

# SURETECH Load Logger



## Introduction

The SURETECH Load Logger System is a Long Term Data Logging System.

The System is capable of collecting pre-processed data, by summarising the Voltage, Current & Phase Angle for multiple RS232 Lines, which are monitored by SURETECH Smart Load Transducers over a period of up to 50 days.

The data collected is accumulated, averaged, maximised & minimised representation per minute, of 60 samples taken in a minute, this is done to avoid extreme amounts of data and a friendlier data transportation system.

The complete SURETECH Load Logger System (SLLS) includes 1-4 SURETECH Smart Load Transducers (SLT), 1 SURETECH Port Splitter MUX (PSMUX), Standalone DOS PC running the SURETECH Load Logger Software (LLS).

### **Hardware Requirements**

The SLLS requires a PC with at least 1 serial port, 1 parallel port, 40mb Hard Drive, 1Mb RAM, 1 Monochrome Monitor and a 286 Processor to run the LLS.

Additional hardware to the system includes the PSMUX & SLT's

### **SURETECH Multiplier / Accumulator (MAC)**

## **Software Capabilities**

SURETECH Load Logger Software Text User Interface a Screen capture of the TUI is shown above:

TITLE            Name of System Running, thus SLLS  
TIMING          Timing Display - Process Cycle Indicator  
DATA            Data Display Area - Present User with Latest Results

### **DATA analysis:**

Rows Represent SLTs  
Columns represent values measured or calculated.

1 Second Readings

Vinst Instantaneous Representation of unscaled Voltage.  
Iinst Instantaneous Representation of unscaled Current.  
Ainst Instantaneous Representation of unscaled Phase Angle.

### **1 Minute Averages, Min, Max**

Vavg: Unscaled average representation of fullscale Voltage  
 $SUM(60 \text{ inst SAMPLES})/2.45701$   
Iavg: Unscaled average representation of fullscale Current  
 $SUM(60 \text{ inst SAMPLES})/2.45701$   
Aavg: Unscaled average representation of fullscale Phase Angle  
 $SUM(60 \text{ inst SAMPLES})/2.45701$   
Vmin: Minimum Value Encountered while calculating Vavg  
Imin: Minimum Value Encountered while calculating Iavg  
Amin: Minimum Value Encountered while calculating Aavg  
Vmax: Maximum Value Encountered while calculating Vavg  
Imax: Maximum Value Encountered while calculating Iavg  
Amax: Maximum Value Encountered while calculating Aavg

ERRORS : Space to Indicate Error Flag Conditions (Clear If No Errors)

### **Flags & Their Meanings**

INV\_FILE\_STAMP  
The timestamp on the data files do not match or is an invalid format.  
DATx\_CLOSE\_ERR  
An error has occurred while trying to CLOSE the relevant DATAx file.  
DATAx\_OPEN\_ERR  
An error has occurred while trying to OPEN the relevant DATAx file  
T\_O\_SIGNAL\_ERR  
(disabled) Indicates that there is no incoming signal on one of the SLT's  
HEXCHTOINT\_ERR  
(disabled) Indicates that a non-hexadecimal char was recieved.

## **DataFile Analysis**

A sample DataFile can look like this:

```
962296213 29/06/2000 12:30:13          }TimeStamp
SeqNu Vavg Iavg Aavg Vmin Imin Amin Vmax Imax Amax  }Header
00001 xxxxx xxxxx xxxx xxxx xxxx xxxx xxxx xxxx }DATA
00002 xxxxx xxxxx xxxx xxxx xxxx xxxx xxxx xxxx }DATA
00003 xxxxx xxxxx xxxx xxxx xxxx xxxx xxxx xxxx }DATA
00004 xxxxx xxxxx xxxx xxxx xxxx xxxx xxxx xxxx }DATA
00005 xxxxx xxxxx xxxx xxxx xxxx xxxx xxxx xxxx }DATA
00006 xxxxx xxxxx xxxx xxxx xxxx xxxx xxxx xxxx }DATA
962296603 29/06/2000 12:36:43          }TimeStamp
SeqNu Vavg Iavg Aavg Vmin Imin Amin Vmax Imax Amax  }Header
```

### **TimeStamp**

See TimeStamping Section

### **Header**

Same as in TUI Section except for SeqNu which is discussed in timestamp.

### **DATA**

Same as in TUI Section except for SeqNu which is discussed in timestamp.

### **TimeStamping**

TimeStamping on a file occur at only two occasions namely:

When user starts the program

When user ends the program

The first timestamp is the base value from which all sequence numbers are calculated from. The first value in the timestamp is a representation of the time elapsed in seconds since 0h00 1 Jan 1970, the second string is the date in a DD/MM/YYYY configuration and the last string the time.

The sequence number is calculated by the amount of minutes passed since start of execution of the LLS. The reason for all this is to allow the program to be interrupted by a power failure and yet not lose its timing. It also is a very compact way of presenting the exact time, thus saving in hard drive space.

The timestamp at the end of the file indicates when the program was stopped and also function as a correlation & data integrity check.

## **Software Features**

The LLS will log up to 4 RS232 channels by driving the PSMUX.

The LLS reads the serial port on a polling and timing system.

Interrupts is used (timer tick) to detect timed out signals.

The Software Sets up the serial port to:

- 1 Start bit
- 1 Stop bit
- 8 Data bits
- no parity
- FIFO buffers disabled
- DTR & RTS permanently high

The parallel port is used as a standard parallel port. Data is read in streams of characters with a new line and "STA" forming the start sequence and a return carriage end character.

A delay in between the transmitting of Data Stream will allow the LLS to auto-synch incoming signals.

## **Error Messages**

The file LOADLOG.ERR will be created in the event of a error. All errors will be logged with a realtime stamp to indicate when they happend + a detailed description of the error. If no errors occurred, no file will be created.

## **File Generation & File List**

Data files will only be created anew if the "DATA1.TXT" file is not existing in the current directory. Files is opened for appending and reading, thus no data is ever lost. If files were left in directory after last execution, the LLS will simply append to the end of the data files.

### **File List**

LOADLOG1.EXE : The Executable file that loads the LLS  
README.TXT : This File  
DATA1.TXT : Data File for STL1  
DATA2.TXT : Data File for STL2  
DATA3.TXT : Data File for STL3  
DATA4.TXT : Data File for STL4  
  
LOADLOG.ERR : Error Log File