

# Field Trouble-Shooting Manual for PowerDash Data Loggers

## Basic Troubleshooting

### *Tools required*

- Digital volt meter (DVM)
- Analog telephone (for phone connected systems)

### *Where Useful*

- Logger not reporting to server
- No production from KYZ Channel

### *Verify 5V DC power present*

- Use DC voltage tester, probe VCC to GROUND on logger terminal block (should be 5V DC).
- Check for LEDs on the TStik board during startup.

### *Verify communication channel*

- If this is an Ethernet-connected system, verify that the Ethernet connection is “live”. When the datalogger is powered on, there should be activity on the 3 LEDs near the top center of the processor card when an Ethernet cable is attached:



- If it is a phone-connected system, unplug incoming phone line from the logger, plug it into a phone and check for dial tone.

***NOTE:** By default, the logger checks for active Ethernet immediately after it is reset, and uses this information to set whether or not it is in dialup mode or ethernet mode. An active Ethernet should be attached BEFORE the logger is powered up. If desired, the `logrcfg` utility can be used to force the logger into a specific mode.*

**. Check KYZ connection and Verify incoming pulses from kWh meter (if used, and no production is registering at PowerDash)**

- NOTE: A KYZ connection CANNOT affect whether or not the logger reports to the server, only the presence or absence of data.
- A single KYZ meter should be connected with the Red wire to GND, Black to COUNT1, and (optionally) Yellow to COUNT2. A second meter should be connected to COUNT 3 and (optionally) COUNT 4. If 3 or 4 KYZ meters are used, connect only the Red and Black wires to COUNT1, 2, 3, 4.
- Assuming that a KYZ meter is connected to COUNT1, 2, 3, or 4, measure the voltage from the COUNT input to GND, and verify that it changes from 0 to 5 Volts and back to 0 (see Note).

*Note: The rate of this change is a function of the pulse constant of the meter and the power through the meter. When PowerDash provides KYZ meters, the “pulse constant” of the meter is set to the same as the meter's Kh, printed on the meter face. For example, if the meter has a Kh of 7.2, one “pulse” represents 7.2 watt hours of electric energy. A “pulse” is a switch opening or closure. So, if 1 kW is flowing through a meter with a Kh of 7.2, there should be  $1000/7.2 = 139$  pulses per hour. The voltage at the COUNT input should stay high or low for 26 seconds before changing to the other state.*

**. Reboot the logger**

- Reboot the logger by unplugging its power supply and plugging it back in again. Upload attempt should begin within 3-5 minutes.
- Use the Logger Reporting Status in the PowerDash support pages to check if the logger has contacted the server.

## Using the Built-in LED Blinker Codes

### . **Tools Required**

- None
- This involves reading and interpreting the status codes presented by the LED mounted at the upper left corner of the Tstik processor card, which are always active:



### . **Where Useful**

- If the logger is not reporting to the server.

### . **Reading the Status Codes**

- The logger blinks a **series** of status codes that provide a great deal of information about the logger's operation.
- Each status code is 2 digits.
- Each digit can be a number from 0 to 9
- A zero is a single long blink of the LED (represented below as “-”).
- Non-zero digits are each a series of quick blinks of the LED (shown below as, for example, “...” (for the digit 3)).
- The 2-digit codes are separated by a slight pause.
- The sequence of codes is followed by a longer pause.
- Don't blink or you'll miss it!
- It may take a little practice to read these.

## . **Interpreting the Status Codes**

- The status codes are grouped by the basic functional area that they represent.
- Basic logger health check:
  - **0 0 (- -)** - This code represents the most basic level of logger health check. It indicates that the main firmware loop is functioning. This code should be seen within a couple of minutes of power up, and should be a part of every status blinking sequence as long as the logger is operating.
- Communication Channel Indicator (exactly one of this group should flash)
  - **0 1 (- .)** - The logger will be attempting to communicate in dialup mode.
  - **0 2 (- ..)** - The logger will attempt to communicate via Ethernet, and has successfully obtained an IP address from the DHCP server.
  - **0 3 (- ...)** - This logger will attempt to communicate via Ethernet, but has **FAILED** to obtain an IP address from DHCP. This is a fatal error, and the logger will be unable to contact the server.
  - **0 4 (- ....)** - The logger will attempt to communicate via Ethernet, and is using a Static IP address.
  - **0 5 (- .....)** - The logger will attempt to communicate via GPRS modem.
- Ethernet Communication Status – multiple error codes can be shown. All error codes clear after a successful server transaction.
  - **1 0 (. -)** The last attempt to communicate with the server was successful.
  - **1 1 (. .)** **ERROR:** The logger's last attempt to communicate failed.
  - **1 2 (. ..)** **ERROR:** The logger's last communication attempt had a DNS error
  - **1 3 (. ...)** **ERROR:** Problem occurred sending logger configuration
  - **1 4 (. ....)** **ERROR:** Problem occurred sending channel configuration
  - **1 5 (. ....)** **ERROR:** Problem occurred sending logger status
  - **1 6 (. .....)** **ERROR:** Problem occurred sending data
  - **1 7 (. ....)** **ERROR:** Bad XML returned from server.
  - **1 8 (. ....)** **ERROR:** Server indicated error.
  - **1 9 (. ....)** **ERROR:** Bad URL. Could indicate a misconfiguration of the datalogger.
- Phone Communication Status (most codes also apply to GPRS)
  - **2 0 (. . -)** **ERROR** opening serial port.
  - **2 1 (. . .)** **ERROR:** no dial tone.
  - **2 2 (. . ..)** **ERROR:** Phone line busy.
  - **2 3 (. . ...)** **ERROR:** Connection timed out.
  - **2 4 (. . ....)** **ERROR:** Authorization refused.
  - **2 5 (. . ....)** **ERROR:** PPP Connection rejected.
  - **2 6 (. . ....)** **ERROR:** Modem did not respond.
  - **2 7 (. . ....)** **ERROR:** Other PPP Error.

## . **What to do with status codes**

- In most cases the status code will point to a local problem. If not, seek help from PowerDash.

. **Example**

● - - - . . . - - - . -

- This is a fairly typical sequence, shown twice. It says that the logger's main loop is operating (0 0) in Ethernet mode, with an IP address obtained from DHCP (0 2), and that the last communication with the server completed normally (1 0).

## Other Troubleshooting Possibilities

### . **Tools required:**

- Laptop computer configured with Logger Support tools
- Logger connected to Ethernet network or phone
- Ethernet crossover cable
- Serial Null Modem (DB9 male to DB9 male, at least 6 foot recommended), or standard serial cable with Null Modem adapter, or a verified functioning USB/serial adapter, with Null Modem adapter.

### . **Configure field laptop with Logger Support tools**

- Download the file loggersupport.zip from the Support area at [www.PowerDash.com](http://www.PowerDash.com).
- Unpack the contents of loggersupport.zip to a new directory on your field laptop's hard drive (recommend c:\loggersupport\).
- Find the file javakit.bat in c:\loggersupport\. In Windows, right-click on the file and choose "Send To," "Desktop (create shortcut)."

### . **Use the "findloggers" utility**

This utility does a UDP broadcast command over the network that is recognized by a very simple server running in each logger. Each logger on the LAN responds with its IP address and its Ethernet MAC address. For example, from a command line:

```
cd c:\loggersupport
findloggers
```

```
IP Addr: 192.168.0.41 Response: 00:60:35:00:94:e1
IP Addr: 192.168.0.8 Response: 00:60:35:00:d1:4f
IP Addr: 192.168.0.9 Response: 00:60:35:00:ce:97
IP Addr: 192.168.0.6 Response: 00:60:35:00:c6:65
IP Addr: 192.168.0.2 Response: 00:60:35:01:08:52
```

Shows a sequence captured from a LAN that had 5 installed dataloggers.

The utility can also be used to capture the configuration of a specific logger, if the unit's MAC address is passed as a command line parameter:

```
findloggers 00:60:35:01:08:52
```

This will return an XML string that describes the logger's configuration (not the device configuration, though).

You can also invoke findloggers by double clicking on the findloggers.bat icon from Windows Explorer.

findloggers also will work on a dialup system. The PC should be connected directly to the logger using an Ethernet crossover cable. It will probably return an IP address of 0.0.0.0, but it can be comforting to show that the logger is alive.

*Note: If using findloggers to directly communicate using a crossover cable, it may be useful to set a static IP on your PC.*

### **Use the “logrcfg” utility**

“logrcfg” incorporates the “findloggers” capability, but can also be used to customize the function of a datalogger.

This is a Java program that can be used to configure a datalogger without using the server-based configuration management. **logrcfg** has been run on Windows and on Linux. It communicates with the logger over the Ethernet.

There are certain operations for which **logrcfg** is recommended:

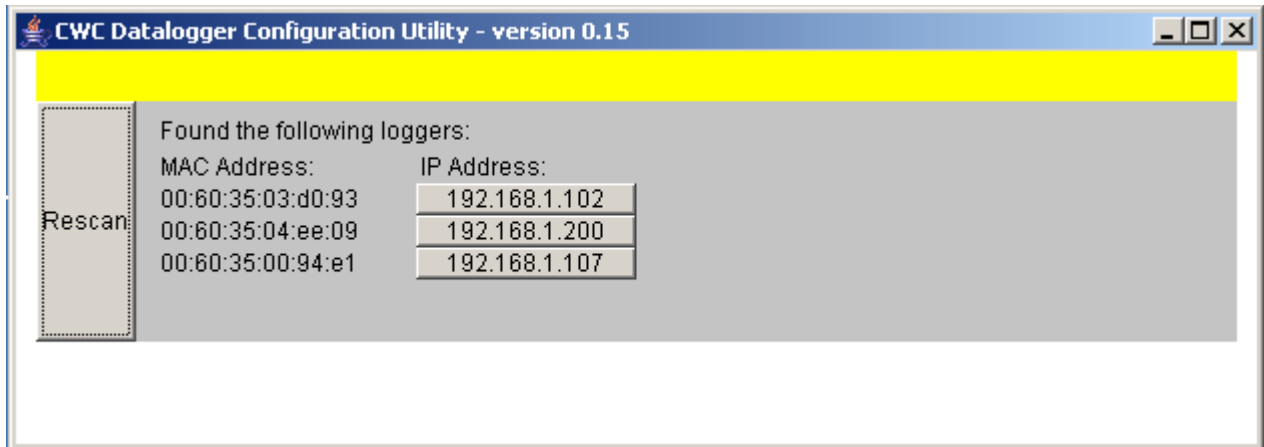
- Setting static IP configuration
- Setting Proxy server information
- Setting information for dialup operation
- Forcing Ethernet Only or Dialup Only mode
- Committing configuration to Flash memory, so that it will persist across all types of resets.

**logrcfg** is invoked from the **c:\loggersupport** directory by executing the **logrcfg.bat** file from a dos command line:

```
cd c:\loggersupport
logrcfg
```

It can also be invoked by double clicking on the logrcfg.bat icon from Windows Explorer.

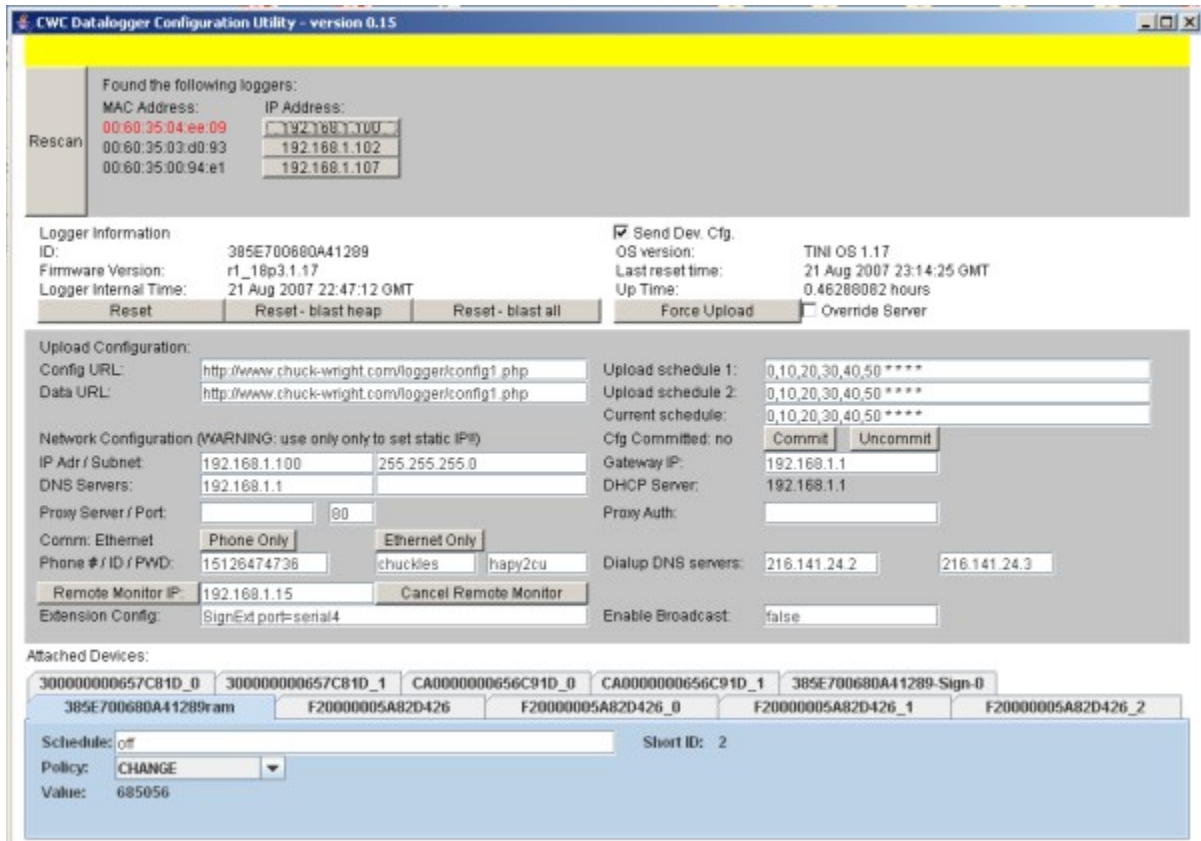
When invoked, logrcfg uses UDP packets to query the local network for dataloggers, just like "findloggers". It opens a window that displays a button for each datalogger that it locates:



The "rescan" button causes the utility to rescan the network for loggers.

If just one datalogger is detected, then its configuration will be accessed and displayed. Otherwise, the configuration for the desired datalogger can be obtained by clicking the

button for the desired datalogger. The screen that is displayed after accessing the datalogger looks like the following:



The yellow bar across the top is used as a status window.

The second horizontal band displays the rescan button and a button for each detected logger. The IP and MAC addresses of the currently displayed logger are highlighted in red. If a window appears in the right portion of this band, it is used as a remote console from the logger, displaying status messages.

The third horizontal band displays logger information that cannot be changed. Across the bottom of this band are some buttons and a checkbox:

- "Reset" - A soft reset is performed. The datalogger's RAM memory content is retained, meaning that all configuration is retained, as well as the content of the data queues.
- "Reset - Blast Heap" - A reset is performed, and the RAM memory is cleared, but the communication configuration is retained. All other logger configuration and data are lost. Also, if the datalogger is executing firmware from RAM, this will be lost and the datalogger will revert to whatever level of firmware is stored in the flash memory.
- "Reset - Blast All" - Like above, but in addition, all communication setup (such as static IP) is lost (except if committed to flash - see below).

- "Force Upload" - If this is clicked, then the logger will be forced to contact the server at the next 1-minute interval boundary, instead of waiting for its programmed schedule.
- If the "override" box is checked, then the next server contact sequence will be accompanied by an XML tag that requests that the configuration presented by the datalogger should override what is in the server's database.

If one of the "Reset" buttons is clicked, there will be no refresh of the configuration for that logger. After a period of a minute or two, the "Rescan" button may be clicked, and the logger should reappear.

The fourth band displays programmable configuration elements in the logger:

- The first group "Upload Configuration", shows the current setting for the URLs that the logger contacts, and the schedule that is being used.
- The next group, "Network Configuration", shows the current network configuration settings.

To set a static IP address, first enter the subnet mask, the Gateway IP and the Primary and Secondary Domain Name Servers. **Then** change the IP address and hit "enter". The configuration display will NOT be refreshed, because the datalogger is busy recycling its servers using the new network configuration. After a minute or so, you can click the "rescan" button, and the datalogger, with its new IP address, should be rediscovered. The configuration should reflect the static IP address that you set, plus the other network parameters that you entered, and the field that normally shows the DHCP Server IP will be blank.

A static network configuration will persist across most reset events, including "blast heap". However, it will not persist across a "blast all". To make it completely safe, click the "Commit" button. This will save it into flash memory.

**NOTE:** To remove a static IP, you must first uncommit the configuration from flash (see below). Then a "Reset - blast all" should be done. After a minute or so, "Rescan" and the logger should be rediscovered, and it will be in DHCP mode.

Certain configuration parameters can be committed to flash memory so that they persist across all kinds of resets. Whether or not values are currently committed to flash is indicated by the "Cfg Committed: yes" (or "no"), displayed just before the "Commit and Uncommit" buttons. Here is how to use these buttons:

1. Prior to firmware version 1.17, only the basic network configuration (IP address, subnet, gateway, and DNS) was saved using the "commit" function, and it was saved in the same flash memory page as firmware. The committed settings could only be removed when new firmware was programmed into flash memory. The "uncommit" function does not work on these versions.
2. As of version 1.17, the commit function saves much more information, and utilizes a different flash memory sector, so it can be uncommitted and recommitted independently of firmware loading. For these versions, the flash configuration includes the following settings:

1. The default URLs to which the unit connects for configuration and data exchanges.
2. IP address, Subnet mask, Gateway, DNS1, DNS2 (if static IP active)
3. Proxy Server IP and port, Proxy authorization string.
4. Dialup ID, password, DNS1 and DNS2, and whether dialup is being forced or prevented.

These values override the defaults that are stored in firmware, and will affect where the logger reports, and which phone number it uses, after any reset, soft or hard, including when the heap is blasted. The only way to remove these is to "uncommit", which will make the unit revert to values stored in firmware.

- The following group, "Dialup Configuration", gives the parameters that are used for dialup operation.
- The button marked "Network Monitor IP", if clicked, puts the IP address of the computer you are using into the following box. This causes a remote monitoring session to be started, with the datalogger's console messages displayed in the box in the upper right of the window. Clicking on the cancel button on the right causes the remote monitoring to be halted and the box cleared. NOTE: Some datalogger firmware builds do not provide this capability, and these buttons and the box will not be displayed.

As of firmware version 1.17, the unit can be forced into "Ethernet only" mode. As of version 1.18p3, it can also be forced into "Dialup only" mode. If it is already in either of these modes, **logrcfg** displays a "Release" button displayed that will disable these modes. If neither is in effect, there are 2 buttons, one for "Ethernet only" and one for "Phone only". Note that these modes persist across a soft reset or power recycle, but not across a "blast heap". To make them "permanent" follow with a "Commit" operation. They will then persist until an "Uncommit" is done.

- The buttons dealing with "Remote Monitor" are not yet functional.
- The field marked "Extension Config" is the control string for enabling code extensions in the logger, and corresponds exactly to the field in the server's configuration management screen.
- The field labeled "Enable Broadcast", if "true", causes the unit to broadcast, using UDP, an XML string giving the current values of all channels, once per minute. This field corresponds to the field in the server's configuration manager.

**WARNING: Please remember that many of the fields correspond to fields in the server's configuration manager. Unless the "Override Server" box is checked, the value that is in the server's configuration will override whatever is set locally by this utility. This can get very confusing!!**

The area at the bottom of the window displays configuration information for all devices that have been sensed on by the datalogger.

*Note: If using findloggers to directly communicate using a crossover cable, it may be useful to set a static IP on your PC.*

## **Use Telnet to View Console Stream**

If you can determine the IP address of the datalogger (say, by looking at the admin screens on your router), you can use **telnet** to view the logger's console stream:

```
telnet <ip_addr> 58324
```

The logger should respond with

```
Hi from Logger!!
```

Then start spewing stuff like

```
Mon Dec 03 02:59:00 GMT 2007
```

```
Q: 0 S: 1 T: 6
```

```
192.168.1.107
```

```
upload XML
```

```
Config->
```

```
setting broadcast: false
```

```
set broadcast to false
```

```
Return: OK
```

```
Status->
```

```
Return: OK
```

```
Data->
```

```
Return: OK
```

```
STATMSG: 300
```

```
End upload
```

You should be able to watch it go through its paces, and it will complain if it is upset about anything.

To close the session, type “q”.

## **. Use the logger's built-in HTTP server**

Loggers running firmware version 0.84 or later have a built in http server that runs regardless of whether the logger is connected by phone or Ethernet. The server can be used as an aid in debug and configuration. The server can be accessed from a computer on the same LAN as the logger. To access, follow these steps:

- Connect an Ethernet cable from the network to the logger.
- Restart the logger and determine its IP address (section 6, 7, & 8, or with JavaKit)
- Open a web browser and type the following address (syntax described below):

```
http://<ip addr>/0 (or 1, 2, 3, 4, 5, 6)
```

```
0 - return config XML (Logger AND Device)
```

```
1 - (same)
```

```
2 - return data queue status XML
```

```
3 - return data queue XML
```

```
4 - return status queue status XML
```

```
5 - return status queue XML
```

```
6 - return network parameters as XML
```

*WARNING: This is NOT a production-quality HTTP server. It is suitable for debug use only!*

## JavaKit

JavaKit is used to initially load firmware and operating system. Once in the field, it is not generally to be used except in unusual circumstances.

### . **Start JavaKit and connect to the logger.**

- Connect serial cable between PC serial port and Logger “PORTA”.
- Plug in the logger.
- Double-click on the javakit.bat desktop shortcut (a DOS window and JavaKit will open, see Figure 3).
- Under Port Name at the bottom left, select a COM port (normally COM 1 unless it is being used by another application).
- Click the “Open Port” button.
- Click the “Reset” button.
- The logger should respond with a command prompt:

```
TINI loader 05-15-00 17:45
Copyright (C) 2000 Dallas Semiconductor. All rights
reserved.
>
```

### . **Use JavaKit to initiate a “hard” or “soft” reboot**

- a. Open JavaKit and establish a connection to the logger.
- b. **Soft reboot.** For a soft reboot, press “E” then [Enter]. JavaKit will then display the status messages associated with the reboot operation. A soft reboot is equivalent to unplugging the logger and plugging it back in again, all previously-collected data is retained).
- c. **Hard reboot.** For a hard reboot, press:
  - i. B0 [Enter]
  - ii. F0 [Enter]
  - iii. E [Enter]

***Warning! A hard reboot permanently erases previously collected data and reformats the logger’s memory structures. It does not reset configuration information that has been committed to Flash.***

### . **Use JavaKit to load a new Operating System or Firmware**

- a. Open JavaKit and establish a connection to the logger.
- b. **Operating System.** The logger runs on a version of the TINI Operating System (OS). To reload or update the OS, go to the JavaKit menu, choose “File,” “Load,” and select the OS file to load. Choose the OS file from the loggersupport/firmware directory on your hard drive. (The current OS version is tini\_400\_1.17.tbin.)
- c. **Firmware.** The process for uploading firmware to the logger is the same as that for uploading the OS. On the JavaKit menu, choose “File,” “Load,” and select the firmware file to upload. Choose the firmware file from the

loggersupport/firmware directory on your hard drive. Current firmware versions may be obtained from the Support area of the PowerDash website ([www.PowerDash.com](http://www.PowerDash.com)).