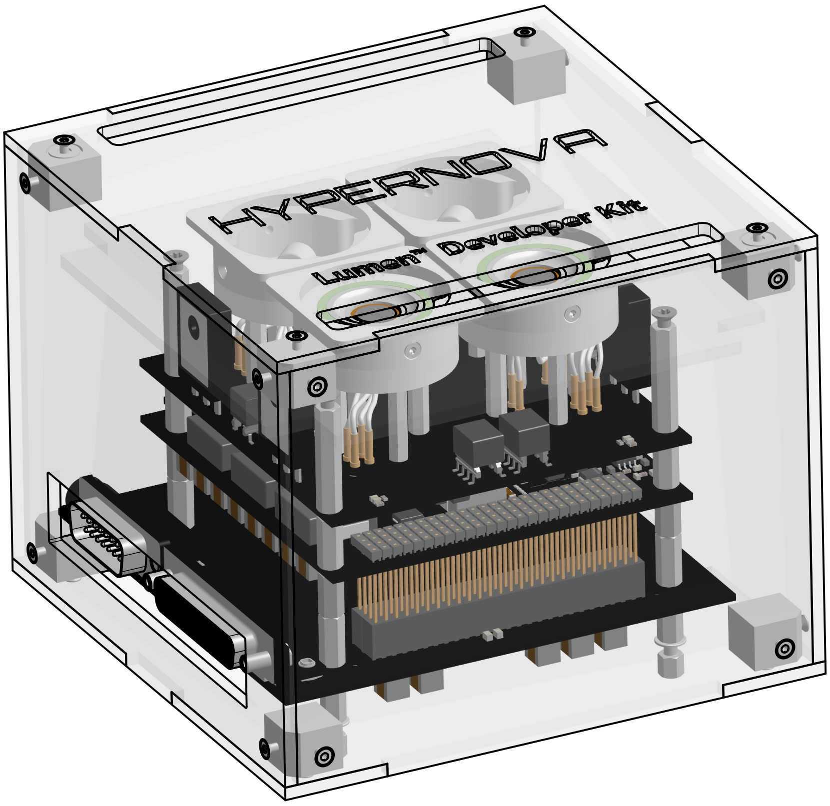
HYPERNOVA

SPACE TECHNOLOGIES



Lumen™ Developer Kit [V1.0]

User Manual

Suite 2i, Arun Place, Somerset West, 7130, Western Cape, South Africa
https://www.hypernovaspace.com/

Document Version 0.1

Author(s): Jonathan Lun & Daniel Behrens

Controlled document note: It is the user's responsibility to verify that the version of any printed documentation matches the configuration-controlled version.

Document Information

I. Reviewer(s)

Number	Date	Reviewed By
1	2023-07-27	Jonathan Lun
2		
3		

II. Document History

Revision	Date	Change details	Sections
0.1	2023-07-27	First Draft	All

III. Acronyms, Abbreviations and Symbols

The ECSS glossary of terms (ECSS-S-ST-00-01) is used as the normative reference for this document. The table below intends to add to or highlight the terms described in ECSS S-ST-00-01.

Term	Description
ACK	Acknowledge
ADC	Analogue to Digital Converter
COTS	Custom Off The Shelf
CRC	Cyclic Redundancy Check
CU	Control Unit
DB	Distribution Board
FIFO	First-In First-Out
FPGA	Field Programmable Gate Array
FW	Firmware
GSE	Ground Support Equipment
HW	Hardware
IGBT	Insulated-Gate Bipolar Transistor
NAK	Negative Acknowledgment
PPU	Power Processing Unit
PWM	Pulse Width Modulation
SRAM	Static Random-Access Memory
SW	Software
TBD	To Be Defined
TC	Telecommand
TM	Telemetry
TU	Thruster Unit
UART	Universal Asynchronous Receiver-Transmitter
UTC	Coordinated Universal Time

Table of contents

List of Figures

List of Tables

1 INTRODUCTION

The Lumen™ Developer Kit [Version 1.0] (otherwise referred to as "the Kit") is a modular benchtop system for delivering pulsed plasma jets in a controlled and precise manner. The Kit layout described in this document is for the standard configuration containing a total of four (4) thruster modules located on top of the Kit PCB boards.

One pair of thruster modules is configured in *"Dummy Mode"*, where the pulses are discharged through a surge arrestor. The discharge lights up the arrestor for visual aid. *Dummy Mode* is most suitable for non-destructive & long-term test firings.

The second pair of thruster modules is configured in *"Hot Mode"*, where the pulses are discharged through air-gapped thrusters. The discharge creates an exposed plasma arc into the ambient air, which is loud and bright. *Hot Mode* is the closest representation of plasma jet creation in a non-vacuum environment.

This document gives a functional and operational overview of the Kit and describes details of the digital communications interface.

2 DOCUMENTS

2.1 Applicable Documents

The following documents are applicable and are referred to as [ADxx] in the text. Documents are applicable in their entirety. For unspecified issues of document, the latest signed version should be used. For specified issues, subsequent amendments to or revisions of any of these publications do not apply. However, parties to the agreement based on this document are encouraged to investigate the possibility of applying the most recent issue.

Table 2-1: Applicable Documents

Reference	Document Number	Issue/Date	Title
AD01	HST/ICD10010	0.1	Lumen Developer Kit - ICD
AD02			

2.2 Reference Documents

The following documents are referenced for supporting information only and are referred to as [RDxx] in the text. For unspecified issues of a document, the latest issue should be used. Any sections that are applicable will be referenced directly in the requirements section of this technical specification.

Table 2-2: Reference Documents

Reference	Document Number	Issue/Date	Title
RD01			
RD02			

3 SYSTEM OVERVIEW

A high-level block diagram of the Kit is shown in the following figure.

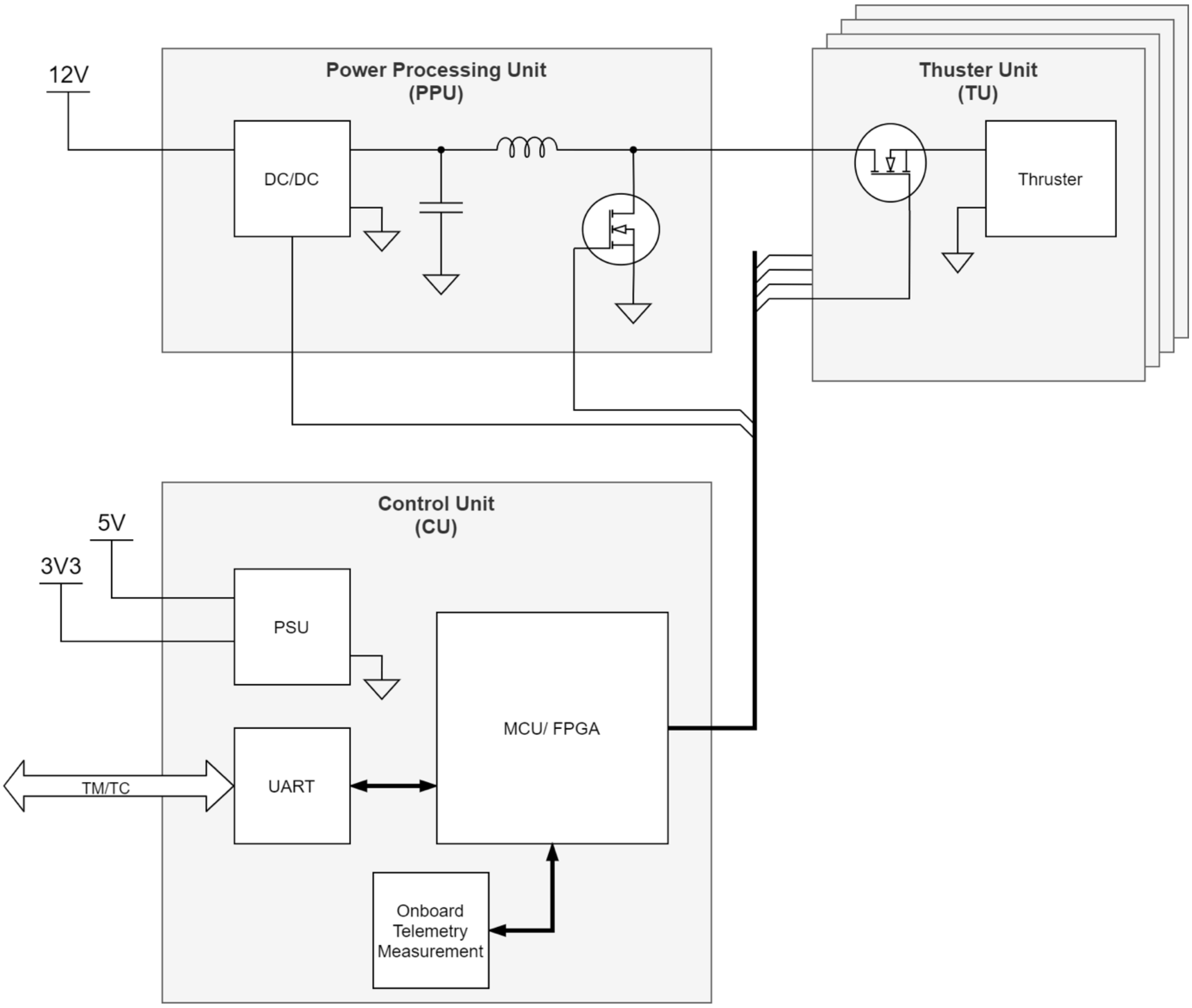


Figure 3-1: High-Level System Block Diagram

The Lumen™ Developer Kit consists of three main functional blocks, namely the Control Unit (CU), the Power Processing Unit (PPU) and the Thruster Unit (TU), which is comprised of the thrusters and Distribution Board (DB).

There are four thrusters in the Kit. To select which thruster to fire, the associated IGBT in the DB is enabled to allow current to pass from the PPU to the selected thruster. The main parts of the PPU are the DC-to-DC converter, a capacitor-inductor circuit and a trigger IGBT. The DC-to-DC converter steps the input voltage up to between approximately 20V and 80V for charging the storage capacitor. To fire a thruster, the trigger IGBT is enabled for a pre-determined length of time which starts current flowing through the inductor. When the trigger IGBT is disabled, it creates a high voltage spike at the output of the inductor that ignites an arc at the thruster. The arc extinguishes when the energy in the storage capacitor has been depleted.

The CU controls the PPU and TU. It has a digital interface for sending telecommands to the Kit and receiving telemetry from it.

3.1 Functional Overview

The functional block diagram of the CU is show below.

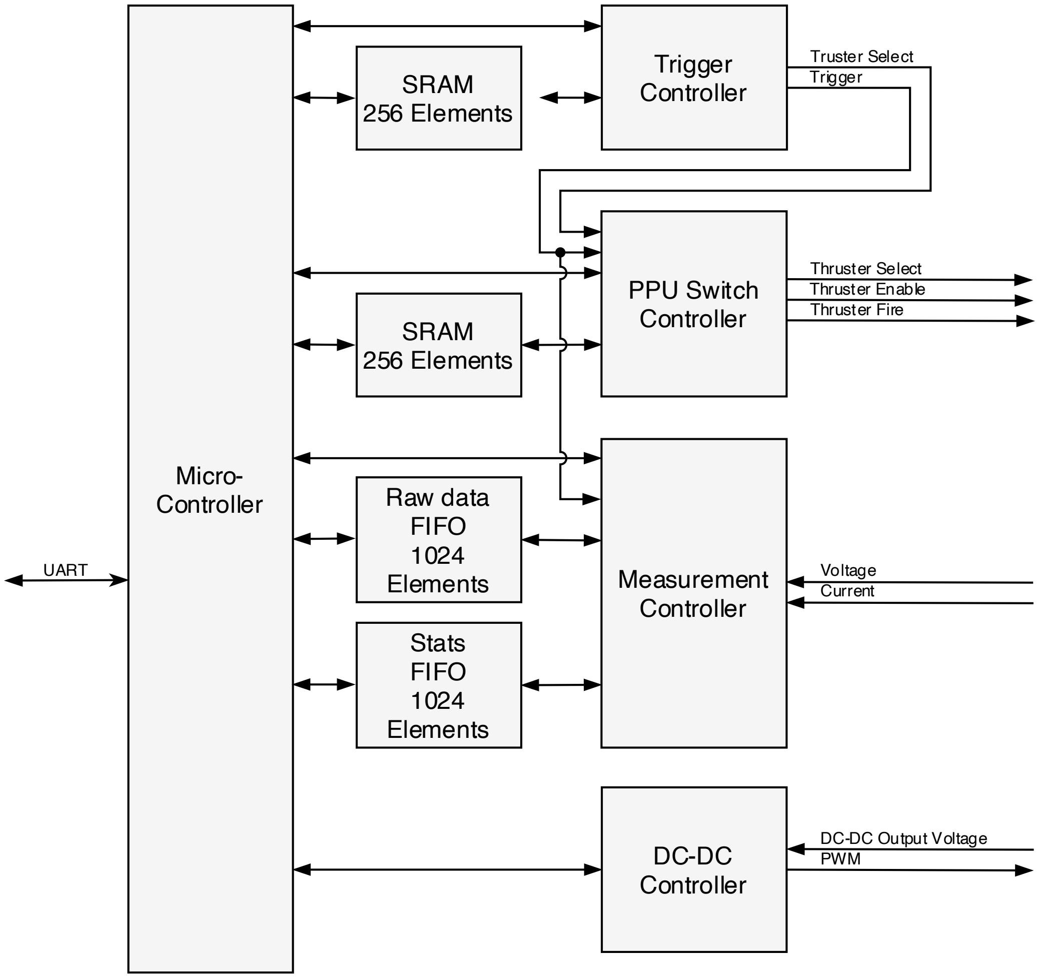


Figure 3-2: Functional Block Diagram of the CU

The Trigger Controller generates a sequence of triggers for firing of thrusters. It triggers the Switch Controller and Measurement Controller. A trigger sequence table is uploaded via the micro-controller to SRAM. Each entry in the table contains a dwell time and a thruster select value. When started, it will loop the given number of times from the given start position to the given stop position in the table.

The Switch Controller controls the timing of the switches on the PPU. A switch sequence is uploaded via the micro-controller to SRAM. Each entry in the table contains a dwell time and an on/off state for each switch. The PPU Switch Controller starts when it receives a trigger from the Trigger Controller and traverses the table from the given start position to the given stop position only once.

The Measurement Controller reads voltage and current samples from the ADCs and calculates simple statistics i.e. pulse width, max and average. The Measurement Controller also stores raw samples for the selected thruster for off-board processing. The Measurement controller is triggered by the Trigger Controller.

The DC-to-DC Controller uses a PID controller to set the output voltage of the DC-DC converter on the PPU. The setpoint voltage is set via a telecommand to the CU. The feedback to the PID controller is via an ADC that reads the output voltage of the DC-DC converter. The output of the PID controller is a PWM signal that that drives a FET.

3.2 Operation

3.2.1 Safety precautions

1. Always observe adequate electrical grounding of equipment as well as yourself when handling the Lumen.
2. Observe ESD safety practices such as an ESD wrist strap and mat where possible.
3. Operate the Lumen in a well-ventilated area. The thrusters release a high voltage, high current arc discharge, which when fired in air partially ionises and creates gases such as ozone.
4. Make sure there is at least several cm's of proximity between the thruster heads and yourself. The thrusters release a high voltage, high current arc discharge which can cause electrical shock and personal harm if physical contact is make with the discharge plasma and active thruster components.
5. Do not place foreign objects on the thruster heads which can interfere with the discharge process. The Lumen circuitry can be damaged if an unintended conducting path is created which bypasses normal system functioning and safety mechanisms.
6. Perform checks on all connections <u>before</u> powering up.
7. Perform voltage and current level checks when powering up and prior to any thruster firing.

3.2.2 Firing Sequence

A typical firing sequence is as follows:

8. Power on 3.3V, 5V and 12V
9. Read and log:
    a. Part number
    b. Serial number
    c. Version information
    d. Supply voltages and currents
    e. Temperatures
10. Abort if any of the supply voltages or currents are out of range.
11. Upload the switch control table.
12. Set switch control table start and stop pointers.
13. Upload the trigger sequence control table.
14. Set trigger sequence control table start and stop pointers and loop count.
15. Configure measurement capture settings.
16. Set DC-to-DC converter setpoint voltage.
17. Enable DC-to-DC converter.
18. Read and log DC-to-DC output voltage.
19. Abort if the DC-to-DC output voltage is out of range.

20. Send start command.
21. Wait until firing control status is not busy.
22. Read and log:
    a. Runtime
    b. Over-current flags
    c. Supply voltages and currents.
    d. Temperatures.
    e. Thruster trigger counter
    f. Thruster statistics
    g. Raw voltage and current waveform samples.
23. Power off.

4 INTERFACE DEFINITION

Communication with the Kit occurs through the UART interface (default). The Kit is a slave on the bus and responds only to telecommands or to telemetry requests. In the case of a telecommand, the Kit responds with an ACK, or a NAK if the telecommand was invalid. In the case a telemetry request, the Kit response with associated data, or a NAK if the telemetry request was invalid. The UART settings are given in the following table:

Table 4-1: UART Settings

Parameter	Setting
Baud Rate	115 200
Data Bits	8
Parity	None
Stop Bits	1

Messages, as describe in section 4.2, have a header, a data payload and end with a 16-bit CRC. The message header contains a source and destination address.
As described in section 4.1, messages are framed using Serial Line Internet Protocol (SLIP) framing.

4.1 Framing

Messages are encapsulated into framed data packets using Serial Line Internet Protocol (SLIP) framing. A special character END (0xC0) marks end of each frame. Wherever END occurs within the message, it is replaced by two bytes: ESC ESC_END (0xDB 0xDC). Wherever ESC (0xDB) occurs within the message, it is replaced by ESC ESC_ESC (0xDB 0xDD).
When processing a SLIP framed message, a ESC character can only be followed by a ESC_END or a ESC_ESC character. Any other character is incorrect.

4.2 Message Format

Messages are composed of five data fields as shown in Table 4-2. Each of the data fields will be described in detail in the paragraphs below.

Table 4-2: Message Fields

Destination Address	Source Address	Message Control	Message Data	Message CRC
(1 byte)	(1 byte)	(1 byte)	(≥0 bytes)	(2 bytes)

The length of a message is five plus the number of data bytes.

4.2.1 Destination/Source Address

The Destination Address and Source Address are eight bits long. The Destination Address identifies the receiver of the message, and the Source Address identifies the initiator of the message. Recommended address ranges are between 1 and 127. By default, the address of the Kit is 0x01, but this can be customized.

4.2.2 Message Control

The Message Control is an eight-bit character with four different bit fields as shown in Table 4-3.

Table 4-3: Message Control Byte

Bit 7 (MSB)	Bit 6	Bit 5	Bits 4 – 0
Poll bit	B bit	A bit	Command Code

There are three individually decoded bits, Poll bit, B bit, A bit and five Command Code bits. Valid command codes for the Kit are given below.

Table 4-4: Command Codes

Command Code	Command
0x04	TELEMETRY
0x05	TELECOMMAND

The setting of and interpretation of the Message Control bit fields, is defined in section 4.4 for telecommands, Section 4.5 for telemetry.

4.2.3 Message Data

The Message Data is a variable number of bytes, from 0 or more bytes. The setting of and interpretation of the Message Data, is defined in section 4.4 for telecommands, Section 4.5 for telemetry.

The data in a message is sequence of 8-bit bytes. Wherever multiple adjacent bytes are grouped together to form larger storage units (16-bit integers, 32-bit floats, etc.) the least significant bytes are transmitted first i.e. little-endian format. Floating point numbers are stored in accordance with the IEEE-754 interchange format. Below are some examples of the order in which bytes are transmitted.

Table 4-5: Byte Order Examples

Type	Value	Bytes (in order of transmission)
32-bit unsigned	1 234 567 890	0xD2, 0x02, 0x96, 0x49
32-bit float	6.67430e-11	0x2E, 0x92, 0xC4, 0xF8

4.2.4 Message CRC

Each message ends with a 2-byte Message CRC. The Message CRC is sent least significant byte first and then the most significant byte. The Message CRC is used to verify the integrity of the message. The message CRC is calculated according to CRC-16/KERMIT (CRC-16/CCITT), in which the CRC is initialised to 0x0000 and calculated using the polynomial 0x1021. Bytes are fed into the CRC computation starting with the Destination Address and concluding with the last byte of the Message Data field. An implement of the algorithm using a lookup table is provided in the appendices.

4.3 ACK and NAK Messages

Messages received by the Kit are processed by the internal state-machine of the processor. If the message includes at least a destination address field, and the destination address matches that of the Kit, it will respond with either an ACK or a NAK message. In the case that a message was correctly interpreted and processed, the Kit will respond with an ACK as presented in Table 4-6.

Table 4-6: ACK Message Format

Data Field		Description	Example
Destination Address		The address of the Kit that initiated the command.	0x11
Source Address		The address of the Kit.	0x01
Message Control	Poll bit	Set to '1' indicating a response message.	0xA5
	B bit	Not used. Set to '0'.	
	A bit	Set to '1' indicating an ACK.	
	Command Code (5 bits)	Echo of the successful command code.	
Message Data: TM/TC Address (1 byte)		Echo of the address of the successful TM/TC. See sections 4.4 and 4.5 for a list of the addresses.	0x00
Message Data: Telemetry Data (0 or more bytes)		The data payload of the telemetry response in the case of a successfully execute telemetry request. See section 4.5.	–
Message CRC		The computed CRC from the destination address up to the last byte of the message data field.	0x5481

In the case that there was an error, or that the telecommand or a telemetry request message was not correctly processed, the Kit will respond with an NAK message as presented in Table 4-7.

Table 4-7: NAK Message Format

Data Field		Description	Example
Destination Address		The address of the Kit that initiated the command. [1]	0x11
Source Address		The address of the Kit.	0x01
Message Control	Poll bit	Set to '1' indicating a response message.	0x85
	B bit	Not used. Set to '0'.	
	A bit	Set to '0' indicating an NAK.	
	Command Code (5 bits)	Echo of the unsuccessful command code. [1]	
Message Data: TM/TC Address (1 byte)		Echo of the address of the unsuccessful TM/TC. [1]	0x00
Message Data: Error Code (1 byte)		An error code indicating the type of error. 0x01 Framing Error. The length of the message is less than the minimum of 5 bytes. 0x02 CRC Error. The calculated CRC does not match that in the received message. 0x03 Invalid Command Code. 0x04 Invalid Telecommand. 0x05 Invalid Telemetry Request. 0x06 Invalid Length. The length of the message data for the given TM/TC is incorrect. 0x07 Invalid Parameter. One of the parameters in the given TM/TC request is invalid.	0x02
Message CRC		The computed CRC from the Destination Address up to the last byte of the Message Data field.	0xB5FC

[1] In the case that invalid message is sent to the Kit in which the source address, the message control byte, or the TM/TC address is missing, the echoed values in those respective fields will be undefined.

4.4 Telecommands

The format of the telecommand messages is given in Table 4-8.

Table 4-8: Telecommand Message Format

Data Field		Description	Example
Destination Address		The address of the Kit.	0x01
Source Address		The address of the Kit that initiated the command (typically the OBDH).	0x11
Message Control	Poll bit	Set to '1'.	0x85
	B bit	Not used. Set to '0'.	
	A bit	Set to '0'.	
	Command Code (5 bits)	The TELECOMMAND command code. See Table 4-4.	
Message Data: TC Address (1 byte)		See Table 4-9	0x00
Message Data: TC Parameters (0 or more bytes)		See Table 4-9	0x05 0x00
Message CRC		The computed CRC from the Destination Address up to the last byte of the Message Data field.	0x86AE

Possible telecommands are given in the following table.

Table 4-9: List of Telecommands

Address	Description	Parameters
0x00	Software Reset. Initiates a software reset of the FPGA fabric..	–
0x01	Set UTC Time. The time is Unix format.	Table 4-10
0x02	Set Trigger Source. Selects the trigger source for initiating a firing sequence. Possible sources are (i) immediate software trigger, (ii) UTC time match, (iii) external discrete trigger.	TBD
0x03	Upload Trigger Table. The trigger table contains 256 entries where each entry contains the thruster selection and a dwell time in ms.	Table 4-11
0x04	Set Trigger Table Config. Sets the start and stop pointers to entries in the trigger table, and a loop counter that indicates the number of times the Kit should repeat the trigger sequence.	Table 4-12
0x05	Upload Switch Table. The switch table contains 256 entries where each entry contains the state of the switches on the PPU, as well as the dwell time in µs.	Table 4-13
0x06	Set Switch Table Config. Sets the start and stop pointers to entries in the switch table. The Kit will execute the selected sequence each time a trigger is generated.	Table 4-14
0x07	Start Firing Sequence. Generates an immediate software trigger that starts the firing sequence configured using the Set Trigger Table Config telecommand (0x04).	–
0x08	Stop firing Sequence. Immediately halts the any firing sequence currently executing.	–
0x09	Set PPU Config. Resets over-current flags, enable/disable the DC-DC converter and sets the DC-DC converter setpoint voltage.	Table 4-16
0x0A	Set Measurement Config. Used to clear the raw and stats FIFOs, set the pulse detection threshold and the thruster filter selection for storing raw data samples.	Table 4-17

The parameters for each telecommand are presented in the remainder of this section.

Table 4-10: Set UTC Time Telecommand Parameters

Offset	Parameter	Description	Type	Bytes
0	UTC Time	Unix time in seconds i.e. the seconds that have elapsed since the Unix epoch (00:00:00 UTC on 1 January 1970), minus leap seconds	64-bit unsigned	8

Table 4-11: Upload Trigger Table Telecommand Parameters

Offset	Parameter	Description	Type	Bytes
0	Address Offset	The address in the trigger table at which to begin writing the given data. The trigger table contains 256 entries, so the offset address range is 0 to 255.	16-bit unsigned	2
2 + 4i	Dwell Time	The time in ms to wait to wait before moving to the following entry in the trigger table.	16-bit unsigned	2
4 + 4i	Thruster Select	The index of the thruster to fire, indexed from 0.	16-bit unsigned	2

Table 4-12: Set Trigger Table Config Telecommand Parameters

Offset	Parameter	Description	Type	Bytes
0	Start Pointer	The address in the trigger table at which to begin the next trigger sequence.	8-bit unsigned	1
1	Stop Pointer	The address in the trigger table at which to end the next trigger sequence.	8-bit unsigned	1
2	Loops	The number of times to repeat the trigger sequence.	16-bit unsigned	2

Table 4-13: Upload Switch Table Telecommand Parameters

Offset	Parameter	Description	Type	Bytes
0	Address Offset	The address in the switch table at which to begin writing the given data. The switch table contains 256 entries, so the offset address range is 0 to 255.	16-bit unsigned	2
2 + 4i	Dwell Time	The time in µs to wait to wait before moving to the following entry in the trigger table.	16-bit unsigned	2
4 + 4i	Switch Select	The state of the switches: b2: Over-Current Reset b1: Trigger IGBT Enable b0: Thruster Enable	16-bit unsigned	2

Table 4-14: Set Switch Table Config Telecommand Parameters

Offset	Parameter	Description	Type	Bytes
0	Start Pointer	The address in the switch table containing the initial state of the switches for each firing.	8-bit unsigned	1
1	Stop Pointer	The address in the switch table containing the final state of the switches for each firing.	8-bit unsigned	1

The following table shows the default values that should be programmed into the switch table.

Table 4-15: Default Switch Table Values

Address	Dwell [µs]	B2: Over-Current Reset	B1: Trigger IGBT Enable	B0: Thruster Enable
0	10	1	0	0
1	100	0	0	1
2	25	0	1	1
3	1000	0	0	1

Table 4-16: Set PPU Config Telecommand Parameters

Offset	Parameter	Description	Type	Bytes
0	Control Bits	Control bits: b3: 12V on b2:DC-to-DC converter enable b1:DC-to-DC converter over-current reset b0:Trigger IGBT over-current reset	8-bit unsigned	1
1	DC-DC Setpoint	The voltage setpoint value for the DC-DC converter for powering the thrusters. $.f(x) = x/150 \times 4095 \ [bits]$ Where x is the desired voltage. The default value should be 50V = 0x555	16-bit unsigned	2

Table 4-17: Set Measurement Config Telecommand Parameters

Offset	Parameter	Description	Type	Bytes
0	Control Bits	Control bits: b2: Raw FIFO clear b1: Stats FIFO clear b0: Trigger count clear	8-bit unsigned	1
1	Pulse Threshold	The threshold level above which the pulse is considered valid. $.f(x) = x/66 \times 4095 \ [bits]$ Where x is the current in amps.	16-bit unsigned	2
3	Sample Rate	Reserved. Sample rate is currently fixed at 10us in the firmware.	8-bit unsigned	1
4	Capture Select	The thruster for which to store raw voltage and current waveform samples. Indexed from 0.	8-bit unsigned	1

4.5 Telemetry

The format of the telemetry request messages is given in Table 4-18.

Table 4-18: Telemetry Request Message Format

Data Field		Description	Example
Destination Address		The address of the Kit.	0x01
Source Address		The address of the Kit that initiated the command	0x11
Message Control	Poll bit	Set to '1'.	0x84
	B bit	Not used. Set to '0'.	
	A bit	Set to '0'.	
	Command Code (5 bits)	The TELEMETRY command code. See Table 4-4.	
Message Data: TM Address (1 byte)		See Table 4-19	0x80
Message Data: TM Parameters (0 or more bytes)		See Table 4-19	–
Message CRC		The computed CRC from the Destination Address up to the last byte of the Message Data field.	0xAC56

Possible telemetry requests to the are given in the following table.

Table 4-19: List of Telemetry Requests

Address	Description	Parameters	Response
0x80	Get Part Number	–	Table 4-20
0x81	Get Serial Number	–	Table 4-21
0x82	Get Version Info	–	Table 4-22
0x83	Get Device Info	–	Table 4-23
0x84	Get Runtime	–	Table 4-24
0x85	Get UTC Time	–	
0x86	Get Onboard Telemetry	–	Table 4-26
0x87	Get Trigger Status	–	Table 4-27
0x88	Get Switch Status	–	Table 4-28
0x89	Get PPU Status	–	Table 4-29
0x90	Get Measurement Status	–	Table 4-30
0x91	Read Raw Data FIFO	Elements to read: 16-bit unsigned	Table 4-31
0x92	Read Stats FIFO	Elements to read: 16-bit unsigned	Table 4-32
0x93	Get Resettable Trigger Counters	–	Table 4-33
0x94	Get Persistent Trigger Counters	–	Table 4-34

As described in section 4.5, the Kit responds to a telemetry request with either an ACK or a NAK. In the case of an ACK, the message data contains the telemetry data payload. The remainder of this section describes the telemetry data outputted in response to a successful telemetry request.

Table 4-20 Get Part Number Telemetry Data

Offset	Parameter	Description	Type	Bytes
0	Part Number	The part number of the Kit.	String	≤ 128

Table 4-21 Get Serial Number Telemetry Data

Offset	Parameter	Description	Type	Bytes
0	Serial Number	The serial number of the Kit.	String	≤ 128

Table 4-22 Get Version Info Telemetry Data

Offset	Parameter	Description	Type	Bytes
0	HW Mod Status	The modification count of the hardware.	16-bit unsigned	2
2	HW Minor Version	The minor version (revision) of the hardware. Incremented with hardware improvements that do not affect the compatibility of software, firmware or external interfaces.	8-bit unsigned	1
3	HW Major Version	The major version of the hardware. Hardware with different major versions are not pin-for-pin compatible and therefore not interchangeable.	8-bit unsigned	1
5	SW Build	Incremented each time the software is compiled and reset when a new version is released.	16-bit unsigned	2
6	SW Minor Version	Incremented on a minor software release, including bug fixes and inclusion of additional features.	8-bit unsigned	1
7	SW Major Version	Incremented on a major software release, where the new version is incompatible with the previous version.	8-bit unsigned	1
9	FW Build	Incremented each time the firmware is compiled and reset when a new version is released.	16-bit unsigned	2
10	FW Minor Version	Incremented on a minor software release, including bug fixes and inclusion of additional features.	8-bit unsigned	1
11	FW Major Version	Incremented on a major firmware release, where the new version is incompatible with the previous version.	8-bit unsigned	1

Table 4-23 Get Device Info Telemetry Data

Offset	Parameter	Description	Type	Bytes
0	Device Serial Number	The 128-bit device serial number of the processor.	Array of 8-bit unsigned	16
4	Device User Code	The 32-bit user code set in the processor.	Array of 8-bit unsigned	4
8	Device Design Version	The design version set in the processor	Array of 8-bit unsigned	2

Table 4-24 Get Runtime Telemetry Data

Offset	Parameter	Description	Type	Bytes
0	Runtime	The runtime in seconds of the FPGA fabric.	32-bit unsigned	4

Table 4-25 Get UTC Time Telemetry Data

Offset	Parameter	Description	Type	Bytes
0	UTC Time	The Unix time in seconds i.e. the seconds that have elapsed since the Unix epoch (00:00:00 UTC on 1 January 1970), minus leap seconds	32-bit unsigned	4

Table 4-26 Get Onboard Telemetry Data

Offset	Parameter	Description	Type	Bytes
0	Channel 0	Power Unit +3V3. $f(x) = x/\,4095 \times 5\ [V]$	16-bit unsigned	2
2	Channel 1	Power Unit +5V. $f(x) = x/4095 \times 1.56 \times 5\ [V]$	16-bit unsigned	2
4	Channel 2	Power Unit +12V $f(x) = x/4095 \times 4.3 \times 5\ [V]$	16-bit unsigned	2
6	Channel 3	Power Unit BATT-RAW. $f(x) = x/4095 \times 11 \times 5\ [V]$	16-bit unsigned	2
8	Channel 4	Power Unit +3V3 Current. $f(x) = x/4095\ [A]$	16-bit unsigned	2
10	Channel 5	Power Unit +5V Current. $f(x) = x/4095\ [A]$	16-bit unsigned	2
12	Channel 6	Power Unit +12V Current. $f(x) = x/4095\ \times 8.33\ [A]$	16-bit unsigned	2
14	Channel 7	DC-DC Converter Temperature $f(x) = 1/\left(\frac{1}{298.15} + \frac{1}{3936} ln\left(\frac{1}{4095/x - 1}\right)\right) - 273.15\ [°C]$	16-bit unsigned	2
16	Channel 8	IGBT Temperature. $f(x)$ as per Channel 7.	16-bit unsigned	2
18	Channel 9	Inductor Temperature. $f(x)$ as per Channel 7.	16-bit unsigned	2
20	Channel 10	Thruster1 Temperature $f(x) = 1/\left(\frac{1}{298.15} + \frac{1}{3435} ln\left(\frac{1}{4095/x - 1}\right)\right) - 273.15\ [°C]$	16-bit unsigned	2
22	Channel 11	Thruster2 Temperature. $f(x)$ as per Channel 10.	16-bit unsigned	2
24	Channel 12	Thruster3 Temperature. $f(x)$ as per Channel 10.	16-bit unsigned	2
26	Channel 13	Thruster4 Temperature. $f(x)$ as per Channel 10.	16-bit unsigned	2
28	Channel 14	Reserved	16-bit unsigned	2
30	Channel 15	Reserved	16-bit unsigned	2
32	Channel 16	Control Unit +1V2. $f(x) = x/\,4095 \times 5\ [V]$	16-bit unsigned	2
34	Channel 17	Control Unit +3V3. $f(x) = x/\,4095 \times 5\ [V]$	16-bit unsigned	2
36	Channel 18	Control Unit +5V. $f(x) = x/4095 \times 1.56 \times 5\ [V]$	16-bit unsigned	2
38	Channel 19	Reserved	16-bit unsigned	2
40	Channel 20	Reserved	16-bit unsigned	2
42	Channel 21	Control Unit +1V2 Current. $f(x) = x/4095\ [A]$	16-bit unsigned	2
44	Channel 22	Reserved	16-bit unsigned	2
46	Channel 23	Reserved	16-bit unsigned	2
48	Channel 24	Reserved	16-bit unsigned	2
50	Channel 25	Reserved	16-bit unsigned	2
52	Channel 26	Reserved	16-bit unsigned	2
54	Channel 27	Control Unit Temperature0. $f(x)$ as per Channel 10.	16-bit unsigned	2
56	Channel 28	Control Unit Temperature1. $f(x)$ as per Channel 10.	16-bit unsigned	2
58	Channel 29	Reserved	16-bit unsigned	2
60	Channel 30	Reserved	16-bit unsigned	2
62	Channel 31	Reserved	16-bit unsigned	2

Nominal and out of range values for the onboard telemetry is given in the appendix.

Table 4-27 Get Trigger Status Telemetry Data

Offset	Parameter	Description	Type	Bytes
0	Busy	A non-zero value indicates that the Kit is busy executing a trigger sequence.	8-bit unsigned	1
1	Current Pointer	The address of the currently executing entry in the trigger table.	8-bit unsigned	1
3	Remaining Loops	The remaining number of loops in the currently executing trigger sequence.	16-bit unsigned	2

Table 4-28 Get Switch Status Telemetry Data

Offset	Parameter	Description	Type	Bytes
0	Busy	A non-zero value indicates that the switch controller is running.	8-bit unsigned	1
1	Current Pointer	The address of the currently executing entry in the switch table.	8-bit unsigned	1

Table 4-29 Get PPU Status Telemetry Data

Offset	Parameter	Description	Type	Bytes
0	Over Current	Over-current flags: b1:DC-to-DC converter over-current b0:Trigger IGBT over-current	8-bit unsigned	1
1	DC-DC Voltage	The DC-DC converter output voltage. $f(x) = x/4095 \times 150\ [V]$ Where x is the raw value in bits.	16-bit unsigned	2

Table 4-30 Get Measurement Status Telemetry Data

Offset	Parameter	Description	Type	Bytes
0	Busy	A non-zero value indicates that the measurement controller is running.	8-bit unsigned	1
1	Raw FIFO Used	The number of elements stored in the raw data FIFO.	16-bit unsigned	2
3	Stats FIFO Used	The number of elements stored in the stats data FIFO.	16-bit unsigned	2

Table 4-31 Get Raw Data FIFO Telemetry Data

Offset	Parameter	Description	Type	Bytes
0 + 5i	Waveform Index	Waveform data is sampled every 10µs and so multiple waveforms can be stored in the FIFO for the selected thruster. The waveform index is a number that increments for each new waveform and can be used to group the samples that belong to each captured waveform. After 255 it rolls over back to 0.	8-bit unsigned	1
1 + 5i	Voltage	The voltage value sample $f(x) = x/4095 \times 1500\ [V]$ Where x is the raw value in bits.	16-bit unsigned	2

3 + 5i	Current	The current value sample $f(x) = x/4095 \times 66\ [A]$ Where x is the raw value in bits.	16-bit unsigned	2

Table 4-32 Get Stats FIFO Telemetry Data

Offset	Parameter	Description	Type	Bytes
0 + 11i	Thruster Index	The statistics FIFO stores basic statistical data for each thruster firing. The thruster index indicates which thruster the statistical data the current entry belongs.	8-bit unsigned	1
1 + 11i	Sample Count	Waveform data is sampled every 10µs. The sample count indicates how many samples were used to determine the statistics in the current entry. By multiplying this value by 10µs one can closely estimate the pulse width of the waveform.	16-bit unsigned	2
3 + 11i	Peak Voltage	The peak voltage value $f(x) = x/4095 \times 1500\ [V]$ Where x is the raw value in bits.	16-bit unsigned	2
5 + 11i	Average Voltage	The average voltage value $f(x) = x/4095 \times 1500\ [V]$ Where x is the raw value in bits.	16-bit unsigned	2
8 + 11i	Peak Current	The peak current value $f(x) = x/4095 \times 66\ [A]$ Where x is the raw value in bits.	16-bit unsigned	2
10 + 11i	Average Current	The average current value $f(x) = x/4095 \times 66\ [A]$ Where x is the raw value in bits.	16-bit unsigned	2

Table 4-33 Get Resettable Trigger Counters Telemetry Data

Offset	Parameter	Description	Type	Bytes
0	Total Trigger Count	The total number of times that all the thrusters together have been fired since the last reset.	32-bit unsigned	4
4	Thruster1 Count	The number of times that thruster 1 has been fired since the last reset.	16-bit unsigned	2
6	Thruster2 Count	The number of times that thruster 2 has been fired since the last reset.	16-bit unsigned	2
8	Thruster3 Count	The number of times that thruster 3 has been fired since the last reset.	16-bit unsigned	2
10	Thruster4 Count	The number of times that thruster 4 has been fired since the last reset.	16-bit unsigned	2

Table 4-34 Get Persistent Trigger Counters Telemetry Data

Offset	Parameter	Description	Type	Bytes
0	Total Trigger Count	The total number of times that all the thrusters together have been fired in their lifetimes.	32-bit unsigned	4
4	Thruster1 Count	The number of times that thruster 1 has been fired in its lifetime.	16-bit unsigned	2
6	Thruster2 Count	The number of times that thruster 2 has been fired in its lifetime.	16-bit unsigned	2
8	Thruster3 Count	The number of times that thruster 3 has been fired in its lifetime.	16-bit unsigned	2
10	Thruster4 Count	The number of times that thruster 4 has been fired in its lifetime.	16-bit unsigned	2

A1. CRC-16/KERMIT Implementation using a Lookup Table

```
// Lookup table for CRC-16/KERMIT (0x1021)
const unsigned short crcLookup[] = {
        0x0000,0x1189,0x2312,0x329b,0x4624,0x57ad,0x6536,0x74bf,
        0x8c48,0x9dc1,0xaf5a,0xbed3,0xca6c,0xdbe5,0xe97e,0xf8f7,
        0x1081,0x0108,0x3393,0x221a,0x56a5,0x472c,0x75b7,0x643e,
        0x9cc9,0x8d40,0xbfdb,0xae52,0xdaed,0xcb64,0xf9ff,0xe876,
        0x2102,0x308b,0x0210,0x1399,0x6726,0x76af,0x4434,0x55bd,
        0xad4a,0xbcc3,0x8e58,0x9fd1,0xeb6e,0xfae7,0xc87c,0xd9f5,
        0x3183,0x200a,0x1291,0x0318,0x77a7,0x662e,0x54b5,0x453c,
        0xbdcb,0xac42,0x9ed9,0x8f50,0xfbef,0xea66,0xd8fd,0xc974,
        0x4204,0x538d,0x6116,0x709f,0x0420,0x15a9,0x2732,0x36bb,
        0xce4c,0xdfc5,0xed5e,0xfcd7,0x8868,0x99e1,0xab7a,0xbaf3,
        0x5285,0x430c,0x7197,0x601e,0x14a1,0x0528,0x37b3,0x263a,
        0xdecd,0xcf44,0xfddf,0xec56,0x98e9,0x8960,0xbbfb,0xaa72,
        0x6306,0x728f,0x4014,0x519d,0x2522,0x34ab,0x0630,0x17b9,
        0xef4e,0xfec7,0xcc5c,0xddd5,0xa96a,0xb8e3,0x8a78,0x9bf1,
        0x7387,0x620e,0x5095,0x411c,0x35a3,0x242a,0x16b1,0x0738,
        0xffcf,0xee46,0xdcdd,0xcd54,0xb9eb,0xa862,0x9af9,0x8b70,
        0x8408,0x9581,0xa71a,0xb693,0xc22c,0xd3a5,0xe13e,0xf0b7,
        0x0840,0x19c9,0x2b52,0x3adb,0x4e64,0x5fed,0x6d76,0x7cff,
        0x9489,0x8500,0xb79b,0xa612,0xd2ad,0xc324,0xf1bf,0xe036,
        0x18c1,0x0948,0x3bd3,0x2a5a,0x5ee5,0x4f6c,0x7df7,0x6c7e,
        0xa50a,0xb483,0x8618,0x9791,0xe32e,0xf2a7,0xc03c,0xd1b5,
        0x2942,0x38cb,0x0a50,0x1bd9,0x6f66,0x7eef,0x4c74,0x5dfd,
        0xb58b,0xa402,0x9699,0x8710,0xf3af,0xe226,0xd0bd,0xc134,
        0x39c3,0x284a,0x1ad1,0x0b58,0x7fe7,0x6e6e,0x5cf5,0x4d7c,
        0xc60c,0xd785,0xe51e,0xf497,0x8028,0x91a1,0xa33a,0xb2b3,
        0x4a44,0x5bcd,0x6956,0x78df,0x0c60,0x1de9,0x2f72,0x3efb,
        0xd68d,0xc704,0xf59f,0xe416,0x90a9,0x8120,0xb3bb,0xa232,
        0x5ac5,0x4b4c,0x79d7,0x685e,0x1ce1,0x0d68,0x3ff3,0x2e7a,
        0xe70e,0xf687,0xc41c,0xd595,0xa12a,0xb0a3,0x8238,0x93b1,
        0x6b46,0x7acf,0x4854,0x59dd,0x2d62,0x3ceb,0x0e70,0x1ff9,
        0xf78f,0xe606,0xd49d,0xc514,0xb1ab,0xa022,0x92b9,0x8330,
        0x7bc7,0x6a4e,0x58d5,0x495c,0x3de3,0x2c6a,0x1ef1,0x0f78};

//-------------------------------------------------------------------------------
uint16_t CalculateCrc16(const uint8_t *data, const uint16_t offset, const uint16_t count,
const uint16_t init)
{
    uint16_t res = init;
    uint16_t i = 0;

    for (i = offset; i < (offset + count); i++)
       res = crcLookup[(res ^ data[i]) & 0xFF] ^ (res >> 8);

    return res;
}
```

A2. Nominal and Out of Range Values for Onboard Telemetry

The following table defines the nominal and out of range values for the onboard telemetry.

Channel	Description	Nominal	Out of Range		Units
			Minimum	Maximum	
0	Power Unit +3V3	3.30	3.13	3.47	V
1	Power Unit +5V	5.00	4.75	5.25	V
2	Power Unit +12	12.00	11.4	12.6	V
3	Power Unit BATT-RAW.	–	–	–	V
4	Power Unit +3V3 Current.	115	90	150	mA
5	Power Unit +5V Current.	48	35	60	mA
6	Power Unit +12V Current.	20	0	4000	mA
7	DC-DC Converter Temperature.	25.0	-40	100	°C
8	IGBT Temperature.	25.0	-40	100	°C
9	Inductor Temperature.	25.0	-40	100	°C
10	Thruster1 Temperature.	25.0	-40	100	°C
11	Thruster2 Temperature.	25.0	-40	100	°C
12	Thruster3 Temperature.	25.0	-40	100	°C
13	Thruster4 Temperature.	25.0	-40	100	°C
16	Control Unit +1V2.	1.20	1.14	1.26	V
17	Control Unit +3V3.	3.30	3.13	3.47	V
18	Control Unit +5V.	5.00	4.75	5.25	V
21	Control Unit +1V2 Current.	92	70	110	mA
27	Control Unit Temperature0.	25.0	-40	100	°C
28	Control Unit Temperature1.	25.0	-40	100	°C

A3. Getting Started (with an example)

This section provides a guide for getting started with the Kit in benchtop setup with a PC.
As shown in the figure below,

1. Check that the Kit is fully plugged onto the PC104 connector on the GSE board.
2. Connect the provided power cable between the DB15 on the GSE board and a lab power supply. Set the current limits on the power supply as follows:

Power Up Sequence	Voltage Rail	Nominal Current	Current Limit
1	+5V	50mA	150mA
2	+3V3	110mA	250mA
3	+12V	10mA	> 3A

Since 12V is only used for supplying the power electronics on the Kit, it will still be possible to communicate with and configure the Kit if it is not supplied – it will just not be possible to fire the thrusters. +5V and +3V3 are required for basic operation of the Kit. As shown in the table above, +5V should be applied first, followed by +3V3, then +12V.

3. Connect the provided comms cable between the DB25 on the GSE board and the USB on a PC.

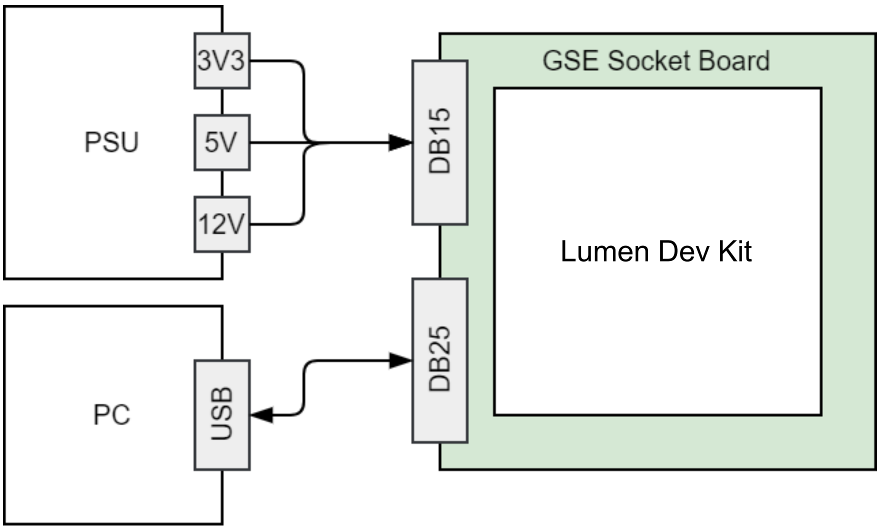


Figure A2-1: Kit Benchtop Setup

4. Switch on the power supply and observe the current on the power supply is within 10% of the nominal current given in the table above.
5. Using a terminal program such as Docklight (https://docklight.de/) configure the serial port settings as follows: Baud Rate: 115 200, Data Bits: 8, Parity: None, Stop Bits: 1
6. Read the Part Number

```
[TX] - 01 00 04 80 D3 FF C0
[RX] - 00 01 A4 80 4E 61 6E 6F 54 68 72 75 73 74 65 72 2D 41 55 97 C0
```

The Kit responds with the string "Lumen Developer Kit"

7. Set PPU Config: Enable 12V, set DC-to-DC voltage = 40V

```
[TX] - 01 00 05 09 0C 44 04 63 82 C0
[RX] - 00 01 A5 09 00 00 C0
```

The Kit responds with an ACK

8. Upload Example Trigger Table. Only the Dummy thrusters (address 0 & 1) should fire.

Address	Dwell [ms]	Thruster
0	500	0
1	500	1
~~2~~	~~500~~	~~2~~
~~3~~	~~500~~	~~3~~

```
[TX] - 01 00 05 03 00 00 F4 01 00 00 F4 01 01 00 F4 01 02 00 F4 01 03 00 C1 9A C0
[RX] - 00 01 A5 03 00 00 C0
```

The Kit responds with an ACK

9. Set Example Trigger Table Config. Start pointer = 0, stop pointer = 3, loops = 1

```
[TX] - 01 00 05 04 00 03 01 00 94 4E C0
[RX] - 00 01 A5 04 00 00 C0
```

The Kit responds with an ACK

10. Upload Switch Table

Address	Dwell [µs]	B2:Reset	B1:Trigger IGBT Enable	B0:ThrusterEnable
0	10	1	0	0
1	100	0	0	1
2	25	0	1	1
3	1000	0	0	1

```
[TX] - 01 00 05 05 00 00 0A 00 04 00 64 00 01 00 19 00 03 00 E8 03 01 00 0A 63 C0
[RX] - 00 01 A5 05 00 00 C0
```

The Kit responds with an ACK

11. Set Switch Table Config: Start pointer = 0, stop pointer = 3

```
[TX] - 01 00 05 06 00 03 3E 8E C0
[RX] - 00 01 A5 06 00 00 C0
```

The Kit responds with an ACK

12. Start Firing Sequence

```
[TX] - 01 00 05 07 BC 16 C0
[RX] - 00 01 A5 07 00 00 C0
```

The Kit responds with an ACK.

Each of the Dummy thrusters in the Kit should fire once with a 500ms delay between firings.

The following figure shows the complete sequence of bytes sent and recieved from the Kit in Docklight.

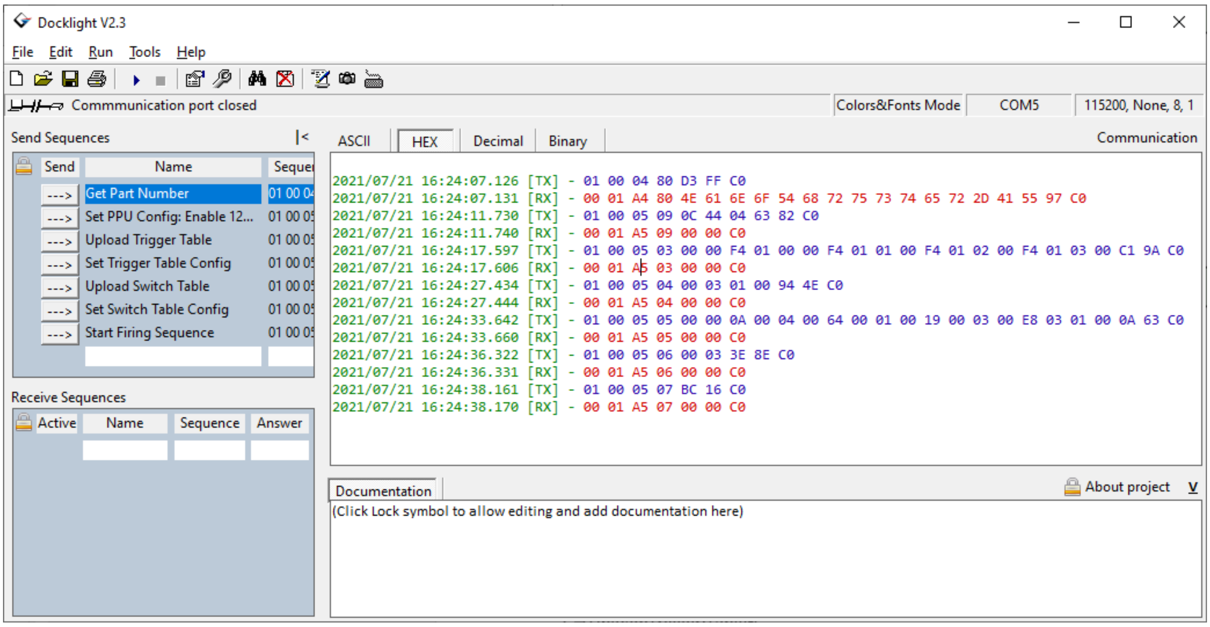


Figure A2-2: Docklight Serial Communications Tool