

ZENER VARIDRIVE SOLUTIONS

ZENER SMARTSTART® 6000

Data logger Reference Manual

Description:

- The Data Logger Option Card provides an SD Card slot to record the soft starter's parameters and operating data.
 1. Operating data is logged at the following rates:
 - i) A rate of 1/10second during acceleration and Deceleration.
 - ii) Every 10 seconds during running.
 - iii) Every 10seconds when in idle mode. ie. control supply 'ON' but starter not running
 2. The Soft Starter parameters are logged onto the data card when the unit is powered and if any parameters are changed.
- The data is stored as a text file (.txt) on to the SD card inserted. The SD card should not be removed while running or a short period after as recent data may be lost.
- The data may be read using 'Notepad' or other application capable of reading text files or to extract and report the data required.

NOTE: The data cannot be read using Modbus communications.

- The Data Logger Option Card Kit includes the following:
 - 4x M2.5mm Screws (H001499)
 - 2x M2.5 x 10mm Nylon Spacers (H6052)
 - 1x Option Card

Compatibility:

SMARTSTART 6000 firmware revision 0.96X or later is required.

SD Cards Accepted:

Standard SD Card;



Micro SD Card with Adapter



If the SD card has a write protect tab ensure it is not in the lock position.



NOTE: Earlier Data Logger PCB's could accept 1G or 2G SD cards only

IMPORTANT: The SD card must only be inserted & removed when control power is off.

Operation:

- With the control power to the soft starter off. Insert the SD card.
- Power up the control circuit. The Data Logger is now operational.
- The soft starter will create 2 files onto the data card:
 - i) config.txt file 'config.txt', which configures the Baud Rate and Data Format.
 - ii) SEQLOG or LOGXXXX.txt file. This is where the data will be stored. A new text file will be created each time the control supply is cycled or can save data to the one text file.
- The text file may be deleted on a computer at any stage, and a new file will be created.
- The text files may be deleted, copied and/or sent via email to Zener for analysis should a problem exist.

Warning:



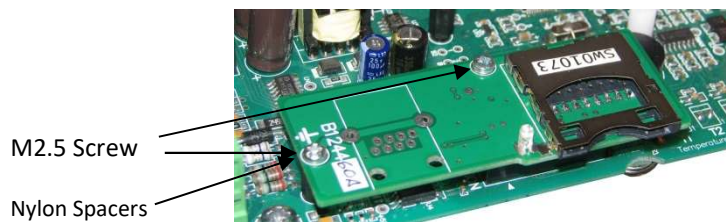
Your Smartstart® and options must be applied and installed by a suitably qualified and experienced electrical tradesperson in accordance with the manual, good engineering practice and all local rules and regulations. There are hazardous voltages inside the Smartstart® whenever it is connected to an electrical supply.

Do not operate with the covers removed or the doors of the enclosure in which it is installed open. Do not touch the terminals of the Smartstart® or any associated motor and wiring when it is energised, even if the Smartstart® and motor are stopped. Electric shock may result.

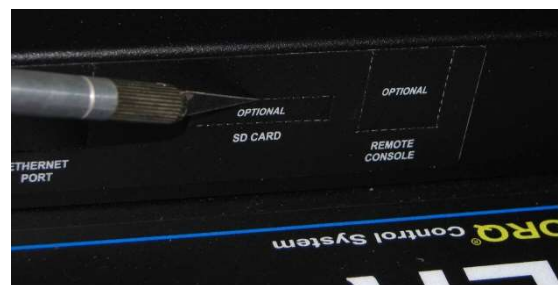
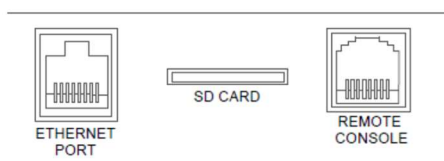
Before removing any covers to fit any options, do repairs or to modify any wiring ensure all power is off and isolated from the starter. Ensure all power supplies connected to the control terminals are switched off and isolated.

Installation:

1. Remove the top cover. There are 4x hex type screws holding the top cover in place.
2. The Option Board plugs into the control board into the socket marked 'OPTION' located on the right hand side of the control Board.
3. There are 2 nylon spacers and 4x 2.5mm screws supplied to secure the option card in place. The hex spacers (4) and nuts (2) securing the control board to the chassis will need to be carefully removed to access the underside to fit the nylon spacers. Do not remove plugs/cables unless necessary.



4. Carefully cut out the label marked 'SD Card'.

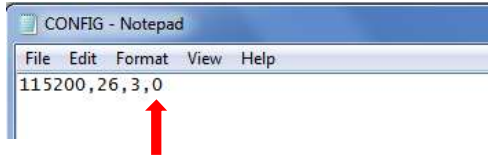


5. Secure the control board in place ensuring all connectors/plugs are secure and screws in place. An insulation piece should be positioned over the gate leads (3x connectors located at the top end of the control board).

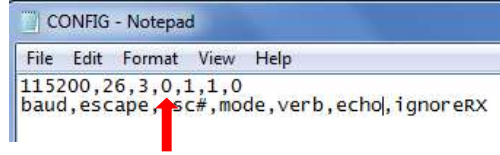
Modifying the config.txt file:

To change the logging behavior, write over the last digit (1) or the fourth digit (2) and save the changes.

(1) Old Format



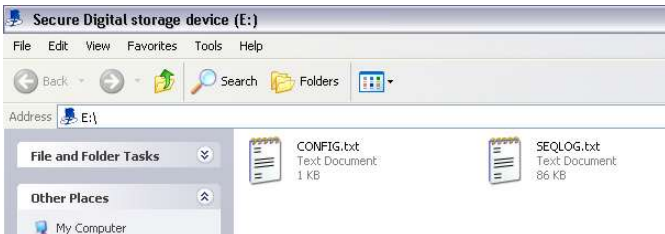
(2) Current format



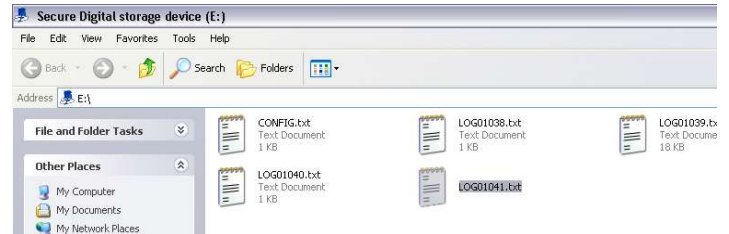
This digit sets the logging behavior as follows:

- 0 = Creates a new data file when power is cycled
- 1 = Saves data in same file 'SEQLOG.txt'

Single storage TXT File = 1



Multiple storage TXT Files = 0



Logger Format:

This section defines the format of information saved to the Data Logger Option Card of the SMARTSTART® 6000.

All characters logged are standard ASCII text and the log file can be viewed in a standard text editor (e.g. Windows Notepad). Be aware that unexpected and/or non-ASCII characters may also be saved to the log file on system power up and power down. Some offline parsing tools have been developed to help analyse data log files (e.g. to extract process data and produce trend graphs)

Each entry in the log file is a single line of ASCII text with the following format:

<Time> <Token> <Data> [Comment] <Newline>

Examples of log entries:

```
13:56:50.00 $mcuDown 9435 ; s
13:56:50.00 $product Zener Smartstart 6R060
13:56:50.00 $version V1.12B
13:56:50.00 $date 12-12-2013 ; DD-MM-YYYY
13:56:50.00 $rptFmt 1.08
13:56:50.00 $rptOps Off Stby Rdy Kick Acc UTS Cont BP Dec Trip
13:56:52.20 $parSet M01 Motor Amps = 56.0A ; ****
13:56:52.40 $parDef M02 Motor Volts = 415V
14:54:07.00 $note FAN OFF
51.623280 $warn PSU DIP/ISR
16:47:35.81 $trip T13 MOTOR LOSS
22:50:20.67 $rptVars Op V Hz A kW Torq Vn Ta Ts Tj MTC Rt
22:50:20.67 $rpt B 433 50.0 52.4 21.6 53 0 23 35 55 75 250
19.123000 $note GATES RUN
19.123112 $note BP OFF
11:15:45.99 $tripElec 431V 50.0Hz 59.2A 23.4kW 63%Torq
```

Log Entry Fields:

Field	Description
Time	<p>Each log entry begins with the time that the log entry was queued for output to the logger. Most log entries begin with the standard time stamp while critical events are logged with the precision time stamp to allow optimum diagnosis of unit operation.</p> <p>Standard time stamp. Format is: HH:MM:SS.FF (e.g. "19:45:55.75"). Rolls over at midnight from "23:59:59.99" to "00:00:00.00". The fractional part of seconds ("FF") provides a resolution of 10ms. The clock is synchronised from the internal battery-backed Real Time Clock (RTC) which can be set via the "Display" menu ("D10 Time"). The clock is NOT auto-adjusted for summer time. The current date is also periodically logged (search for "\$date").</p> <p>Precision time stamp. Format is: SS.FFFFFFF (e.g. " 20.123456"). Rolls over every minute from "59.999999" to "00.000000". The fractional part of seconds ("FFFFFF") provides a resolution of 1us, however the accuracy is of the order of 20us as it does take time to sample the clock and queue the log entry.</p>
Token	The token always starts with the '\$' character followed by alphanumeric characters. The token is terminated by white space. The token identifies the type of log entry. These tokens assist offline parsing of the log file and are also useful when searching for specific information in a large log file (e.g. search for "\$strip " to find trips). Specific tokens are described under "List of Tokens".
Data	The data field varies depending on type of entry (defined by token). This can range from fixed text string (e.g. "FAN ON") to a log entry with multiple data values (e.g. see Process Data Reporting).
Comment	A log entry may include an optional comment on the end of the line which is delimited with a semi-colon (;). For example the format of \$date is given by the comment "; DD-MM-YYYY". The comment sequence "*****" is appended to some entries to highlight important information (e.g. a non-default parameter value).
Newline	The <CR><LF> sequence (0x0D, 0x0A) terminates each line (DOS/Windows convention).

Event Logs:

Description	Examples (time stamp not shown)
Events are logged with the \$note, \$warn or \$strip token which indicates the severity of the event (e.g. search for "\$strip " to locate trip events). Some events are logged with the precision time stamp (e.g. ERASE, SAVE, BP, GATES, PSU, BRK events). Some event messages include embedded values (e.g. ERASE & SAVE events include a record count). For specific information, see List of Events.	\$note EN ON \$note RL1 ON \$note LINE ON \$strip T21 STR OVERTEMP \$note GATES OFF \$warn PSU DIP/ISR \$note SAVE DONE 21 \$warn PSU OFF

Multi-line Reports:

Report / Description	Example (time stamp not shown)
<p>Boot-up Report Events at the beginning of this report (before \$mCuDown) might not be captured by the Data Logger Option Card as it takes a few seconds to boot. Use terminal software and a TTL/RS232 converter to capture the complete boot-up report.</p>	<pre>\$note MCU START \$note TRIP RESET: POWER UP \$note FAN ON \$note PSU OK \$note NET RESET \$mCuDown 1234 ; s \$mCuReset 1 ; 2s ago \$mCuInfo ##### ## \$mTrTol 87.6 23.4 19.8 ; % rec adj lev</pre>
<p>System Information Report After boot up and every hour, system information is logged. Includes model, firmware version and date information.</p>	<pre>\$product Zener Smartstart 6R060* ; CUSTOM \$version V1.20A \$date 25-06-2014 ; DD-MM-YYYY \$rptFmt 1.08 \$rptOps Off Stby Rdy Kick Acc ... Dec Trip</pre>
<p>Service Report After boot up and at midnight, service info is logged. This includes counters, meters and stress records. Counters and/or meters may have been cleared at any time by customer. If a logger board was plugged in to get this report, the Ta values will NOT correctly reflect operational history.</p>	<pre>\$starts 3452367 \$trips 123413 \$hrs 12345.6 \$kWh 9878374.3 \$ssTrips 5 98 23 ; Is Ts Tj \$maxStd 370% 45C 85C 98C ; Is Ta Ts Tj \$maxEso 570% 55C 95C 110C ; Is Ta Ts Tj \$esoSecs 32 142 214 ; Is Ts Tj</pre>
<p>Configuration Report Shortly after boot up, all customer parameter name-value pairs are logged with the \$parDef or \$parSet token every 0.1s. These can be interleaved with process data reports (e.g. when motor starts after system boot up). Parameters with a non-default value are logged with \$parSet and appended with “; ****”. Factory parameters are not logged unless they have non-default values (normally only “S01 Model ID”). The configuration report is concluded with the \$parSetCnt log entry which indicates how many parameters have non-default values. Whenever a user edits and saves a parameter, the name-value pair is logged with the \$parSave token.</p>	<pre>\$parSet M01 Motor Amps = 14.0A ; **** \$parDef M02 Motor Volts = 415V \$parDef M03 Motor PF = 0.90 \$parDef C01 Run Mode = Bypass \$parDef C02 Curr Limit = 450% \$parSet C11 Accel Time = 15s ; **** \$parSet C12 Start Torque = 50% ; **** \$parSetCnt 3</pre>
<p>Trip Diagnostic Report Includes trip identification and status at time of trip. Includes: ambient (Ta); sink (Ts) and junction (Tj) temperatures; starter and motor thermal capacities (STC & MTC); and thermistor resistance (Rt).</p>	<pre>\$trip T30 MTR OVERLOAD \$tripElec 433V 50.0Hz 13.5A 9.5kW 75%Torq \$tripVolts 425V 435V 440V 2%lmb \$tripAmps 13.5A 12.1A 14.5A 8%lmb \$tripTherm 25C 42C 105C 15% 0% 320ohm ; /... Ta Ts Tj STC MTC Rt</pre>

Process Data Reporting:

Process data is periodically logged. The format is designed so that a log file can be parsed offline to produce trend graphs for customers.

Examples of process data reporting log entries:

```

13:56:50.00 $rptFmt 1.08
13:56:50.00 $rptOps Off Stby Rdy Kick Acc UTS Cont BP Dec Trip
22:50:20.67 $rptVars Op V Hz Ta Ts MTC Rt
22:50:20.67 $rpt O 433 50.0 23 40 75 282
22:50:50.67 $rpt O 428 49.9 25 38 80 278
22:51:20.67 $rpt O 430 50.1 24 35 87 276
23:05:10.50 $rptVars Op V Hz A kW Torq Vn Ta Ts Tj MTC Rt
23:05:10.50 $rpt K 433 50.0 90.4 60.8 153 300 23 37 77 95 250
23:05:10.51 $rpt K 432 50.0 91.2 61.4 155 300 24 39 79 94 255
23:05:10.52 $rpt A 431 50.1 46.8 31.7 76 300 24 42 72 93 258
23:05:10.53 $rpt A 433 50.1 48.4 32.4 78 280 24 44 74 92 256
23:05:20.00 $rpt U 435 50.0 58.4 38.2 91 0 23 47 77 71 252
23:05:20.01 $rpt U 434 50.0 59.5 39.9 92 0 24 47 77 70 252
23:05:30.25 $rpt B 432 49.9 58.4 38.5 88 0 24 47 77 55 255
23:05:40.25 $rpt B 436 49.9 59.7 39.1 89 0 23 46 76 54 259
23:15:30.61 $rpt D 432 50.0 55.7 35.4 80 0 24 38 66 45 263
23:15:30.62 $rpt D 430 50.0 50.2 32.7 77 0 24 39 64 44 265
23:15:35.70 $rptVars Op V Hz Ta Ts MTC Rt
23:15:35.70 $rpt T 430 50.0 23 40 0 282
23:16:05.70 $rpt T 428 49.9 25 38 5 278
    
```

Process Data Log Entries:

Token <Data>	Description
\$rptFmt #.##	Process data reporting format version (e.g. 1.08). Helps parser deal with different data logging formats (e.g. as reporting format matures).
\$rptOps <names>	List of operation names – informative (e.g. Off Kick Accel ... Trip). Provides an informative list of operation names (does not need to be parsed). It assists the human reader interpret the operation code. The first letter of each “operations name” (capital) corresponds to the single character (capital) “operation code” which is the first token after “\$rpt” (e.g. “\$rpt A ...” indicates the “Accel” operation is in progress).
\$rptVars <Ids>	List of variable Ids for subsequent data logs. The order and quantity of <Ids> correspond directly to the order and quantity of data values in subsequent data logs. The data reporter changes between long and short data logs. This entry is logged just before the changeover. The parser uses this to extract variable Ids for the graph legends (e.g. column titles) and to map data values correctly when data log length changes.
\$rpt <op> <values>	Operation code and list of data values (<op> = O, K, A ... T). The operation code <op> is a single character. This is followed by a list of numeric data values whose order and quantity correspond directly to previously given list of variable identifiers (\$rptVars entry). The data log entry begins with the standard time stamp which should be parsed for graph time values (x-axis). A single space character (0x20) separates each data value and the line length can vary depending on data values (e.g. “0 0.0 25 25 95 20” verses “433 50.0 25 101 1500”).

Process Variables:

The following table lists the supported process variables. The short data log contains a sub-set of these variables.

\$rptVars Id	Resolution & Units	Short Log?	Description
Op	Symbol	Y	Operation code (Op = O, K, A ... T) per \$rptOps
V	1 Volt	Y	Line supply voltage
Hz	0.1 Hertz	Y	Line supply frequency
A	0.1 Amp	-	Line supply current
kW	0.1 or 1 kW	-	Input power (resolution auto adjusts)
Torq	1 %	-	Output/mechanical power as % of motor rating
Vn	1 Volt	-	Notch voltage (back EMF). Used to detect motor up to speed.
Ta	1 °C	Y	Ambient temperature (-99 when DTS not fitted)
Ts	1 °C	Y	Sink temperature (i.e. base, heatsink or starter temperature)
Tj	1 °C	-	Junction temperature (as modelled)
MTC	1 %	Y	Motor Thermal Capacity (121% cold, 21% warm, 0% hot/trip)
Rt	1 ohm	Y	Resistance of motor thermistor (or thermostat cut off switch)

Process Data Log Length:

The long data log (all process variables) is logged whenever the motor is on and for 2 seconds after the motor turns off (stop or trip). Otherwise the short data log (subset) is logged.

Process Data Logging Rate:

During conduction regulation (Kick, Accel, Sync & Decel), data is logged at fastest rate (every 0.1s). In run mode (Cont|Bypass), data is logged every 10s. When the motor is off (Off & Trip), data is logged every 30s. However on changeover to run mode (Cont|Bypass) fast logging (0.1s) continues for 10s and when motor turns off (Off|Trip) fast logging (0.1s) continues for 2s.

List of Tokens:

Token <Data>	Description
\$brkPasses <info>	Diagnostic info logged on a breakaway trip (T6x MOTOR FAULT)
\$date <date>	Current date (format DD-MM-YYYY)
\$esoSecs <abuse times>	ESO starter trip override times (Is, Ts & Tj). Saturates at 9999s
\$ethGetDhcp <value>	DHCP on(1) / off(0) as read from Ethernet adapter
\$ethGetIpAddr <value>	IP address as read from Ethernet adapter
\$ethParDhcp <value>	DHCP on(1) / off(0) as configured by Network parameters
\$ethParIpAddr <value>	IP Address as per SS6K Network parameters
\$hrs <value>	Total operating hours. Can be cleared by clear meters command.
\$kWh <value>	Total energy used. Can be cleared by clear meters command.
\$locCfgAdj <id> <value>	Configuration setting adjusted by local user at console
\$locClkAdj <id> <value>	Date or time adjusted by local user (write to RTC)
\$maxEso <max values>	Max values during ESO operation (Is, Ta, Ts & Tj)
\$maxStd <max values>	Max values during normal (non-ESO) operation (Is, Ta, Ts & Tj)
\$mcuDown <down time>	Indicates MCU down time in seconds
\$mcuInfo <info>	Information on MCU (hex). Watch out for "MCU CHK"
\$mcuReset <code>	MCU reset code (1 = normal POR). Report others to Engineering.
\$mtrTol	Motor thermal overload diagnostics (recorded, adjusted & level)
\$netCfgAdj <id> <value>	Configuration setting adjusted by network client
\$netCfgErr <id> <value>	Configuration setting range error (raw numerical value given)
\$netClkAdj <id> <value>	Date or time adjusted by network client (write to RTC)
\$netClkAdjErr <info>	Error adjusting date or time (network client)
\$note <evtMsg>	Standard event (also see \$warn, \$strip and Appendix B)
\$parDef <id> <value>	Parameter ID and value (at the default value)
\$parSet <id> <value>	Parameter ID and value (set to a non-default value)
\$parSetCnt <cnt>	Count of number of parameters set to non-default value
\$product ... <model>	Zener Smartstart <model>[*]. Note 1.
\$rpt <op> <values>	Log of operation code and process data values.
\$rptFmt <verNum>	Process data format version (e.g. 1.07) to assist offline parsing
\$rptOps <names>	Process data operation names (first letter per operation code)
\$rptVars <ids>	Process data variable IDs (e.g. parse for column titles)
\$ssTrips <counts>	Soft starter trips (Is=T20; Ts=T21/22; Tj=T23). Saturates at 9999.
\$starts <count>	Total starts. Can be cleared by clear counters command.
\$strip <id> <msg>	Trip event (also see \$note, \$warn and Appendix B)
\$stripAmps <status>	Trip diagnostic report (3 phases of current plus imbalance)
\$stripElec <status>	Trip diagnostic report (electrical summary)
\$strips <count>	Total trips. Can be cleared by clear counters command.
\$stripTherm <status>	Trip diagnostic report (thermal summary)
\$stripVolts <status>	Trip status report (3 phases of voltage plus imbalance)
\$version <verId>	Firmware version ID (e.g. V1.02A)
\$warn <evtMsg>	Warning event (also see \$note, \$strip and Appendix B)

Notes:

1. A trailing '*' on \$product indicates a customised model (e.g. "Zener Smartstart 6R100*")
2. "Is" refers to starter current (% of starter rating) or T20 trip
3. "Ta", "Ts" & "Tj" refer temperatures (degrees C): ambient, sink (T21/22) & junction (T23)
4. Any non-zero \$esoSecs value indicates starter abused (trip overridden during ESO)

List of Events:

Event Message	Description
ACCEL KICK VOLT TORQ #s	Acceleration initialisation (type and kick/ramp time)
ACCEL KICK DONE	Acceleration progress (kick done, start ramp up)
ACCEL AT SPEED	Acceleration progress (low back EMF detected)
ACCEL OVERTIME	Acceleration progress (ramp timer expired before REG DONE)
ACCEL REG DONE	Acceleration progress (at max regulation, go full conduction)
ACCEL TIME DONE	Acceleration progress (ramp timer expired after REG DONE)
ACCEL DONE	Acceleration finished
AR CLEARING #<counter>	AR counter decremented by clearing timer
AR LOCKOUT #<counter>	AR Lockout on final trip
AR PENDING #<counter>	AR counter incremented on trip event (AR now pending).
AR TRIP RESET #<counter>	Successful automatic trip reset (may trigger motor restart)
BP FAULT	Tripped on bypass contactor fault (also see \$trip T14/T15)
BP ON OFF	Internal bypass contactor turned ON/OFF (also see RL2)
BP READY	Bypass operation complete (timed, e.g. 200ms later).
BRK G# ON OFF	Breakaway gate firing event (# = 1..3) – first one per gate only.
BRK NCA #.##%	Breakaway init. Non-Conduction Angle in % (= us @ 50Hz)
CFG CHK START DONE	Configuration check status (e.g. after menu exit & motor off)
CFG ERR: <cause>	Configuration error. Report <cause> to Engineering.
CMD LOC ONLY <info>	Motor control command source changeover (local / network)
CMD START STOP COAST	Motor control command
CLK <info>	Clock events
CT# ERR: <phase>	CT phasing error (# = 1..2, <phase> = +/- Ln for n = 1..3)
CTs OK	CTs passed phasing checks
CURR LIM ON OFF	Current limiting activated/deactivated
CURR ON OFF	Current sense detect
CURR LOSS	Current loss while running (also see T13 MOTOR LOSS)
D# ON OFF	Programmable digital input turned ON/OFF (# = 1..2). Note 4.
DECEL ABORT	Stop command before up to speed (coast to stop)
DECEL COAST VOLT TORQ #s	Deceleration initialisation (type and ramp time)
DECEL TORQ RELEASE	Deceleration progress (release torque reached)
DECEL REG DONE	Deceleration progress (at min regulation, turn off gates)
DECEL TIME DONE	Deceleration progress (ramp timer expired)
DECEL DONE	Deceleration finished
EN ON OFF	ENABLE digital input turned ON/OFF
ERASE START ABORT DONE #	NVM erase for persistent data (# = number of records)
ESO ON OFF	Essential Services Override status
ETH <info>	Ethernet port boot status events (advanced)
FAN ON OFF	Internal FAN turned ON/OFF
GATES OFF RUN RUN/LP	Stop/start all gate pulses with low power option (RUN/LP).
LINE ON OFF	Line supply (3 phase) energised / de-energised
LINE 123 321	Line supply phase sequence detected (123 = +ve, 321 = -ve)
LINE LOSS	Line supply loss while running
LOC <info>	Local commands (motor start/stop/coast, etc.)
MCU CHK [code]	Report code and \$mcuInfo to Engineering
MCU START	MCU came out of reset (Note 1). Also see \$resetCode
MTR CHK OK ERR: <info>	Motor wiring check pass/fail (info = detected motor wiring)
NET <info>	Network events (open/close, cmd change/timeout, etc.)
NET CONF <info>	Specific configuration via network events (see V1.30 notes)
PSU DIP DIP/ISR	PSU (24V) UV detected. Should always be "/ISR" (Note 3)
PSU OFF	Periodic (10ms) report of PSU UV until MCU resets
PSU OK	PSU start up or recovery (with 100ms on delay filter)
PWR MOTOR REGEN OFF	Detect power direction change (motoring or regenerating)

RL# ON OFF	Programmable relay turned ON/OFF (# = 1..4). Note 4
RST CHK	Unusual reset code. Report \$mcuReset to Engineering.
RTC ERR	Real Time Clock problem. Check battery.
R_T HOT COOL	Thermistor input status change (HOT when R_T > ~1K6)
SAVE START DONE #	Save of persistent data to NVM (# = number of records)
T<code> <message>	Trip code and message (see product manual)
TEST <info>	Special board test modes (but can send junk to logger!)
TEST BUILD	Special test build NOT FOR RELEASE. Report to Engineering.
TRIP RESET: <method>	Request to reset trip. Various methods available
USR IS LOCAL REMOTE IDLE	User keypad activity (REMOTE = smart console in use)
USR CMD <info>	User command from "Commands" menu
USR LOGIN <info>	User login (pass/fail & access level)
USR LOGOUT <info>	User logout (show logout/default access level)
USR MENU ENTER EXIT	User enters or exits the menu system

Notes:

1. Some events happen before the Data Logger Option Card has booted and therefore will not be seen in the log file. These events can be logged using terminal software (requires a TTL/RS232 converter).
2. Some boot up log entries are deliberately delayed for 2 seconds to allow sufficient time for the Data Logger Option Card to boot up.
3. The system is designed to call an Interrupt Service Routine (ISR) on PSU DIP. This is backed up by polling the UV input.
4. Check parameter log (on boot up) to determine how programmable inputs (D#) and programmable relays (RL#) are configured.