

## 1 Introduction

The 1600W GCU has both RS232 and CANbus interfaces that perform essentially the same functions, these being:

- Configuration (of parameters stored in the GCU's non-volatile memory),
- Control (real-time control of the GCU's various features),
- Monitoring (of measured voltages, currents, temperatures, etc), and
- Updating the GCU's firmware.

Once the GCU has been configured, there is no requirement to connect anything to either communications interface – the GCU will operate quite normally with no communications at all.

The RS232 interface operates at 57600 baud, full-duplex, with 8 data bits and no parity (57600 8N1). The RS232 hardware layer is compliant with TIA/EIA-232-F and ITU V.28.

## 2 Software

A Windows application that provides easy access to most of the 1600W GCU's various features may be downloaded from [www.millswoodeng.com.au/resources.html](http://www.millswoodeng.com.au/resources.html)

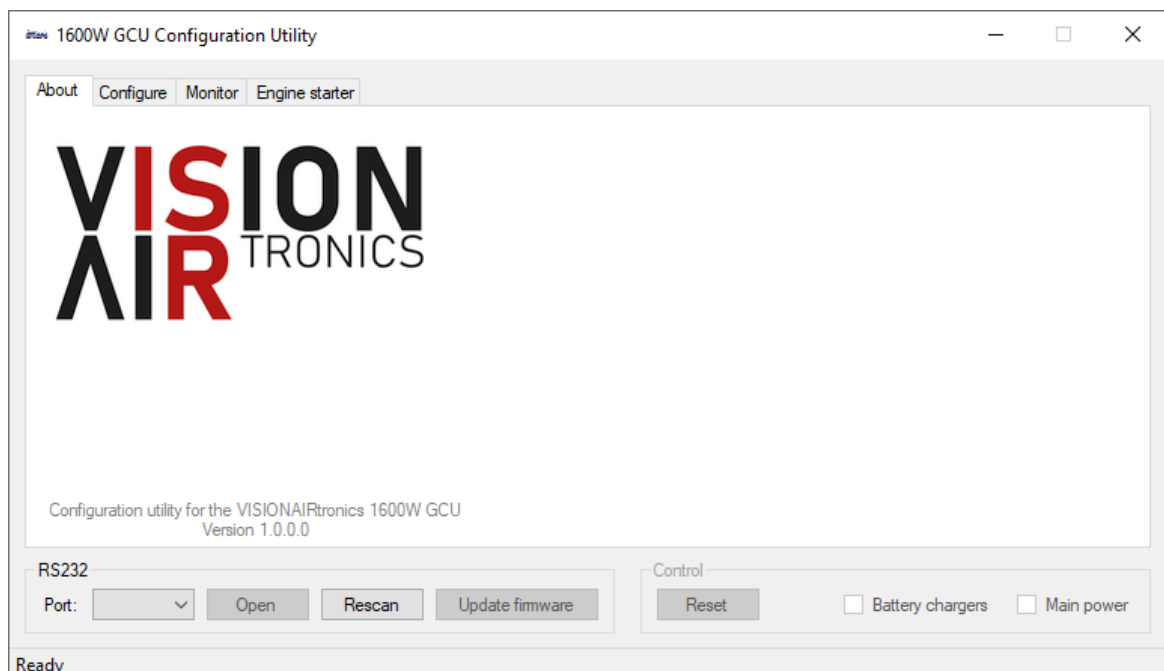


Figure 1 – 1600W GCU Configuration Utility

Using the configuration utility relieves the user from the burden of writing software in order to configure and control the GCU. There is no need to read any more of this document if the configuration utility is used – the RS232 protocol is provided solely for the purpose of more tightly integrating the GCU with other hardware and software.

This page intentionally left blank.

### 3 Contents

1	Introduction .....	1
2	Software .....	1
3	Contents .....	3
4	Commands.....	4
4.1	Status commands .....	4
4.1.1	Show settings .....	4
4.2	Configuration settings.....	4
4.2.1	Set Battery charging voltage .....	4
4.2.2	Set Battery charging current.....	4
4.2.3	Set Packet period .....	5
4.2.4	Set Packets streamed .....	5
4.2.5	Set Temperature limit.....	5
4.2.6	Set Initial state .....	5
4.2.7	Set CAN Baudrate .....	5
4.2.8	Set CAN address .....	5
4.3	Engine starter settings.....	6
4.3.1	Set Cranking timeout.....	6
4.3.2	Set Cranking speed .....	6
4.3.3	Set Cranking current .....	6
4.3.4	Set Cranking flags.....	6
4.3.5	Set Pole count .....	6
4.4	Control commands .....	6
4.4.1	Battery chargers .....	6
4.4.2	Main power .....	6
4.4.3	Engine starter module .....	6
4.4.4	Reset.....	6
5	Monitoring .....	7
5.1	Packet type 0 – Voltage .....	7
5.2	Packet type 1 – Current .....	7
5.3	Packet type 2 – Battery status .....	7
5.4	Packet type 3 – Temperature.....	8
5.5	Packet type 4 – Miscellaneous.....	8
6	Document version history .....	9

## 4 Commands

The GCU supports a number of commands via its RS232 interface. The command processor is not case sensitive, but commands must be entered exactly as shown below, with the correct number of digits and no extraneous spaces. A carriage return or linefeed character is required to terminate each command. A confirmation string is issued if the command is successful.

### 4.1 Status commands

#### 4.1.1 Show settings

**Show** Displays the GCU's settings and enabled features.

The first packet returned shows the stored configuration settings and is 96 bytes long (including CR & LF), and is formatted as follows:

```
C o n f i g u r a t i o n   s e t t i n g s :   V b = 5 0 . 4 V   I b =
4 . 0 A   P p = 0 1 . 0 S   P s = 2 5 5   T u = 0 8 5 C   S 0 = 2
5 5   C b r p = 0   C a = 0 0 0 0 1 CR LF
```

The second packet returned shows the stored engine starter settings and is 71 bytes long (including CR & LF), and is formatted as follows:

```
E n g i n e   s t a r t e r   s e t t i n g s :   C t = 0 5 . 0 S   C s
= 0 1 5 0 0 R P M   I c = 3 5 A   C f = 1 2 8   P c = 0 8 CR LF
```

The third packet returned shows the enabled features and is 44 bytes long (including CR & LF), and is formatted as follows:

```
E n a b l e d   f e a t u r e s :   B C : E n a b l e d   M P : E n a
b l e d CR LF
```

*BC = Battery chargers*

*MP = Main power*

### 4.2 Configuration settings

The following commands store configuration settings in non-volatile memory:

#### 4.2.1 Set Battery charging voltage

**Set Vb=XX.X** Where XX.X is the nominal (fully charged) battery voltage in Volts. V<sub>B</sub> may be set to any value from 40.0 to 55.0 inclusive.

#### 4.2.2 Set Battery charging current

**Set Ib=X.X** Where X.X is the maximum battery charging current (for each battery) in Amps. I<sub>B</sub> may be set to any value from 0.5 to 8.0 inclusive.

#### 4.2.3 Set Packet period

**Set Pp=XX.X** Sets the packet streaming period. XX.X is the desired interval of time between successive transmissions in seconds. Pp may be set to any value from 00.1 to 25.5 inclusive.

#### 4.2.4 Set Packets streamed

**Set Ps=XXX** Sets which packets types are streamed. XXX is a decimal value ranging from 000 to 255 inclusive. Bits have the following significance:

- Bit 0: Set to stream packet type 0 (voltages)
- Bit 1: Set to stream packet type 1 (currents)
- Bit 2: Set to stream packet type 2 (battery statuses)
- Bit 3: Set to stream packet type 3 (temperatures)
- Bit 4: Set to stream packet type 4 (miscellaneous data)
- Bits 5 to 7: Irrelevant

#### 4.2.5 Set Temperature limit

**Set Tu=XXX** Where XXX is the upper temperature limit in degrees Celsius.  $T_U$  may be set to any value from 000 to 255 inclusive. There is approximately 10% hysteresis. Setting  $T_U$  to 0 forces the GCU into thermal shutdown, and setting  $T_U$  to 255 disables thermal shutdown. The main power output and both battery chargers are turned off during thermal shutdown.

#### 4.2.6 Set Initial state

**Set S0=XXX** Sets the GCU's initial state.  $S_0$  defines how the GCU's miscellaneous features are configured after power-up or reset. XXX is a decimal value ranging from 000 to 255 inclusive. Bits have the following significance:

- Bit 4: Battery disconnect detect (1 = enabled)
- Bit 6: Keep fans on (1 = enabled)
- Bits 1 to 3, 5 and 7: Irrelevant

#### 4.2.7 Set CAN Baudrate

**Set Cbrp=X** Sets the CAN Baudrate according to the following formula:

$$\text{Baudrate (kb/s)} = \frac{1000}{X + 1}$$

Values for X other than 0 (1Mb/s), 1 (500kb/s), 3 (250kb/s) and 7 (125kb/s) have not been tested.

#### 4.2.8 Set CAN address

**Set Ca=XXXXX** Sets the 16-bit CAN address. XXXXX may be set to any value from 00000 to 65535 inclusive. See the CAN protocol document for further information.

### 4.3 Engine starter settings

The following commands store engine starter settings in non-volatile memory:

#### 4.3.1 Set Cranking timeout

**Set Ct=XX.X** Sets the maximum cranking time for the engine starter. XX.X is the time limit in seconds, and may be set to any value from 00.5 to 25.0 inclusive.

#### 4.3.2 Set Cranking speed

**Set Cs=XXX** Sets the speed at which the engine is deemed to have started. XXX is the BLDC rotational speed in RPM x 100, and may be set to any value from 001 to 255 inclusive (corresponding to BLDC speeds of 100 to 25500 RPM).

#### 4.3.3 Set Cranking current

**Set Ic=XX** Sets the current that will be drawn when starting the engine. XX is the current in Amps, and may be set to any value from 01 to 50 inclusive.

#### 4.3.4 Set Cranking flags

**Set Cf=XXX** Sets flags that control the behaviour of the engine starter. XXX is a decimal value ranging from 000 to 255 inclusive. Bits have the following significance:

Bit 0: Direction of rotation  
Bits 1 to 7: Irrelevant

#### 4.3.5 Set Pole count

**Set Pc=XX** Sets the number of poles in the BLDC motor. This value is used to calculate the mechanical RPM of the BLDC motor from its electrical RPM, and may be set to any even number from 2 to 32 inclusive.

### 4.4 Control commands

The following commands perform actions (but do not store anything in non-volatile memory):

#### 4.4.1 Battery chargers

**Enable BC** Enables the battery chargers.

**Disable BC** Disables the battery chargers.

#### 4.4.2 Main power

**Enable MP** Enables the main power output.

**Disable MP** Disables the main power output.

*An "Enabled features" packet is returned on receipt of a valid "Enable" or "Disable" command.*

#### 4.4.3 Engine starter module

**Start** Starts cranking the engine.

**Stop** Stops cranking the engine. This command is included for safety reasons only (the engine starter automatically disengages when it detects that the engine has started).

#### 4.4.4 Reset

**Reset** Restarts the PMU. The battery statuses Eba and Ebb are re-initialised to zero (see packet type 2 below).

## 5 Monitoring

The GCU measures and reports a number of quantities via its RS232 interface. The measured data is formatted into a human-readable plain text packets that are streamed regularly at a user-defined rate (configured by the Set P<sub>P</sub> command). Packet streams may be turned on and off individually using the Set P<sub>S</sub> command.

Measurements are fixed width, i.e. leading zeros are always included. Measurements are separated from each other by pairs of spaces, and packets are terminated by both carriage returns and linefeeds.

### 5.1 Packet type 0 – Voltage

Packet type 0 is 66 bytes (including CR & LF), and is formatted as follows:

```
V b a = 5 0 . 4 V   V b b = 5 0 . 4 V   V m = 5 5 . 0 V   V g = 1 1
0 . 5 V   V u = 0 6 0 . 0 V   V h v = 1 0 9 . 0 V CR LF
```

*Vba* = Battery A voltage  
*Vbb* = Battery B voltage  
*Vm* = Main power output voltage  
*Vg* = Generator (BLDC motor) voltage  
*Vu* = Umbilical input voltage  
*Vhv* = Internal HV bus voltage

*Vg* is a DC value. Generator voltage is measured after the active rectification process.

### 5.2 Packet type 1 – Current

Packet type 1 is 50 bytes (including CR & LF), and is formatted as follows:

```
I b a = 7 . 9 A   I b b = 3 . 5 A   I m a = 4 . 1 A   I m b = 4 . 3
A   I g = 1 5 . 3 A CR LF
```

*Iba* = Battery A current  
*Ibb* = Battery B current  
*Ima* = Main power output current, A-side  
*Imb* = Main power output current, B-side  
*Ig* = Generator (BLDC motor) current

### 5.3 Packet type 2 – Battery status

Packet type 2 is 30 bytes (including CR & LF), and is formatted as follows:

```
E b a = + 0 0 0 0 0 m A H   E b b = + 0 0 0 0 0 m A H CR LF
```

*Eba* = Battery A energy  
*Ebb* = Battery B energy

*Eb* is similar to a fuel gauge, except that it represents the change in energy stored (since power-up), rather than the total amount of energy stored. A positive value means that the battery has had a net gain in energy since power-up (i.e. it has been charged).

#### 5.4 Packet type 3 – Temperature

Packet type 3 is 51 bytes (including CR & LF), and is formatted as follows:

```
T = + 0 3 1 C   T b a = + 0 1 9 C   T b b = + 0 2 2 C   T g = + 0 0
1 C   T e = + 1 0 0 C CR LF
```

*T* = Internal GCU temperature

*Tba* = Battery A temperature

*Tbb* = Battery B temperature

*Tg* = Generator (BLDC motor) temperature

*Te* = Engine starter module temperature

For external temperature sensing the GCU will generally return a value of -128C if no temperature sensing device is present. In the case of the engine starter module not being fitted the corresponding value is indeterminate.

#### 5.5 Packet type 4 – Miscellaneous

Packet type 4 is 45 bytes (including CR & LF), and is formatted as follows:

```
S g = 0 9 5 4 0 R P M   R 0 = 0 0 0   R 1 = 0 0 0   R 2 = 0 0 0
R 3 = 0 0 0 CR LF
```

*Sg* = Generator (BLDC motor) speed

*R0* = Flag register 0 – Bit 5: Thermal shutdown flag (1 = shutdown)

*R3* = Flag register 3 – Bit 7: Engine starter module status flag (1 = ready)

*Sg* is only valid if Hall sensors are fitted to the BLDC motor and connected, and the number of poles has been correctly specified.



## 6 Document version history