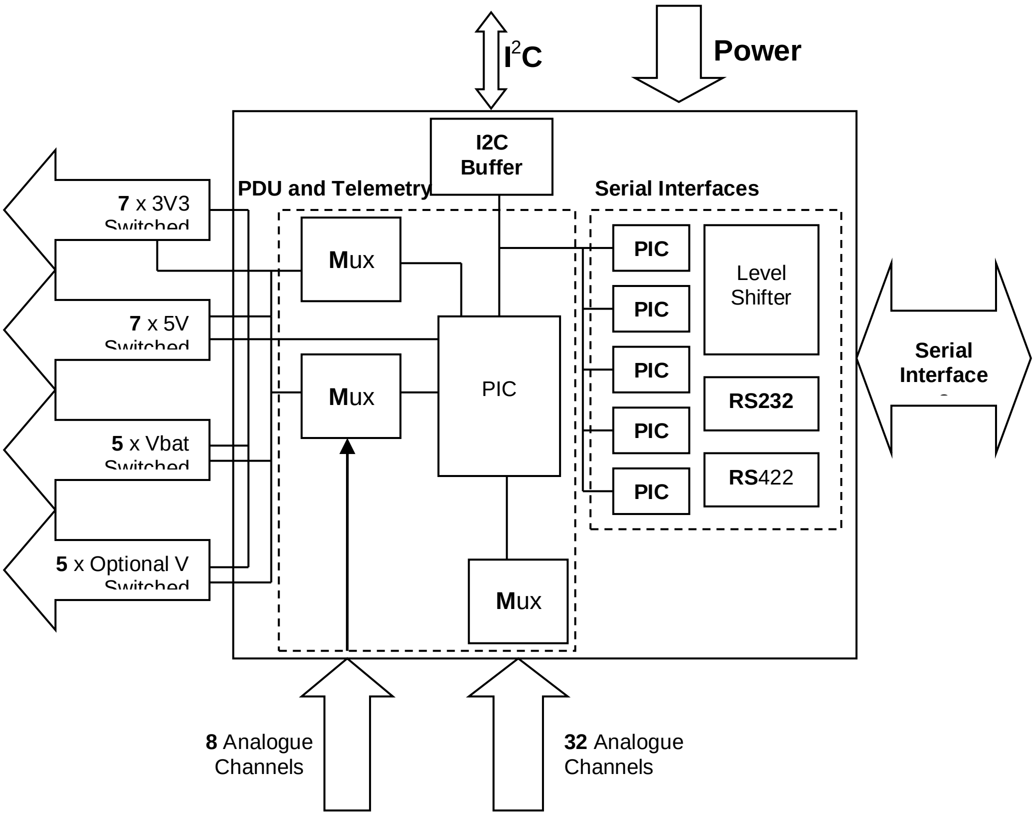

Figure 7-1 System Block Diagram

7.1.1 PDU and telemetry

There are 24 switched lines on the board. Each switch can be commanded on and off, has hardware set over current protection and can provide status telemetry and current telemetry. There are four voltage levels for the switches, 3.3V, 5V, Raw V and Optional V. The telemetry for each switch is supplied via analogue multiplexers that switch through each switched line to provide the relevant telemetry. As standard the switch trip points are shown below:

Switch Number	Voltage Level (V)	Recommended Current Trip (A)
1	3.3	0.5
2	3.3	0.5
3	3.3	0.5
4	3.3	1
5	3.3	1
6	3.3	1
7	3.3	4*
8	5	0.5
9	5	0.5
10	5	0.5

Switch Number	Voltage Level (V)	Recommended Current Trip (A)
11	5	1
12	5	1
13	5	1
14	5	4*
15	Battery	0.5
16	Battery	0.5
17	Battery	1
18	Battery	1
19	Battery	4*
20	12	0.25
21	12	0.5
22	12	0.5
23	12	1
24	12	1

Table 7-1 Switch details

The board can also interface up to 40 analogue signals. This is achieved via two analogue multiplexers. 32 of the analogue signals are required to be 0 to 3V signals. 8 of the analogue input signals can be requested to be different voltages. These 8 channels are channelled through a conditioning circuit. This conditioning circuit is required for the current signals from the switched lines and since the current signals and these 8 channels share the same input the 8 channels are required to be conditioned. The output from the ADC for the current signals and the 8 analogue channels will require to be converted into the true voltage by a conversion equation. These equations are provided in Appendix A.

7.1.2 Serial interfaces

The serial interfaces allow the connection of 5 external serial devices to a single I^2C bus. The serial interfaces that can be implemented on the board are 3.3V and 5V level signals, RS232 and RS422. The user can select the baud rate and the parity of the interface allowing a wider range of devices to be connected to the board. All serial interfaces have an 80 byte input buffer and an 80 byte output buffer. Each serial interface has a full range of commands that allow the user to queue commands to the serial device or send them individually. These are presented in Section 9.

7.1.3 I²C Buffer

The I^2C buffer is included to provide a robust I^2C bus on the board.

7.2 Pin descriptions

Since there are multiple inputs and outputs to this board much of the standard CubeSat header is taken up by pins associated with the Power Distribution Module. Figure 7.2 shows an image of the headers used on this board. The colours highlight different groups of functions. The pin functions are described next.

H1		H2	
1: Sw19	2: Sw19	1: Sw1	2: Sw2
3: Sw19	4: Sw22	3: Sw3	4: Sw4
5: Sw23	6: Sw24	5: Sw5	6: Sw6
7: A25	8: A26	7: Sw7	8: Sw7
9: A27	10: A28	9: Sw7	10: Sw8
11: A29	12: A30	11: Sw9	12: Sw10
13: -	14: A31	13: Sw11	14: Sw12
15: -	16: A32	15: Sw13	16: Sw14
17: A9	18: A33	17: Sw14	18: Sw14
19: A10	20: A34	19: Sw15	20: Sw16
21: A11	22: A35	21: Sw17	22: Sw18
23: A12	24: 36	23: Sw20	24: Sw21
25: A13	26: A37	25: 5V	26: 5V
27: A14	28: A38	27: 3.3V	28: 3.3V
29: A15	30: A39	29: GND	30: GND
31: A16	32: A40	31:AGND	32: GND
33: A17	34:RS232 Rx	33: -	34: -
35: A18	36 RS232 Tx	35: -	36: -
37: A19	38:Serial Rx1	37: -	38: -
38: A20	40:Serial Tx1	39: -	40: -
41: I2C_Data	42:Serial Rx2	41: -	42: -
43: I2C_Clk	44:Serial Tx2	43: -	44: -
45: A21	46:Serial Rx3	45: Battery	46: Battery
47: A22	48:Serial Tx3	47:RS422 RX A	48:RS422 TX A
49: A23	50:Serial Rx4	49:RS422 RX B	50:RS422 TX B
51: A24	52:Serial Tx4	51: Optional V	52: Optional V

Figure 7-2 Power Distribution Module Header

Table 7-2 below provides a brief description of the pin out for H1 and table 7-3 provides this for H2.

Pin	Name	Direction	Use
1 - 3	Switch 19	***OUT***	Switch 19 output - Rated for 4A
4	Switch 22	***OUT***	Switch 22 Out
5	Switch 23	***OUT***	Switch 23 Out
6	Switch 24	***OUT***	Switch 24 Out
7-12	A25 – A30	***IN***	Analogue Input Channels 25-30
13	-	-	-
14	A31	***IN***	Analogue Input Channel 31
15	-	-	-
16	A32	***IN***	Analogue Input Channel 32
17	A9	***IN***	Analogue Input Channel 9
18	A33	***IN***	Analogue Input Channel 33
19	A10	***IN***	Analogue Input Channel 10
20	A34	***IN***	Analogue Input Channel 34
21	A11	***IN***	Analogue Input Channel 11
22	A35	***IN***	Analogue Input Channel 35
23	A12	***IN***	Analogue Input Channel 12
24	A36	***IN***	Analogue Input Channel 36
25	A13	***IN***	Analogue Input Channel 13
26	A37	***IN***	Analogue Input Channel 37
27	A14	***IN***	Analogue Input Channel 14
28	A38	***IN***	Analogue Input Channel 38
29	A15	***IN***	Analogue Input Channel 15
30	A39	***IN***	Analogue Input Channel 39
31	A16	***IN***	Analogue Input Channel 16
32	A40	***IN***	Analogue Input Channel 40
33	A17	***IN***	Analogue Input Channel 17
34	RS232_Rx	***IN***	RS232 Receive Line. Can be 3.3V
35	A18	***IN***	Analogue Input Channel 18
36	RS232_Tx	***OUT***	RS232 transmit Line. Can be 3.3V
37	A19	***IN***	Analogue Input Channel 19
38	Serial_Rx1	***IN***	Serial Communication Receive Line. Can be 3.3V or 5V
39	A20	***IN***	Analogue Input Channel 20
40	Serial_Tx1	***OUT***	Serial Communication transmit Line. Can be 3.3V or 5V
41	I2C_Data	***IN/OUT***	I2C Data Line
42	Serial_Rx2	***IN***	Serial Communication Receive Line. Can be 3.3V or 5V
43	I2C_Clk	***IN***	I2C Clock Line
44	Serial_Tx2	***OUT***	Serial Communication transmit Line. Can be 3.3V or 5V
45	A21	***IN***	Analogue Input Channel 21
46	Serial_Rx3	***IN***	Serial Communication Receive Line. Can be 3.3V or 5V
47	A22	***IN***	Analogue Input Channel 22
48	Serial_Tx3	***OUT***	Serial Communication transmit Line. Can be 3.3V or 5V
49	A23	***IN***	Analogue Input Channel 23
50	Serial_Rx4	***IN***	Serial Communication Receive Line. Can be 3.3V or 5V
51	A24	***IN***	Analogue Input Channel 24
52	Serial_Tx4	***OUT***	Serial Communication transmit Line. Can be 3.3V or 5V

Table 7-2 Pin out for H1

Pin	Name	Direction	Use
1	Switch 1	***OUT***	Switch 1 out
2	Switch 2	***OUT***	Switch 2
3	Switch 3	***OUT***	Switch 3
4	Switch 4	***OUT***	Switch 4
5	Switch 5	***OUT***	Switch 5
6	Switch 6	***OUT***	Switch 6
7-11	Switch 7	***OUT***	Switch 7 out – Rated for 4A
10	Switch 8	***OUT***	Switch 8
11	Switch 9	***OUT***	Switch 9
12	Switch 10	***OUT***	Switch 10
13	Switch 11	***OUT***	Switch 11
14	Switch 12	***OUT***	Switch 12
15	Switch 13	***OUT***	Switch 13
16-18	Switch 14	***OUT***	Switch 14 out – Rated for 4A
19	Switch 15	***OUT***	Switch 15
20	Switch 16	***OUT***	Switch 16
21	Switch 17	***OUT***	Switch 17
22	Switch 18	***OUT***	Switch 18
23	Switch 20	***OUT***	Switch 20
24	Switch 21	***OUT***	Switch 21
25-26	5V_BUS	***IN***	5V Bus
27-28	3V3_BUS	***IN***	3.3V Bus
29-30	GND	***IN***	Return path for all buses
31	AGND	***IN***	AGND
32	GND	***IN***	Return path for all buses
33-44	-	**-**	Switch Connections
45-46	BATT_BUS	***IN***	Battery Bus
47	RS422_RX_A	***IN***	RS422 Rx Connection A
48	RS422_TX_A	***IN***	RS422 Tx Connection A
49	RS422_RX_B	***IN***	RS422 Rx Connection B
50	RS422_TX_B	***IN***	RS422 Tx Connection B
51-52	Optional V	***IN***	Optional Voltage. This line connects to the Optional switches

Table 7-3 Pin out for H2

In addition to the four RS422 connections on the main header a shield ground connection is provided. This connection should be used for the shield of the RS422 cable. On board this connection point goes through a 100Ohm resistor to chassis ground.

The main headers carry the 0-3V analogue inputs for analogue channels 9 to 40. The first 8 analogue signals are routed though a different header. This has been done to differentiate these signals from the other 32 analogue channels as these signals have different voltage levels. Please Contact Clyde Space with your requirements regarding these signals as these levels are set at time of manufacture. This header is a 10 pin Hirose male connector, part DF-10P-

125DSA which has mating half: DF13-10S-1.25C. This connection has a polarity. Table 7-4 shows the pin out for this connector.

Pin	Name	Direction	Use
1	A1	***IN***	Analogue Input Channel 1
2	A2	***IN***	Analogue Input Channel 2
3	A3	***IN***	Analogue Input Channel 3
4	A4	***IN***	Analogue Input Channel 4
5	A5	***IN***	Analogue Input Channel 5
6	A6	***IN***	Analogue Input Channel 6
7	A7	***IN***	Analogue Input Channel 7
8	A8	***IN***	Analogue Input Channel 8
9-10	GND	***OUT***	Ground Connection

Table 7-4 Pin out for Header 3

The locations of the headers are shown in Figure 7-4.

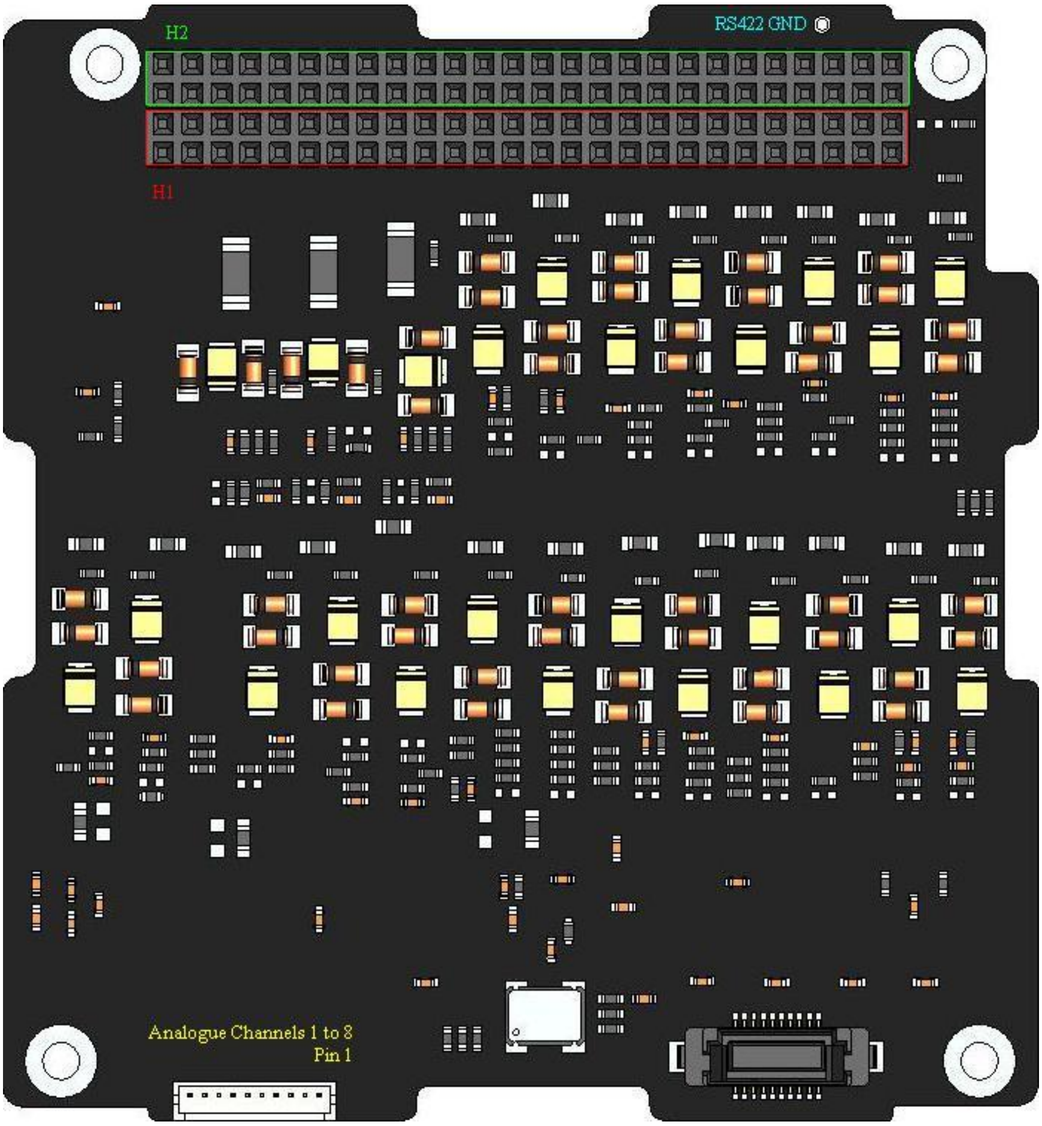

Figure 7-4 Header Locations

The additional header on the board is a programming header for the programmable devices on the board. This header should not be connected to nor should any attempt be made to connect to it as this may affect the programmable IC's on the board.

7.3 Mass and mechanical configuration

The mass and physical dimensions of the board can be found in Table 7-5. This section also provides an image of the mechanical configuration of the board.

	Min	Nominal	Max	Unit
Mass	59	60	61	g
Dimensions				**Unit**
L	90.0	90.81	91.0	mm
W	90.0	90.17	90.50	mm
H	24.50	24.839	25.0	mm

Table 7-5 Mass and Mechanical Configurations

The dimensions of the EPS, including all the connector locations, are given in Figure 7-5 below.

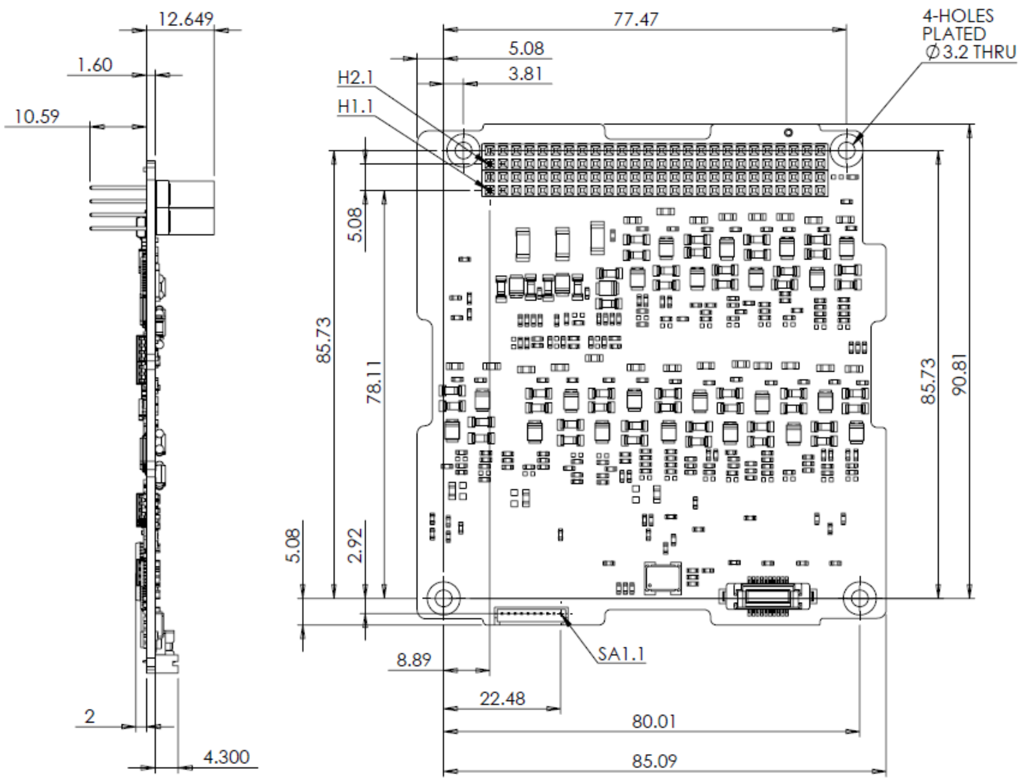

Figure 7-5 Board Dimensions (mm)

8 Switches and analogue channels

The PDM has two distinct sections, the switch and telemetry part and the serial interface part. The switch and telemetry part controls the switches and analogue channels. There exist 24 switches on the board. Each switch has a status line and a current sense. The switches can be commanded on and off and the status and current flow through the switch can be retrieved. Also on the board are 40 telemetry channels. Each channel has its own command to allow the user to retrieve the data.

To control the switches and access all the telemetry that is available on this part of the board the I^2C network is used to communicate with the microcontroller that implements this part of the circuit. The I^2C address to be used is 0x50 (80 in decimal). All the I^2C addresses of the board can be found in Appendix A. The I^2C address is a 7bit address.

8.1 Basic operation

The board is simple to use once the commands are known. On power up the switches will be set into an initial state. On receipt of the board this will be all switches OFF. If the user has a requirement for some switches to be on at power up this can be set by the user through a telecommand. The switches are simple to operate; the user can command them on or off, retrieve telemetry on whether the switch is on or off and retrieve telemetry on the current flowing through the switch. Each switch has a dedicated ON command, OFF command and current telemetry command. The status of every switch is sent back via one command. Every switch can also be turned ON and OFF at once, in reality a small delay exists between each switch, with the issue of one command, described in the next section.

In the OFF state the user will receive a status of 0 with a current telemetry showing no current is being drawn. When an ON command is sent the switch is turned on. The status returned to the user will be 1 and the current telemetry will return a value associated with the current flowing through the switch. The switch will remain on until the user issues an OFF command or the current flow through the switch exceeds the current trip point. If the switches current trip is exceeded then the switch switches off. The status returned by the switch will be 0 and the current telemetry will indicate that no current is flowing through the switch. To set the switch to an on state again, the user should issue a switch OFF command then a switch ON command. If the over current condition is removed the switch will turn on, however if the current through the switch still exceeds the trip point then the switch will turn off again.

The user can change the initial state of the switches; that is the state the switches power up in if the board resets or if the boards 3.3V is power cycled. There is also a command that allows the user to set all the switches into this defined initial state. This allows the user to set the switches in a known state if required. The command to do this is described in the next section.

The analogue channels on the board are accessed through the telecommands shown. Each channel returns a 10bit number which represents the signal attached to the channel.

To retrieve the 10bit number for any analogue channel, including the switch current telemetry, a command is sent to the node. This command instructs the node to read the chosen analogue input and carry out the analogue to digital conversion. After 1.2ms the node can then be read and the output is transferred. Once a conversion has begun *DO NOT* send any other command to this node as the analogue to digital conversion will be interrupted and the data lost. Once the data has been read out the node can be used as normal again. The operation of the analogue reads can be summarised as:

1. Send command to begin conversion on selected channel
2. Wait 1.2ms
3. Read the node, data will be transferred out
4. End of conversion

8.2 Basic command overview

The following table shows the commands for the switch and telemetry node on the board.

Command		Description	Data Range	Length (Bytes)	
Hex	Dec			Cmd	Return
01	1	Return Analogue Channel 1 Reading	NA	1	2
02	2	Return Analogue Channel 2 Reading	NA	1	2
03	3	Return Analogue Channel 3 Reading	NA	1	2
04	4	Return Analogue Channel 4 Reading	NA	1	2
05	5	Return Analogue Channel 5 Reading	NA	1	2
06	6	Return Analogue Channel 6 Reading	NA	1	2
07	7	Return Analogue Channel 7 Reading	NA	1	2
08	8	Return Analogue Channel 8 Reading	NA	1	2
09	9	Return Analogue Channel 9 Reading	NA	1	2
0A	10	Return Analogue Channel 10 Reading	NA	1	2
0B	11	Return Analogue Channel 11 Reading	NA	1	2
0C	12	Return Analogue Channel 12 Reading	NA	1	2
0D	13	Return Analogue Channel 13 Reading	NA	1	2
0E	14	Return Analogue Channel 14 Reading	NA	1	2
0F	15	Return Analogue Channel 15 Reading	NA	1	2
10	16	Return Analogue Channel 16 Reading	NA	1	2
11	17	Return Analogue Channel 17 Reading	NA	1	2
12	18	Return Analogue Channel 18 Reading	NA	1	2
13	19	Return Analogue Channel 19 Reading	NA	1	2
14	20	Return Analogue Channel 20 Reading	NA	1	2
15	21	Return Analogue Channel 21 Reading	NA	1	2
16	22	Return Analogue Channel 22 Reading	NA	1	2
17	23	Return Analogue Channel 23 Reading	NA	1	2
18	24	Return Analogue Channel 24 Reading	NA	1	2
19	25	Return Analogue Channel 25 Reading	NA	1	2
1A	26	Return Analogue Channel 26 Reading	NA	1	2
1B	27	Return Analogue Channel 27 Reading	NA	1	2
1C	28	Return Analogue Channel 28 Reading	NA	1	2
1D	29	Return Analogue Channel 29 Reading	NA	1	2
1E	30	Return Analogue Channel 30 Reading	NA	1	2
1F	31	Return Analogue Channel 31 Reading	NA	1	2

20	32	Return Analogue Channel 32 Reading	NA	1	2
21	33	Return Analogue Channel 33 Reading	NA	1	2
22	34	Return Analogue Channel 34 Reading	NA	1	2
23	35	Return Analogue Channel 35 Reading	NA	1	2
24	36	Return Analogue Channel 36 Reading	NA	1	2
25	37	Return Analogue Channel 37 Reading	NA	1	2
26	38	Return Analogue Channel 38 Reading	NA	1	2
27	39	Return Analogue Channel 39 Reading	NA	1	2
28	40	Return Analogue Channel 40 Reading	NA	1	2
29	41	Return Switch 1 Current Reading	NA	1	2
2A	42	Return Switch 2 Current Reading	NA	1	2
2B	43	Return Switch 3 Current Reading	NA	1	2
2C	44	Return Switch 4 Current Reading	NA	1	2
2D	45	Return Switch 5 Current Reading	NA	1	2
2E	46	Return Switch 6 Current Reading	NA	1	2
2F	47	Return Switch 7 Current Reading	NA	1	2
30	48	Return Switch 8 Current Reading	NA	1	2
31	49	Return Switch 9 Current Reading	NA	1	2
32	50	Return Switch 10 Current Reading	NA	1	2
33	51	Return Switch 11 Current Reading	NA	1	2
34	52	Return Switch 12 Current Reading	NA	1	2
35	53	Return Switch 13 Current Reading	NA	1	2
36	54	Return Switch 14 Current Reading	NA	1	2
37	55	Return Switch 15 Current Reading	NA	1	2
38	56	Return Switch 16 Current Reading	NA	1	2
39	57	Return Switch 17 Current Reading	NA	1	2
3A	58	Return Switch 18 Current Reading	NA	1	2
3B	59	Return Switch 19 Current Reading	NA	1	2
3C	60	Return Switch 20 Current Reading	NA	1	2
3D	61	Return Switch 21 Current Reading	NA	1	2
3E	62	Return Switch 22 Current Reading	NA	1	2
3F	63	Return Switch 23 Current Reading	NA	1	2
40	64	Return Switch 24 Current Reading	NA	1	2
41	65	Turn Switch 1 ON	NA	1	NA
42	66	Turn Switch 1 OFF	NA	1	NA
43	67	Turn Switch 2 ON	NA	1	NA
44	68	Turn Switch 2 OFF	NA	1	NA
45	69	Turn Switch 3 ON	NA	1	NA
46	70	Turn Switch 3 OFF	NA	1	NA
47	71	Turn Switch 4 ON	NA	1	NA
48	72	Turn Switch 4 OFF	NA	1	NA
49	73	Turn Switch 5 ON	NA	1	NA
4A	74	Turn Switch 5 OFF	NA	1	NA
4B	75	Turn Switch 6 ON	NA	1	NA
4C	76	Turn Switch 6 OFF	NA	1	NA
4D	77	Turn Switch 7 ON	NA	1	NA
4E	78	Turn Switch 7 OFF	NA	1	NA
4F	79	Turn Switch 8 ON	NA	1	NA

50	80	Turn Switch 8 OFF	NA	1	NA
51	81	Turn Switch 9 ON	NA	1	NA
52	82	Turn Switch 9 OFF	NA	1	NA
53	83	Turn Switch 10 ON	NA	1	NA
54	84	Turn Switch 10 OFF	NA	1	NA
55	85	Turn Switch 11 ON	NA	1	NA
56	86	Turn Switch 11 OFF	NA	1	NA
57	87	Turn Switch 12 ON	NA	1	NA
58	88	Turn Switch 12 OFF	NA	1	NA
59	89	Turn Switch 13 ON	NA	1	NA
5A	90	Turn Switch 13 OFF	NA	1	NA
5B	91	Turn Switch 14 ON	NA	1	NA
5C	92	Turn Switch 14 OFF	NA	1	NA
5D	93	Turn Switch 15 ON	NA	1	NA
5E	94	Turn Switch 15 OFF	NA	1	NA
5F	95	Turn Switch 16 ON	NA	1	NA
60	96	Turn Switch 16 OFF	NA	1	NA
61	97	Turn Switch 17 ON	NA	1	NA
62	98	Turn Switch 17 OFF	NA	1	NA
63	99	Turn Switch 18 ON	NA	1	NA
64	100	Turn Switch 18 OFF	NA	1	NA
65	101	Turn Switch 19 ON	NA	1	NA
66	102	Turn Switch 19 OFF	NA	1	NA
67	103	Turn Switch 20 ON	NA	1	NA
68	104	Turn Switch 20 OFF	NA	1	NA
69	105	Turn Switch 21 ON	NA	1	NA
6A	106	Turn Switch 21 OFF	NA	1	NA
6B	107	Turn Switch 22 ON	NA	1	NA
6C	108	Turn Switch 22 OFF	NA	1	NA
6D	109	Turn Switch 23 ON	NA	1	NA
6E	110	Turn Switch 23 OFF	NA	1	NA
6F	111	Turn Switch 24 ON	NA	1	NA
70	112	Turn Switch 24 OFF	NA	1	NA
71	113	Set All Switches	NA	4	NA
72	114	Return All Switches Status	NA	1	3
73	115	Setup the Initial Switch State	NA	4	NA
74	116	Switch all Switches to Initial Switch Status	NA	1	NA
DA	218	Status of Switch and Telemetry Node	NA	1	1
DB	219	Clear Watchdog Reset Status Bit	NA	1	NA
DF	223	Force a Watchdog Reset of the Switch and Telemetry Node	NA	1	NA

Table 8-1 Basic command overview

8.3 Command description

Each command is described in detail in this section. When each of these commands is issued additional registers may be affected. The additional registers affected are described. Since many of the commands are similar they will be described as one.

8.3.1 Return analogue channel 1-8 reading

Issuing any of these commands will give back a two byte value which relates to the analogue signal that has been requested. The actual value returned is a 10bit number. The number returned will need to be entered in to an equation, supplied by Clyde Space, to be correctly interpreted. These equations can be found in Appendix A. The reason for this is that signal conditioning is done on the analogue channels 1 to 8 and this needs to be accounted for when converting between the number returned and the signal. The first byte received is the most significant byte and the second byte received is the least significant byte.

8.3.2 Return analogue channel 9-40 reading

Issuing any of these commands will give back a two byte value which relates to the analogue signal that has been requested. The actual value returned is a 10bit number. This number can be converted to a voltage by the user by multiplying the value returned by (3/1024). The first byte received is the most significant byte and the second byte received is the least significant byte.

8.3.3 Return switch 1-24 current reading

When any of these commands are issued a two byte (10bit) number is returned. This number represents the current flowing through the associated switch. Equations for each switch will be provided by Clyde Space to allow this number to be converted into the current flowing. The equations can be found in Appendix A.

8.3.4 Turn switch 1-24 ON

To switch any switch on the correct ON command is sent. This command simply switches the relevant switch on.

8.3.5 Turn switch 1-24 OFF

To switch any switch off the correct OFF command is sent. This command simply switches the relevant switch off.

8.3.6 Set all switches

This command can be used to switch all the switches at once. Each switch can be set ON or OFF. To do this the command is sent followed by three bytes. The three bytes are set up with a 1 indicating the switch is to be turned ON and a 0 to switch a switch OFF. The format of the three bytes is as follows:

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Byte Number
SW24	SW23	SW22	SW21	SW20	SW19	SW18	SW17	1
SW16	SW15	SW14	SW13	SW12	SW11	SW10	SW9	2
SW8	SW7	SW6	SW5	SW4	SW3	SW2	SW1	3

To send this the correct sequence would be:

Command	Byte 1	Byte 2	Byte 3

8.3.7 Return all switch status

One command is used to return the status of all the switches. When issued three bytes are returned. The three bytes returned indicate the status of the switches, with a 1 indicating a switch is ON and a 0 indicating a switch is OFF. The three bytes are arranged as follows:

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Byte Number
SW8	SW7	SW6	SW5	SW4	SW3	SW2	SW1	1
SW16	SW15	SW14	SW13	SW12	SW11	SW10	SW9	2
SW24	SW23	SW22	SW21	SW20	SW19	SW18	SW17	3

8.3.8 Set initial switch state

On power up the switches will default to an initial state. When shipped this is set to be all switches OFF. This state can be set by the user using this command. Each switch can be set ON or OFF. To do this the command is sent followed by three bytes. The three bytes are set up with a 1 indicating the switch is to be turned ON and a 0 to switch a switch OFF. This command only changes the state of the switches on a power up, reset or on the issue of the initial switch command. The format of the three bytes is as follows:

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Byte Number
SW24	SW23	SW22	SW21	SW20	SW19	SW18	SW17	1
SW16	SW15	SW14	SW13	SW12	SW11	SW10	SW9	2
SW8	SW7	SW6	SW5	SW4	SW3	SW2	SW1	3

To send this the correct sequence would be:

Command	Byte 1	Byte 2	Byte 3

8.3.9 Switch all switches to initial switch state

This command will set all the switches to the initial values selected by the user or to the shipped default state if the user has not changed the initial switch status.

8.3.10 Status of switch and telemetry node

This command sets the switch and telemetry node up to return the status byte. The status byte is set up as shown in the table below.

Bit	Description
0	**HIGH**
1	**LOW**
2	Set **HIGH** if last command sent to the board was not recognised.
3	**HIGH**
4	Set **HIGH** if External Clock used
5	Set **HIGH** if Power Up reset Occurred
6	Set **HIGH** if Brown Out reset Occurred
7	Set **HIGH** if Watchdog reset occurred

8.3.11 Clear watchdog reset status bit

If the serial interface under goes a watchdog reset then a bit in the status register is set high to indicate this. This bit remains high until the user issues this command. This command clears this set bit.

8.3.12 Force watchdog reset of switch and telemetry node

This command allows the user to force a watchdog reset of the switch and telemetry node. A watchdog reset will result in the switch and telemetry rebooting. The initial state of the switches will depend upon the switch setup configuration stored in memory.

9 Serial interfaces

There exist five serial based communication connections on the Power Distribution Module. Though each may have a different physical layer or protocol, the telecommands for each of the serial interfaces are the same. This was done to make the interface to the board a simple as possible. The I^2C addresses to be used to communicate with the board are 0x51 to 0x55 (81 to 85 in decimal). All the I^2C addresses of the board can be found in Appendix A. The I^2C addresses are 7bit addresses.

9.1 Basic operation

The serial interfaces are designed to act as data buffers that provide a bridge between an I^2C network and a serial communication based device. The basic concept of the serial interfaces is that when the user wishes to send data to a serial device, the data, up to 80 bytes, is transferred to the serial interface over the I^2C network. Once there the user has the option to double check the data by reading it back, clearing the data from the buffer or transmitting the data to the serial device. Each time the user issues a command to write data to the device from the I2C network, the internal buffer counter is set to zero, technically overwriting any data that is present in the buffer. When the buffer is sent to the serial device the data is retained in the buffer. Where it can be resent, cleared or written over. Buffer overflow Flags indicate if the user overwrites data in the buffer. This condition occurs when the user writes more than 80 bytes to the buffer. If the buffer contains 80 bytes a second flag within the status byte indicates this, however this flag is only high when 80 bytes exist and is cleared if buffer overflow occurs. When new data is written to the buffer a flag indicates that new data is present. If this data is read or sent then the flag clears.

The serial interface uses interrupt based serial communication and if the connected serial device is transmitting data the serial interface captures the data and stores it in a buffer. To check for data coming from the serial device the user should poll the status byte or check the size of the serial buffer. If data is present in the serial buffer then the user simply sends a command to indicate that they wish to read this buffer. The operation of the data buffer is different from that of the I^2C buffer. When data is received by the serial buffer it is placed in the next available location within the buffer. Three error flags exist for the serial interface, Parity error, overrun error and framing error. These flags are discussed in the next section. The same flags, buffer full, buffer overrun and new data exist for the serial buffer and operate in the same way as the I^2C flags. Serial communication is disabled during any data transfers to the I^2C network. This ensures that no data is missed.

The user has the option to adjust the serial port to allow a range of different baud rates and parity types to be used. To change these, the user simply issues the commands described in the next section. When issued the new settings are stored in EEPROM and implemented immediately. On a reset or power cycle the new settings will be used.

9.2 Basic command overview

The following table shows the commands for the serial interfaces on the board.

Command		Description	Data Range	Length (Bytes)	
Hex	Dec			Cmd	Return
0xD0	208	Clear Watchdog Reset Status Bit	NA	1	0
0xD1	209	Clear Serial Interfaces Serial Buffer	NA	1	0
0xD2	210	Clear Serial Interfaces I2C Buffer	NA	1	0
0xD3	211	XOR Checksum of Data in the I2C Buffer	NA	1	1
0xD4	212	Current Number of bytes in the I2C Buffer	NA	1	1
0xD5	213	Current Number of bytes in the Serial Buffer	NA	1	1
0xD6	214	Retrieve I2C buffer	NA	1	Up to 80
0xD8	216	Clear the Overflow indication bits	NA	1	0
0xDA	218	Return Status Information from this Serial Interface	NA	1	1
0xDB	219	Send I2C Buffer to the Serial Port	NA	1	0
0xDC	220	Retrieve Serial buffer	NA	1	Up to 80
0xDD	221	Send Data to I2C Buffer	0-FF	Up to 82	0
0xDF	223	Force a Watchdog Reset of the Serial Interface	NA	1	0
0xE0	224	Set Baud Rate for the Serial Port	0-8	2	0
0xE1	225	Return Serial Interfaces Serial Port Setup	NA	1	1
0xE2	226	Set Parity bit for the Serial Port	0-2	2	0

Table 9-1 Basic command overview

9.3 Command descriptions

Each command is described in detail in this section. When each of these commands is issued additional registers may be affected. The additional registers affected are described.

9.3.1 Clear watchdog reset status bit

If the serial interface under goes a watchdog reset then a bit in the status register is set high to indicate this. This bit remains high until the user issues this command. This command clears this set bit.

9.3.2 Clear serial interfaces serial buffer

When this command is issued the 80byte serial buffer is cleared. This results in all data bytes in the buffer becoming 0x00. Resetting the buffers counters to zero and clearing multiple flags in the status register (Overflows, New data present and buffers full).

9.3.3 Clear serial interfaces I²C buffer

When this command is issued the 80byte I^2C buffer is cleared. This results in all data bytes in the buffer becoming 0x00. Resetting the buffers counters to zero and clearing multiple flags in the status register (Overflows, New data present and buffers full).

9.3.4 XOR checksum of data in I²C buffer

The XOR checksum is designed to allow the user to ensure that the data sent to the I^2C buffer of the serial interface maintains integrity. When a write to I^2C buffer command is issued the XOR checksum is set to 0 and XOR's the incoming data bytes. If the user wishes to check that all the data is received correctly then the XOR checksum can be returned. The checksum will only show up very simple errors but can be used as an indication. The data write command and data length are not included in the XOR checksum.

9.3.5 Current number of bytes in I²C buffer

Returns the number of bytes currently stored in the I^2C buffer.

9.3.6 Current number of bytes in serial buffer

Returns the number of bytes currently stored in the Serial buffer.

9.3.7 Retrieve I²C buffer

This command sets the serial interface up to send back the current data in the I^2C Buffer.

9.3.8 Clear overflow indication bits

When an overflow of either the I^2C buffer or the serial buffer occurs then relevant overflow bits are set in the status byte. The user is required to clear these by issuing this command.

9.3.9 Return status information from this serial interface

This command sets the serial interface up to return the status byte. The status byte is set up as shown in table 2 below.

Bit	Description
0	Set **HIGH** if Serial Buffer is full. Is cleared if buffer overrun occurs.
1	Set **HIGH** if I^2C Buffer is full. Is cleared if buffer overrun occurs.
2	Set **HIGH** if last command sent to the board was not recognised.
3	Set **HIGH** if new data is present in the I^2C buffer
4	Set **HIGH** if new data is present in the Serial buffer
5	Set **HIGH** if buffer overrun occurs on the Serial Buffer
6	Set **HIGH** if buffer overrun occurs on the I^2C Buffer
7	Set **HIGH** if Watchdog reset occurred

9.3.10 Send I²C buffer to serial port

When issued the data currently in the I²C buffer is transmitted to the attached serial device. The I²C buffer is not cleared. However the new I²C data flag will be cleared, as will be the associated buffer full flag.

9.3.11 Retrieve serial buffer

If data is present in the serial buffer the user can issue this command to set the serial interface up to send the serial buffer over the I²C network. Once the data is read the buffer is cleared. The new serial data flag is cleared, as will be the associated buffer full flag.

9.3.12 Send I²C data to buffer

This command is used to send data to the serial interfaces I²C buffer. This command requires 2 command bytes, the first is the command and the second is the number of bytes being transferred. The number of bytes that can be transferred is 80. An example of a packet to be sent would be:

Cmd	Length	Data							
0xDD	0x08	0x48	0x65	0x6C	0x6C	0x6F	0x28	0x28	0x28

9.3.13 Force watchdog reset of serial interface

This command allows the user to force a watchdog reset of the serial interface. A watchdog reset will result in the serial interface rebooting, clearing all buffers and resetting the device to an initial state. The initial state will depend upon the serial port configuration stored in memory.

9.3.14 Set baud rate for serial port

The user has the option to set the baud rate for the serial port. The baud rate is selected using:

Desired Baud Rate	Data Byte
300	0
1200	1
2400	2
4800	3
9600	4
19k2	5
38k4	6
57k6	7
115k2	8

To change the baud rate of the device the user should issue a command similar to:

Cmd	Data
0xE0	0x06

This command will change the baud rate of the device to 38k6. Once this command has been issued the new baud rate is set and is also stored in memory. Upon a reset, the baud rate set by the user will become the initial baud rate.

9.3.15 Return serial interfaces serial port setup

To retrieve the current settings of the serial port for the serial interface the user should issue this command. The table below shows how the returned byte should be decoded.

Byte								Value
7	6	5	4	3	2	1	0	
x	x	x	x	0	0	0	0	300 baud
x	x	x	x	0	0	0	1	1200 baud
x	x	x	x	0	0	1	0	2400 baud
x	x	x	x	0	0	1	1	4800 baud
x	x	x	x	0	1	0	0	9600 baud
x	x	x	x	0	1	0	1	19k2 baud
x	x	x	x	0	1	1	0	38k4 baud
x	x	x	x	0	1	1	1	57k6 baud
x	x	x	x	1	0	0	0	115k2 baud
x	x	0	0	x	x	x	x	No Parity
x	x	0	1	x	x	x	x	Odd Parity
x	x	1	0	x	x	x	x	Even Parity
0	1	x	x	x	x	x	x	Parity Error
1	0	x	x	x	x	x	x	Overrun Error
1	1	x	x	x	x	x	x	Framing Error

For example if the following byte was received, 00100101, this would be decoded as Even Parity and 19k2 baud.

Whenever a parity error occurs the parity error code will be transmitted. The Overrun error indicates if data from the external serial device has overwritten data within the receive buffer of the node. This may indicate an incorrect baud rate, or a failure of the serial port on the node. If this error exists then the user should check the serial port baud rate. The framing error is present if the serial data received does not conform to the standard expected, 8-N-1 for example. This error may also indicate a baud rate mismatch. When read, this register clears the Error flags.

9.3.16 Set parity bit for serial port

This command is used to set the parity bit of the device. To set the parity the following table should be used:

Desired Parity	Data Byte
None	0
Odd	1
Even	2

To change the baud rate of the device the user should issue a command similar to:

Cmd	Data
0xE2	0x01

This command will change the parity of the device to Odd. Once this command has been issued the new parity is set and is also stored in memory. Upon a reset, the parity set by the user will become the initial parity setting.

9.4 Using the serial interface

The serial interface commands have been described and the use of them briefly described. A couple of sample data transfers are described next. These are just examples of possible transfers that can be carried out.

9.4.1 Transmission to an attached serial device

The OBC wishes to send the following data stream to a connected serial device:

]> VOLT 28, CURR 0.5

One recommended method of doing the transfer is given below:

Retrieve the status byte:	Send: 0xD0 Read: 0x02
I^2C buffer is full, clear buffer:	Send: 0xD2
Retrieve the status byte:	Send: 0xD0 Read: 0x00
Transfer the Data:	Send: dd-14-5d-3e-20-56-4f-4c-54-20-32-38-2c-20-43-55-52-52-20-30-2e-35 (Note: - designed to show individual characters, do not send)

Check Status: Send: 0xD0

Read: 0x08

Check I2C buffer Length: Send: 0xD4

Read: 0x14

Transmit to Serial Device: Send: 0xDB

Serial Port Out: 5d-3e-20-56-4f-4c-54-20-32-38-2c-20-43-55-52-52-20-30-2e-35

(Note: the first two bytes have been removed as these are interfaces board commands. Again the - shows the individual bytes)

Check Status: Send: 0xD0

Read: 0x00

9.4.2 Retrieving data from an attached serial device

This procedure assumes that data has been send from the serial device and that the data is in the serial buffer. If the serial device automatically sends data then the serial interface will store 80 bytes. More than this and a buffer overflow will occur. If the serial device requires a command to be sent to allow data to be received then the user needs to go through the process described above.

Retrieve the status byte: Send: 0xD0

Read: 0x10

How many bytes present: Send: 0xD5

Read: 0x24

36 Bytes present, Read: Send: 0xDC

Read: 5d-3e-20-56-4f-4c-54-20-32-38-2c-20-43-55-52-52-20-30-2e-35-5d-3e-20-56-4f-4c-54-20-32-38-2c-20-43-55-52-76

Retrieve the status byte: Send: 0xD0

Read: 0x00

Clear Serial Buffer: Send: 0xD1

Check Length of buffer:	Send: 0xD5
	Read: 0x00

10 Test

All PDMs are fully tested prior to shipping, and test reports are supplied. In order to verify the operation of the PDM please use the following outlined instructions.

The instructions provide procedures that should be followed and expected results. Any deviation from the expected results should be investigated. Due to the nature of this board, mainly the data present in the buffers and the way the user can setup the board, every aspect of the board cannot be covered nor every use. If the board passes these tests and you are still experiencing difficulties the problem may be related to an integration issue.

10.1 Short Check

The first stage of the procedure verifies there are no shorts on the board. The connections between the power pins and ground should be buzzed. The connections between each power pin should also be checked for continuity. These results should be logged in the test report. Using a suitable meter the following connections should be buzzed; pin locations are noted in Table 10-1:

GND	H2.29, 30, 32
3.3	H2.27, 28
5	H2.25, 26
Battery	H2.45, 46
Optional	H2.51, 52

Table 10-1 pin locations

1. GND to 3.3V
2. GND to 5V
3. GND to Battery Voltage
4. GND to Other Voltage (Optional depending on board configuration)
5. 3.3 to 5V
6. 3.3V to Battery Voltage
7. 3.3V to Optional Voltage (Optional depending on board configuration)
8. 5V to Battery Voltage
9. 5V to Optional Voltage (Optional depending on board configuration)
10. Battery Voltage to Optional Voltage (Optional depending on board configuration)

10.2 Test set-up and equipment

The rest of the tests require the following test equipment. Please note that the test equipment may vary slightly, depending on the configuration of the PDM. The PDM requires the test equipment shown in Table 10-2.

Equipment	Description
3.3V PS	3.3V voltage level at 4.5-5A
5V PS	5V voltage level at 4.5-5A
6.1-8.2V PS	battery voltage levels at 4.5-5A
Variable V PS	To test the optional switches voltage level at 1.2-1.5A
Electronic Load	To allow testing of the switches should be rated for 40W.
PC/Laptop	To allow the board to be tested
Analogue Header	10 pin header for analogue channels 1 to 8
RS422 Cable + Adapter	Required for RS422 testing (FTDI USB-RS422-WE)
RS232 Cable + Adapter	Required for RS232 testing
USB to Serial Adapter	Required for LVTTL/TTL level Serial Signals (FTDI UM232R)
I2C adapter	To enable I2C connection to the PDM
Suitable test Fixture	Required to seat the board on

Table 10-2 Test equipment

The required test software is shown in Table 10-3. For the remainder of this procedure it is required that the software be installed on the test PC.

Software	Description
Terminal Software	Any suitable terminal Software. To allow communication to and from the board.
I2C adapter software	Required for any communicating to the board

Table 10-3 Test software

The setup is shown in the figure below:

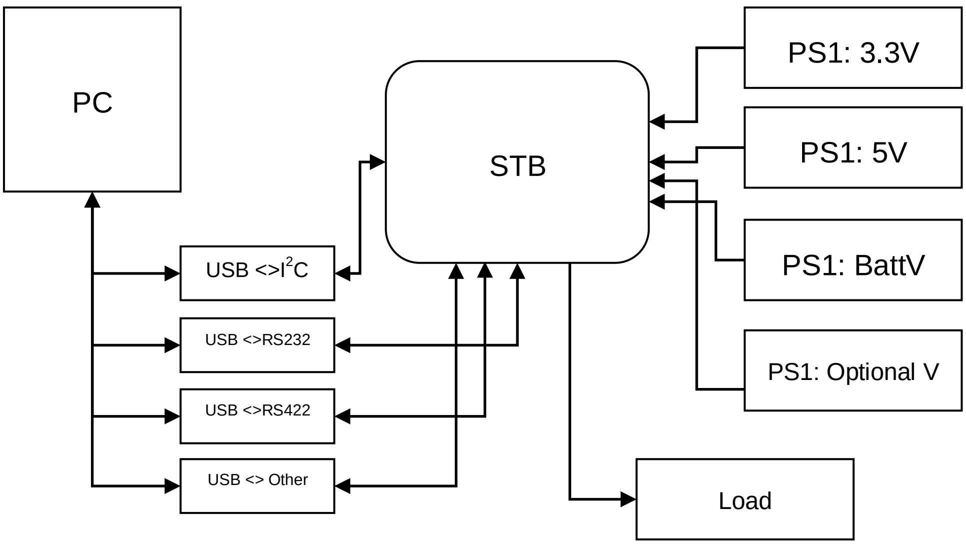

Figure 10-1 Block diagram representation of test setup

10.3 Switches

There are 24 switches on the PDM. The switches are grouped into voltage levels and each can have a different current trip point. Table 10-4 provides the expected voltage levels and the position on the header of the switch output.

Switch Number	Switch Voltage, V	Pin Out
1	3.3	H2.1
2	3.3	H2.2
3	3.3	H2.3
4	3.3	H2.4
5	3.3	H2.5
6	3.3	H2.6
7	3.3	H2.7, 8, 9
8	5	H2.10
9	5	H2.11
10	5	H2.12
11	5	H2.13
12	5	H2.14
13	5	H2.15
14	5	H2.16, 17, 18
15	Battery	H2.19
16	Battery	H2.20
17	Battery	H2.21
18	Battery	H2.22
19	Battery	H1.1, 2, 3
20	5 or Optional	H2.23
21	5 or Optional	H2.24
22	Battery or Optional	H1.4
23	Battery or Optional	H1.5
24	3.3 or Optional	H1.6

Table 10-4 Expected voltage levels

The switches can be commanded ON and OFF and return current and status telemetry. The testing is designed to verify the features of each switch as well as a number of commands that are designed to affect all the switches. The features to be tested are covered in Table 10-5.

Command/Telemetry	Operation	Affects/Concerns
ON Command	Turns switch On	Individual switch
OFF Command	Turns switch Off	Individual switch
Switch Current	Reads Current following through the Switches	Individual switch
Set All	Sets all the switches	All Switches
Reset to Initial Switch State	Returns Switches to initial	All Switches

	State	
Switch Initial Set	Sets Initial Switch State	All Switches

Table 10-5 Features for test

The next parts of the board to be tested are the switches. To carry out this test the following should be at hand:

- Assembled test setup

  Ensure the power supply current limits are set to a suitable limit for the switch testing; a limit of ~25% above the trip point of the switch would be acceptable. It may be necessary to use a parallel power supply to carry out some of the switch testing. No power supply should have a limit of more than 4.2A set on it, as this is the expected maximum trip point of any EPS or power supply attached to the system.
- Suitable test software
- Board Specification indicating the switch voltage and the Trip current. This is included in Appendix A.

Once it is confirmed that the board has been setup in the required manner. All voltage levels should be applied to the board. There is no sequence that this should be done but it may be desirable to apply the voltages as follows: Battery V, Optional V, 5V, 3.3V. As the voltages are applied the currents should be monitored to ensure that no issues are seen at this stage. Ensure that I^2C communication with the board is available. Without I^2C communication the testing cannot occur.

10.3.1 Switch operation

The first stage to be tested is the operation of each switch. To carry out this test the output on each switch should be checked to ensure that it is at the correct voltage. No load should be connected during these tests.

For each Switch the test procedure is as follows:

1. Apply positive end of Voltmeter to the corresponding switch output on the header. Ensure negative end is connected to system ground.
2. Read the voltage on the switch. This should be ~0V. (this is assuming that the board is still in its initial shipping state)
3. Issue the relevant ON Command
4. Check the Voltage reading. This should match the expected voltage.
5. Check the Status of the switch by issuing the switch status command. Ensure the status is for the desired switch is ON. Ensure the other switches have not changed status.
6. Issue the relevant OFF Command.
7. Check the Voltage reading. This should be ~0V.
8. Check the Status of the switch by issuing the switch status command. Ensure the status is for the desired switch is OFF. Ensure the other switches have not changed status.
9. Repeat for each switch

10.3.2 Set all switches

A one command approach to turning on or off all the switches is available. To test this, the following procedure should be used:

1. Check that all switches are OFF; use the switch status command to do this.
2. Issue the *set all switches* command with the data byte series 0xAA 0xAA 0xAA.
3. Check the Status of the switches by issuing the switch status command. Ensure that every even switch is ON.
4. Turn OFF all switches by issuing the set all switches command with the data byte series 0x00 0x00 0x00.
5. Check the Status of the switch by issuing the switch status command. Ensure that every switch is OFF.

10.3.3 Initial switch state

Configuration of the initial state of the switches and the reset to initial state feature are verified using the following procedure:

1. Check that all switches are OFF; use the switch status command to do this.
2. Use the Initial Setup command and construct the data bytes as follows:
    - All switches ON, except the switches 7, 14, 19 and 21.
3. Send this command
4. Power down the board and turn off the software
5. Wait one minute
6. Power the board again and turn on the software
7. Check the Status of the Switches, either by checking the voltage levels at the switch outputs or by retrieving the switch status registers.
8. Ensure that the status of the switches matches the sequence entered above.
9. Use the Initial Setup command and construct the data bytes as follows:
    - All switches OFF
10. Issue the command Switch all Switches to Initial Switch Status
11. Check the Status of the Switches, either by checking the voltage levels at the switch outputs or by retrieving the switch status registers.
12. Ensure that the status of the switches matches the sequence entered above.

10.3.4 Switch current and trip setting test

The next test is to test the current for each switch and establish the current trip point for each switch. The correct current trip points for the board being tested will be included in the board specification and should be at hand.

The test procedure is:

1. Ensure all switches are OFF; use the switch status command to do this.
2. Connect a load to the output of the switch under test. Ensure the load is at zero and is set for constant current.
3. Turn the switch on by sending the relevant switch on command
4. Retrieve the switches current reading. When entered into the equation this should show zero, or near zero depending on inaccuracies.
5. Increase the load by steps of a 5^{th} of the rated current trip. Note each value. When converted the number calculated should match the current set.
6. Repeat Step 5 until just below the current trip point of the switch. This should be the final current reading to be noted.
7. Increase the current until the switch turns OFF. Note the current and compare this to the expected trip point. Also check the state of the Switches and ensure that the tripped switch is OFF.
8. Decrease the load to zero
9. Check the status of the Switch, it should still be off. Turn on the switch and ensure the switch is on.
10. Turn the switch OFF

Carry out this procedure for all Switches.

10.3.5 Analogue channel testing

The next stage in the testing of the board is to ensure that the analogue channels work as expected.

There are 40 analogue channels on the board. They are to be tested in two groups. The first 8 analogue channels and the other 32 channels. An additional power supply is required for this test.

10.3.5.1 Analogue channels 1-8 test

The test on these channels may involve different voltage levels. The specification of the board being test should be consulted to ensure that the minimum and maximum voltage levels are correct.

To test the analogue channels:

1. With the power supply acting as the analogue input source OFF, connect to the correct analogue input pin.
2. Set the voltage to the required voltage, start at 0.25V over the minimum voltage, log this and turn the power supply ON
3. Retrieve the data using the correct command.
4. Repeat stages 2 to 4 for 5 steps up to the maximum input voltages
5. Repeat this sequence for the analogue input channels 2-8.

The data logged should be entered into the supplied equations and compared to the expected result, which is the input voltage.

10.3.5.2 Analogue channel 9-40 test

The test for the remaining analogue channels follows. This test only requires a 0-3V input voltage.

To test the analogue channels:

1. With the power supply acting as the analogue input source OFF, connect to the correct analogue input pin.
2. Set the voltage to the required voltage, start at 0.25V, log this and turn the power supply ON
3. Retrieve the data using the correct command.
4. Repeat stages 2 to 4 for 5 steps (1V, 1.5V, 2V, 2.5V, 2.9V)
5. Repeat this sequence for the other analogue input channels

The data logged should be converted by multiplying it by 3/1024. Then it can be compared to the expected result.

10.3.6 Other PDU related tests

This procedure has covered the main functions of the PDU. The next stage is to test the status of the PDU, this includes testing the watchdog.

10.3.6.1 Watchdog timer

To test the watchdog timer the following steps are required:

1. Connect up a switch with a load on it and turn the switch on. To do this Section 6.3 should be followed up to Step 5.
2. Send the Reset Command to the board
3. The switch should turn off and remain off
4. Retrieve the status byte and ensure that a watchdog reset has taken place
5. Clear the watchdog reset by sending the clear watchdog bit command
6. Retrieve the status byte and ensure that the watchdog reset has been cleared

10.3.6.2 Test bad command

To issue a bad Command:

1. Send any number in the ranges 117-217, 220-222 and 224-255 on its own
2. Retrieve the status byte and ensure the bad command bit is set
3. Retrieve the status byte and ensure the bad command bit is clear. This bit is automatically cleared when the status byte is read.

10.4 Serial port testing

This section will cover the testing of the serial devices on the PDM. The serial devices are configurable. There are hardware options that require to be set at time of build and software options that are to be tested. The serial devices specification is included as part of the board under test specification. This should be used to choose which tests are to be carried out on the board. If the RS232 option is used on the board then Section 10.4.1 of the test procedure should be followed. If the RS422 option has been selected then Section 10.4.2 should be followed. For 3.3V and 5V serial devices tests Section 10.4.3 should be followed. The test begins at Section 10.4.5.

The first stage in the testing should be to check that each device responds to a command.

The initial test procedure is:

1. With NO serial communication cables attached the user should power up the board
2. For each of the serial interfaces retrieve and note the status and serial registers
3. Turn off the unit

10.4.1 RS232 set-up

DOUBLE CHECK THE SPECIFICATION TO ENSURE THAT RS232 HAS BEEN SELECTED. Applying RS232 voltage levels to a board that has not been designed for it, will damage the board.

To test the RS232 communication a PC with a serial port is required.

Attach a RS232 header to the PDM using the Table 10-6.

Function	Board Pin out
RS232 Tx (PDM to PC)	H1.36
RS232 Rx (PC to PDM)	H1.34
GND	H2.29, 30, 32

Table 10-6 RS232 connection pins

Connect the PDM to the PC.

The PDM RS232 link is now ready for testing. Follow section 10.4.5 to test the device.

10.4.2 RS422 set-up

DOUBLE CHECK THE SPECIFICATION TO ENSURE THAT RS422 HAS BEEN SELECTED. Applying RS422 voltage levels to a board that has not been designed for it, will damage the board.

To test the RS422 communication a PC and a suitable rs422 interface to the PC are required.

Attach the RS422 header to the PDM using the Table 10.2. H5 is shown on figure 10-1. This connection is a solder or clip connection.

Function	Board Pin out
TxD+	H2.47
TxD-	H2.49
RxD+	H2.48

RxD-	H2.50
GND	H5

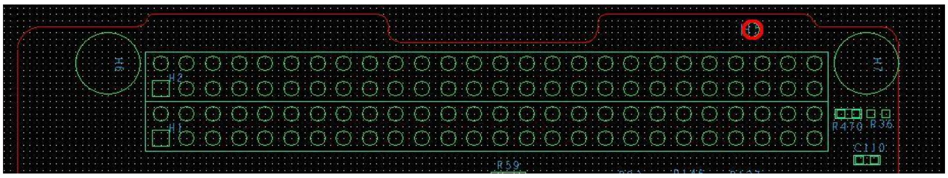

Figure 10-1 Red circle shows H5 connection

Connect the PDM to the PC.

The PDM RS422 link is now ready for testing. Follow section 10.4.5 to test the device.

10.4.3 3V3 and 5V set-up

The 3.3V and 5V serial connections require suitable connections to the PC. The possibility is that only one of these devices is available so care must be taken when swapping between 3.3V and 5V lines if both need to be tested on the board.

Table 10-7 shows the serial connections for the 3.3V and 5V connections. Tx indicates transmission from the board and Rx indicates transmissions to the board. The voltage level only indicates if the device can do 3.3V, 5V or both, not RS232 or RS422.

Serial Device	Tx Pin	Rx Pin	Voltage Level, V
RS232	H1.36	H1.34	3.3
1	H1.40	H1.38	3.3 or 5
2	H1.44	H1.42	3.3 or 5
3	H1.48	H1.46	3.3 or 5
RS422	H1.52	H1.50	3.3 or 5

Table 10-7 Serial connections for 3V3 and 5V

10.4.4 Serial device testing

The serial devices on the board all have a common software interface. The difference between each interface is the physical layer. The same procedure for each device can be followed. The only aspect that changes is the physical connection to the board.

10.4.4.1 Physical connection

Ensure that the serial interface to be tested is currently setup. This will involve connecting cables to the correct location on the board and ensuring that the connection is the correct; RS232, RS422 or 3.3V or 5V serial.

10.4.4.2 Ensure hardware is working

Once the connection has been checked the interface can be tested. To do this the following should be done:

1. Power up the PDM

2. Ensure no anomalous behavior is seen
3. Run the I^2C interface software
4. Run the terminal software selected. Care must be taken when setting up the terminal software. The standard setting is 9600-8-N-1, this assumes the user has not change any of the settings.
5. Send 10 Bytes of data, &ABCDEFGHI, to the board via I^2C
6. Retrieve the size of the I^2C buffer. This should return 10.
7. Retrieve status. This should read 00001000
8. Send the buffer to the serial port.
9. On the serial terminal screen the following should appear &ABCDEFGHI or 40 41 42 43 44 45 46 47 48 49 depending if HEX or ASCII is being displayed.
10. Using the terminal software either manually send 0 1 2 3 4 5 6 7 8 9 or use a text file with a send file command
11. Retrieve the status. Note on test report. This should read 00010000
12. Retrieve the size of the Serial buffer. This should return 10.
13. Retrieve the Serial buffer. The following should appear: 30 31 32 33 34 35 36 37 38 39.

10.4.4.3 Additional data transfer tests

The next stage is to ensure that full buffers can be sent and received. To do this:

1. Power up the PDM
2. Ensure no anomalous behavior is seen
3. Run the I^2C interface software
4. Run the terminal software selected. Care must be taken when setting up the terminal software. The standard setting is 9600-8-N-1, this assumes the user has not change any of the settings.
5. Send 80 Bytes of data, &ABCDEFGHI eight times would give the required data, to the board via I^2C
6. Retrieve the size of the I^2C buffer. This should return 80.
7. Retrieve status. This should read 00001010
8. Send the buffer to the serial port.
9. On the serial terminal screen the following should appear &ABCDEFGHI or 40 41 42 43 44 45 46 47 48 49 eight times depending if HEX or ASCII is being displayed. If this does not happen the unit has failed
10. Using the terminal software either manually send 0 1 2 3 4 5 6 7 8 9 eight times or use a text file with a send file command
11. Retrieve the status. Note on test report. This should read 00010001
12. Retrieve the size of the Serial buffer. This should return 80.
13. Retrieve the Serial buffer. The following should appear: 30 31 32 33 34 35 36 37 38 39 eight times.

10.4.4.4 Overflow test

To test the overflow:

1. Power up the PDM
2. Ensure no anomalous behavior is seen
3. Run the I^2C interface software
4. Run the terminal software selected. Care must be taken when setting up the terminal software. The standard setting is 9600-8-N-1, this assumes the user has not change any of the settings.
5. Send 80 Bytes of data, *&ABCDEFGHI* eight times with an additional *a* would give the required data, to the board via I^2C
6. Retrieve the size of the I^2C buffer. This should return 01.
7. Retrieve status. This should read 01001000
8. Clear the overflow bits and the I^2C Buffer
9. Using the terminal software either manually send *0 1 2 3 4 5 6 7 8 9* eight times with an additional *a* or use a text file contain the same data with a send file command
10. Retrieve the status. Note on test report. This should read 00110000
11. Retrieve the size of the Serial buffer. This should return 1.

10.4.4.5 Return I²C buffer test

To test this, the following should be done:

1. Power up the PDM
2. Ensure no anomalous behavior is seen
3. Run the I^2C interface software
4. Run the terminal software selected. Care must be taken when setting up the terminal software. The standard setting is 9600-8-N-1, this assumes the user has not change any of the settings.
5. Send 10 Bytes of data, &ABCDEFGHI, to the board via I^2C
6. Retrieve the size of the I^2C buffer. This should return 10.
7. Retrieve status. This should read 00001000
8. Send the return I^2C buffer command, &ABCDEFGHI should be returned.

10.4.4.6 Serial baud test

This test is designed to test the serial port baud rate. This test assumes that the starting configuration of the board is 9600-8-N-1.

1. Power up the PDM
2. Ensure no anomalous behavior is seen
3. Run the I^2C interface software
4. Run the terminal software selected. Care must be taken when setting up the terminal software. The standard setting is 9600-8-N-1, this assumes the user has not change any of the settings.

5. From the Terminal Program send &ABCDEFGHI.
6. Retrieve this data via the I^2C network and ensure that the data is correct
7. Send the *Set Baud Rate for the serial Port* command with the data byte 0x06. This will set the serial interface port up for 38K4 baud.
8. Set the terminal program to 38K4 baud.
9. Send &ABCDEFGHI via the serial terminal program
10. Retrieve this data via the I^2C network and ensure that the data is correct
11. Send the *Set Baud Rate for the serial Port* command with the data byte 0x04. This will set the serial interface port up for 9600 baud.
12. Without resetting the terminal program send &ABCDEFGHI via the serial terminal program
13. Retrieve the *Serial Interface Serial Port Setup* byte. This should show either an Overrun error or a Framing Error. The error depends on when the Serial interface picks up the first transition on the receive line.
14. Reset the terminal program to 9600 Baud
15. Clear the serial Buffer on the I^2C interface
16. send &ABCDEFGHI via the serial terminal program
17. Retrieve the data from via the I^2C network and ensure no errors exist.

10.4.4.7 Parity test

This test is designed to test the serial interface parity bit. This test assumes that the starting configuration of the board is 9600-8-N-1.

1. Power up the PDM
2. Ensure no anomalous behavior is seen
3. Run the I^2C interface software
4. Run the terminal software selected. Care must be taken when setting up the terminal software. The standard setting is 9600-8-N-1, this assumes the user has not changed any of the settings.
5. From the Terminal Program send &ABCDEFGHI.
6. Retrieve this data via the I^2C network and ensure that the data is correct
7. Send the *Set parity bit for the serial port* command with the data byte 0x01. This will set the serial interface port up for odd parity.
8. Set the terminal program to odd baud.
9. Send &ABCDEFGHI via the serial terminal program
10. Retrieve this data via the I^2C network and ensure that the data is correct
11. Send the *Set Baud Rate for the serial Port* command with the data byte 0x02. This will set the serial interface port up for even parity.
12. Without resetting the terminal program send &ABCDEFGHI via the serial terminal program
13. Retrieve the *Serial Interface Serial Port Setup* byte. This should show a parity error.

14. Reset the terminal program to even Baud
15. Clear the serial Buffer on the I^2C interface
16. send &ABCDEFGHI via the serial terminal program
17. Retrieve the data from via the I^2C network and ensure no errors exist
18. Set the serial port back to the original configuration

10.4.4.8 Watchdog reset test

To test the watchdog timer the following steps are required:

1. Power up the board
2. Send the Reset Command to the desired serial interface
3. Retrieve the status byte and ensure that a watchdog reset has taken place. Check the correct status bit.
4. Clear the watchdog reset by sending the clear watchdog bit command
5. Retrieve the status byte and ensure that the watchdog reset has been cleared
6. Repeat steps 2 to 5 for each of the serial interfaces

10.4.4.9 Unknown command test

To issue a bad Command:

1. Send any number in the ranges 117-217, 220-222 and 224-255 on its own
2. Retrieve the status byte and ensure the bad command bit is set
3. Retrieve the status byte and ensure the bad command bit is clear

11 Developer AIT

AIT of the PDM with other CubeSat modules or subsystems is the responsibility of the CubeSat developer. Whilst Clyde Space outlines a generic process which could be applicable to your particular system in this section, we are not able to offer more specific advice unless integration is between other Clyde Space products (or those of compatible products), see Table 12-1. AIT is at the risk of the developer and particular care must be taken that all subsystems are cross-compatible.

Throughout the AIT process it is recommended that comprehensive records of all actions be maintained, tracking each subsystem specifically. Photo or video detailing of any procedure also helps to document this process. Comprehensive records are useful to both the developer and Clyde Space; in the event of any anomalies complete and rapid resolution will only be possible if good records are kept. The record should contain at least;

- Subsystem and activity
- Dates and times of activity (start, finish, key milestones)
- Operator(s) and QAs
- Calibration of any equipment
- Other subsystems involved
- Method followed
- Success condition or results
- Any anomalous behaviour

Before integration each module or element should undergo an acceptance or pre-integration review to ensure that the developer is satisfied that the subsystem meets its specification through analysis, inspection, review, testing, or otherwise. Activities might include:

- Satisfactory inspection and functional test of the subsystem
- Review of all supporting documentation
- Review of all AIT procedural plans, identifying equipment and personnel needs and outlining clear pass/fail criteria
- Dry runs of the procedures in the plan

Obviously testing and analysis is not possible for all aspects of a subsystem specification, and Clyde Space is able to provide data on operations which have been performed on the system, as detailed in Table 11-1.

	Performed on	Availability
Functional	Module supplied	Provided with module
Calibration	Module supplied	Provided with module
Vacuum	Not performed	Not available
Thermal	Not performed	Not available
Simulation & modelling	Not performed	Not available

Table 11-1 Acceptance test data

Following this review, it is recommended the system undergoes further testing for verification against the developer's own requirements. An example compliance matrix structure is shown in Table 11-2.

ID	Requirement	Procedure	Result (X)	Success criteria	Compliance (pass / fail)
SYS-0030	The system mass shall be no more than 1 kg	TEST-01	0.957 kg	X < 1 kg	PASS
SYS-0040	The error LED remains off at initialisation	TEST-02	LED flashing	LED off	FAIL
SYS-0050	…	…	…	…	…

Table 11-2 Compliance matrix example

All procedural plans carried out on the PDM should conform to the test setups and procedures covered in Section 10.

During testing it is recommended that a buddy system is employed where one individual acts as the quality assurance manager and one or more perform the actions, working from a documented and reviewed test procedure. The operator(s) should clearly announce each action and wait for confirmation from their QA. This simple practice provides a useful first check and helps to eliminate common errors or mistakes which could catastrophically damage the subsystem.

Verification is project dependant, but should typically start with lower-level subsystem-specific requirements which can be verified before subsystems are integrated; in particular attention should be paid to the subsystem interfaces to ensure cross-compatibility. Verification should work upwards towards confirming top-level requirements as the system integration continues. This could be achieved by selecting a base subsystem (such as the EPS, OBC or payload) and progressively integrating modules into a stack before structural integration. Dependent upon the specific systems and qualification requirements further system-level tests can be undertaken.

When a subsystem or system is not being operated upon it should be stowed in a suitable container, as per Section 5.

12 Compatible systems

	Compatibility	Notes
Stacking Connector	CubeSat Kit Bus	CubeSat Kit definition pin compatible
	Non-standard Wire Connector	User defined
	Other Connectors	Please contact Clyde Space
EPS	Clyde Space EPS Systems	1U, 3U and DEPS variants
Batteries	Clyde Space 3U Battery Systems	10W/hr – 30 W/hr Lithium Ion Polymer
	Lithium Polymer 8.2v	(2s1p) to (2s3p)(1) More strings can be connected in parallel to increase capacity if required
	Lithium Ion 8.2v	(2s1p) to (2s3p) (1) More strings can be connected in parallel to increase capacity if required
	Other Batteries	Please contact Clyde Space
Solar Arrays	Clyde Space 3W solar array	Connects to BCRs 5&6 via SA5&6
	Clyde Space 8W solar array	Connects to BCR 1-4 via SA1-4
	3W triple junction cell arrays	2 in series connection
	8W triple junction cell arrays	6-8 in series connection
	Other array technologies	Any that conform to the input ratings for Voltage and Current
Structure	Pumpkin	CubeSat 3U structure
	ISIS	CubeSat 3U compatible
	Other structures	Please contact Clyde Space

Table 12-1 Compatible systems

Appendix A

This appendix describes the specification and configuration of the associated board. This is done to ensure that any requested modifications or alterations to the board are captured and documented correctly. Any information in this appendix supersedes that of the main user manual.

Board Serial Number: CSXXXXXX

A.1 Board Specification

Switches:	24
Optional Voltage Level:	12V (Routed to board)
Serial Devices:	3 x 3.3V
	1 x RS232
	1 x RS422
Analogue Channels Voltage Levels	
1:	3.3V
2:	3.3V
3:	3.3V
4:	3.3V
5:	3.3V
6:	3.3V
7:	3.3V
8:	3.3V
Shipped Initial Switch Status:	ALL OFF
Shipped Serial Setup:	ALL 9600-8-N-1

A.2 Board I2C Addresses

Device	Address
Switches and Telemetry	0x50
3.3V Serial Device 1	0x51
3.3V Serial Device 2	0x52
3.3V Serial Device 3	0x53
RS232	0x54
RS422	0x55

A.3 Switch Trip Points

Switch Number	Voltage Level (V)	Recommended Current Trip (A)	Actual Current Trip (A)
1	3.3	0.5	TBC
2	3.3	0.5	TBC
3	3.3	0.5	TBC
4	3.3	1	TBC
5	3.3	1	TBC
6	3.3	1	TBC
7	3.3	4*	TBC
8	5	0.5	TBC
9	5	0.5	TBC
10	5	0.5	TBC
11	5	1	TBC
12	5	1	TBC
13	5	1	TBC
14	5	4*	TBC
15	Battery	0.5	TBC
16	Battery	0.5	TBC
17	Battery	1	TBC
18	Battery	1	TBC
19	Battery	4*	TBC
20	12	0.25	TBC
21	12	0.5	TBC
22	12	0.5	TBC
23	12	1	TBC
24	12	1	TBC

A.4 Switch Current Telemetry Equations

Switch Number	Equation
1	TBD
2	TBD
3	TBD
4	TBD
5	TBD
6	TBD
7	TBD
8	TBD
9	TBD
10	TBD
11	TBD

12	TBD
13	TBD
14	TBD
15	TBD
16	TBD
17	TBD
18	TBD
19	TBD
20	TBD
21	TBD
22	TBD
23	TBD
24	TBD

A.5 Analogue Channels 1 to 8 Equations

Analogue Channels	Equation
1	TBD
2	TBD
3	TBD
4	TBD
5	TBD
6	TBD
7	TBD
8	TBD