MILLSWOOD engineering

Generator PMU

CAN PROTOCOL

Introduction

The PMU comes with a choice of two communications interfaces: RS232 or CAN (Controller Area Network). This document describes the default CAN communications interface and protocol (RS232 is described in the Generator PMU datasheet). Custom CAN protocols can also be developed to suit existing CAN ID structures; please contact us to discuss your requirements. RS232 and CAN have different part numbers, and the desired communications interface is specified at the time of ordering.

Overview of CAN

CAN is a multi-master broadcast serial bus, originally developed for automotive applications but now used extensively across a wide range of industries. CAN provides more robust communications than is possible with RS232, and includes automatic arbitration-free transmission, message prioritisation, automatic retries, CRC data protection, fault confinement and more.

Physically CAN is usually implemented as a 2-wire differential serial bus, although a third ground wire is always recommended. A 120 Ohm termination resistor must be fitted between the two signal lines at each end of the bus to reduce signal reflections. The PMU can be fitted with this resistor if required.

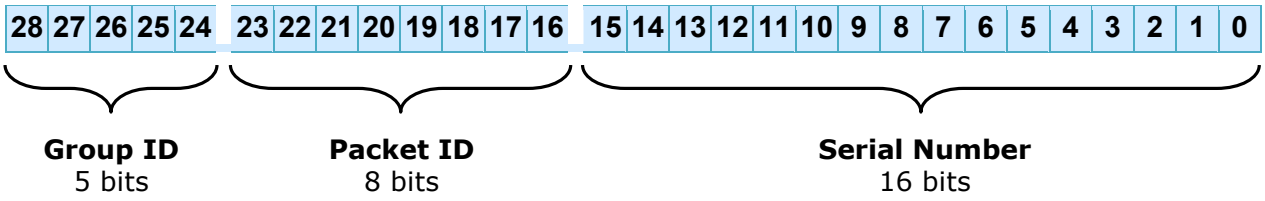
The baud rate is 1Mbit/sec.

The CAN specification defines four frame types (data, remote, error and overload), but only the data frame can actually transmit any payload data. Like many CAN implementations, only the data frame is used here. Data frames can have 0 to 8 bytes of payload data.

This protocol is based on CAN 2.0B; i.e. CAN frames have a 29-bit message identifier associated with them. The message ID is divided into 3 parts as described in the next section.

CAN Message ID

Each CAN message is preceded by the 29-bit CAN message ID, which specifies the device type (group ID), the message type (packet ID), and the serial number of the device associated with the message:



Group ID: Identifies the category of device that this CAN frame came from or is being sent to. In Cloud Cap Technology parlance the PMU belongs to the POWERBOARD category, and as such always has a group ID of 30 (0x1E).

Packet ID: Identifies the contents of a packet. There are 6 different packet types defined for the PMU:

Packet ID	Packet Type	Description
0x20	STREAM	System operational data
0x21	STREAM_2	Additional system operational data
0x22	CONFIG	System configuration data
0x23	CONFIG_2	Additional system configuration data
0x24	FIRMWARE	System firmware information
0x25	COMMAND	Command packet

Serial number: Each PMU within any given network must have a unique serial number between 0 and 65534 inclusive. 65535 (0xFFFF) is reserved to form a "broadcast" message ID to which all PMUs will respond. PMUs are shipped with a default serial number of 1.

CAN Packet Types

Please note that all 2-byte quantities are transmitted and received in big-endian format; i.e. high byte first, followed by the low byte.

STREAM

This packet contains measured system voltages and currents. The PMU can be configured to stream this packet at regular intervals, or it can be requested using the COMMAND packet.

Packet ID: 0x20
Packet Length: 7 bytes

Byte	Name	Description
0, 1	Input voltage	A 16-bit unsigned integer representing the rectified (DC) input voltage. It is 10 times the measured voltage (i.e. in 0.1V increments). Ranges from 0 to 800 (0.0 to 80.0V).
2, 3	Input current	A 16-bit signed integer representing the rectified (DC) input current. It is 10 times the measured current (i.e. in 0.1A increments). Ranges from -280 to +280 (-28.0 to +28.0A).
4, 5	Main output voltage	A 16-bit unsigned integer representing the main output voltage. It is 10 times the measured voltage (i.e. in 0.1V increments). Ranges from 0 to 350 (0.0 to 35.0V).
6	Main output current	An 8-bit unsigned byte representing the main output current. It is 10 times the measured current (i.e. in 0.1A increments). Ranges from 0 to 255 (0.0 to 25.5A).

STREAM_2

This packet contains additional measured system voltages and currents, as well as some other miscellaneous derived and measured data. The PMU can be configured to stream this packet at regular intervals, or it can be requested using the COMMAND packet.

Packet ID: 0x21
Packet Length: 8 bytes

Byte	Name	Description
0	Auxiliary voltage	An 8-bit unsigned byte representing the auxiliary output voltage. The value returned is 10 times the measured voltage (i.e. in 0.1V increments). Ranges from 0 to 65 (0.0 to 6.5V).
1	Auxiliary current	An 8-bit unsigned byte representing the auxiliary output current. The value returned is 10 times the measured current (i.e. in 0.1A increments). Ranges from 0 to 55 (0.0 to 5.5A).
2, 3	Battery voltage	A 16-bit unsigned integer representing the battery voltage. The value returned is 10 times the measured voltage (i.e. in 0.1V increments). Ranges from 0 to 350 (0.0 to 35.0V).
4	Battery current	An 8-bit signed byte representing the battery current. The value returned is 10 times the measured current (i.e. in 0.1A increments). Ranges from -128 to +127 (-12.8 to +12.7A).
5, 6	Battery energy	A 16-bit signed integer representing the accumulated battery current since power-up in mAh. Ranges from –32768 to +32767.
7	Temperature	An 8-bit signed byte representing the internal air temperature in degrees Celsius. Ranges from -128 to +127.

CONFIG

This packet contains the current values of the PMU's main electrical parameters. This packet must be requested from the PMU using the COMMAND packet. These electrical parameters are all programmable by using the appropriate Set command within a COMMAND packet.

Packet ID: 0x22
Packet Length: 6 bytes

Byte	Name	Description
0, 1	V_M	A 16-bit unsigned integer representing the desired main output voltage. The value returned is 10 times the desired voltage (i.e. in 0.1V increments). Ranges from 100 to 300 (10.0 to 30.0V).
2	I_B	An 8-bit unsigned byte representing the desired maximum battery charge current. The value returned is 10 times the desired current (i.e. in 0.1A increments). Ranges from 20 to 50 (2.0 to 5.0A).
3	I_G	An 8-bit unsigned byte representing the desired maximum input current. The value returned is 10 times the desired current (i.e. in 0.1A increments). Ranges from 0 to 250 (0.0 to 25.0A).
4, 5	P_G	A 16-bit unsigned integer representing the desired maximum input power in Watts. Ranges from 0 to 999.

CONFIG_2

This packet contains additional miscellaneous programmable parameters. This packet must be requested from the PMU using the COMMAND packet. These parameters are also programmable using the appropriate Set command within a COMMAND packet.

Packet ID: 0x23
Packet Length: 5 bytes

Byte	Name	Description
0	P_P	An 8-bit unsigned byte representing the packet period in multiples of 100ms. If data streaming is enabled then packets will be transmitted at this rate. Ranges from 1 to 255.
1	Streaming Settings	An 8-bit field storing packet streaming settings. Bits are as follows, where B_7 = MSB and B_0 = LSB: B_7 – Reserved B_6 – Reserved B_5 – Reserved B_4 – Reserved B_3 – Reserved B_2 – Reserved B_1 – 1 = Enable streaming of STREAM_2 packet (0 = disable) B_0 – 1 = Enable streaming of STREAM packet (0 = disable)
2	I_0	An 8-bit value used to calibrate the battery current sensor. Ranges from 0 to 255.
3	T_0	An 8-bit value used to calibrate the temperature sensor. Ranges from 0 to 255.
4	T_U	An 8-bit unsigned byte representing the upper temperature limit in degrees Celsius. Ranges from 0 to 255.

FIRMWARE

The firmware packet contains information on the PMU firmware version. This packet must be requested from the PMU using the COMMAND packet.

Packet ID: 0x24
Packet Length: 6 bytes

Byte	Name	Description
0	Version Major	Unsigned byte, range 0 – 99.
1	Version Minor	Unsigned byte, range 0 – 99.
2	Revision Day	Unsigned byte, range 1 – 31.
3	Revision Month	Unsigned byte, range 1 – 12.
4, 5	Revision Year	Unsigned word, ranges from 2011 and up.

COMMAND

The command packet is used for issuing various commands to the PMU. This packet is only received by the PMU.

Packet ID: 0x25
Packet Length: Variable (1 – 3)

Byte	Name	Description
0	Command	See list of available commands below
1	Data_0	Data Byte 0 (may or may not be used)
2	Data_1	Data Byte 1 (may or may not be used)

Available Commands

ID	Command	Data Bytes	Description
0x00	Set Serial	2	Change the serial number
0x01	Set V_M	2	Set the main output voltage
0x02	Set I_B	1	Set the maximum battery charge current
0x03	Set I_G	1	Set the input current limit
0x04	Set P_G	2	Set the input power limit
0x05	Set P_P	1	Set the interval between streamed packets
0x06	Set Stream	1	Configure which packets to stream
0x07	Set I_0	1	Set the battery current offset calibration value
0x08	Set T_0	1	Set the temperature offset calibration value
0x09	Set T_U	1	Set the upper temperature limit
0x0A	Set V_0	0	Factory use only – do not use
0x0B	Request Stream	0	Request system stream packet
0x0C	Request Stream_2	0	Request system stream_2 packet
0x0D	Request Config	0	Request system config packet
0x0E	Request Config_2	0	Request system config_2 packet
0x0F	Request Version	0	Request firmware version packet

The *data bytes* field in the table above describes the number of bytes *after* the command byte.

Set Serial command

To set the serial number of the PMU, send the command packet ID (0x25) with Set Serial command (0x00) as the first payload byte, followed by a 16-bit unsigned integer representing the desired serial number.

The default serial number is 1, but the serial number may be set to any value from 0 to 65534 inclusive.

65535 (0xFFFF) is a "broadcast" serial number to which all PMUs will respond (provided of course that the rest of the ID is valid). This is useful for determining unknown or forgotten serial numbers.

Set V_M command

To set the main output voltage, send the command packet ID (0x25) with the Set V_M command (0x01) as the first payload byte, followed by a 16-bit unsigned integer representing the desired main output voltage in 0.1V increments.

V_M may be set to any value from 10.0 to 30.0V inclusive, corresponding to unsigned integer values of 100 to 300.

Set I_B command

To set the maximum battery charge current, send the command packet ID (0x25) with the Set I_B command (0x02) as the first payload byte, followed by an unsigned byte representing the desired maximum battery charge current in 0.1A increments.

I_B may be set to any value from 2.0 to 5.0 A inclusive, corresponding to unsigned byte values of 20 to 50.

Set I_G command

To set the input current limit, send the command packet ID (0x25) with the Set I_G command (0x03) as the first payload byte, followed by an unsigned byte representing the desired maximum input current in 0.1A increments.

I_G may be set to any value from 0.0 to 25.5A inclusive, corresponding to unsigned byte values of 0 to 255. Values greater than 15.0A are not recommended.

Set P_G command

To set the input power limit, send the command packet ID (0x25) with the Set P_G command (0x04) as the first payload byte, followed by an unsigned integer representing the desired maximum input power in Watts.

P_G may be set to any value from 0 to 999W inclusive.

Set Packet Period

To configure the packet period, send the command packet ID (0x25) with the Set Packet Period command (0x05) as the first payload byte, followed by an unsigned byte representing the desired data period in 100ms increments.

The data period may be set to any value from 0.1 to 25.5 seconds, corresponding to unsigned byte values of 1 to 255.

Set Stream

The PMU can be configured to continuously stream both the STREAM and STREAM_2 packets, if desired. Send the command packet ID (0x25) with the Set Stream command (0x06) followed by the stream configuration byte. The bit-field description of the configuration byte is as follows:

Bit	Description
0	1 = Enable streaming of STREAM packet (0 = disable)
1	1 = Enable streaming of STREAM_2 packet (0 = disable)
2	Reserved
3	Reserved
4	Reserved
5	Reserved
6	Reserved
7	Reserved

Set I_0 command

To zero the battery current offset, send the command packet ID (0x25) with the Set I_0 command (0x07) as the first payload byte, followed by an unsigned byte representing the desired offset.

I_0 may be set to any value from 0 to 255 inclusive. I_0 is set at the factory and should not normally need changing.

Set T_0 command

To zero the temperature offset, send the command packet ID (0x25) with the Set T_0 command (0x08) as the first payload byte, followed by an unsigned byte representing the desired offset.

T_0 may be set to any value from 0 to 255 inclusive. T_0 is set at the factory and should not normally need changing.

Set T_U command

To set the upper temperature limit, send the command packet ID (0x25) with the Set T_U command (0x09) as the first payload byte, followed by an unsigned byte representing the desired maximum operating temperature in degrees Celsius.

T_U may be set to any value from 0 to 255 degrees Celsius inclusive. There is approximately 10% hysteresis. Setting T_U to 0 will turn the main converter off, and setting T_U to 255 will disable thermal shutdown. Note that thermal shutdown **DOES NOT** affect the main and auxiliary outputs as long as a battery is connected; it merely shuts down power generation from the 3-phase inputs. T_U is set to 85 degrees Celsius at the factory.

Set V_0 command

Factory use only – do not use.

Request Stream command

To request the system stream information, send the command packet ID (0x25) with the Request Stream command (0x0B) as the first payload byte. The PMU will respond by sending a Stream packet.

Request Stream_2 command

To request the additional system stream information, send the command packet ID (0x25) with the Request Stream_2 command (0x0C) as the first payload byte. The PMU will respond by sending a Stream_2 packet.

Request Config command

To request the system configuration information, send the command packet ID (0x25) with the Request Config command (0x0D) as the first payload byte. The PMU will respond by sending a Config packet.

Request Config_2 command

To request the additional system configuration information, send the command packet ID (0x25) with the Request Config_2 command (0x0E) as the first payload byte. The PMU will respond by sending a Config_2 packet.

Request Version command

To request the firmware version information, send the command packet ID (0x25) with the Request Version command (0x0F) as the first payload byte. The PMU will respond by sending a Firmware packet.

Further Information

Visit us on the web at www.millswoodeng.com.au

Didn't find what you wanted? Send us an email or give us a call – contact details are on our website.

The Fine Print

Regarding this document: Millswood Engineering makes no warranty, representation or guarantee regarding the accuracy or completeness of this document and reserves the right to make changes to specifications and product descriptions at any time without notice.

Regarding this product: Millswood Engineering makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Millswood Engineering assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation special, consequential or incidental damages. Millswood Engineering reserves the right to make changes without further notice to any products herein to improve reliability, function or design.

Regarding typical specifications: "Typical" parameters which may be provided in Millswood Engineering datasheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including "Typicals" must be validated for each customer application by customer's technical experts.

Regarding life support applications: Millswood Engineering products are not designed, intended, or authorised for use as components in systems intended to support or sustain life, or for any other application in which the failure of the Millswood Engineering product could create a situation where personal injury or death may occur.

Regarding intellectual property: No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted under this document.