



HL7900 Standard Debugging Guide

HL7900 Series

Reference	Date	Updated	Version
2174383	Wed Oct 22 16:21:26 2025 +0000	Mon Oct 20 16:59:54 2025 +0800	2174383_HL7900_Standard_Debugging_Guide_r1.4

List of Contents

List of Contents	1
Legal Notice	2
Revision History	3
1 Introduction	4
2 Download Location	4
3 Connect the Host to the Development Kit	4
4 Enable/Disable Firmware traces	5
4.1 Enable Firmware traces with filter	5
4.2 Disable Firmware traces	6
4.3 Examples of Firmware trace filters	6
5 Capturing Firmware traces by STT	7
5.1 Introduction	7
5.2 Capturing traces	7
5.2.1 Usage menu of stt	7
5.2.2 Steps to capture traces	7
6 Getting backtraces	8
6.1 Display and erase backtraces	8
6.2 Upload backtraces to FTP server	8
7 Reference Documents	9
END OF DOCUMENT	9

Legal Notice

Important Notice

Information relating to this product and the application or design described herein is believed to be reliable, however such information is provided as a guide only and Semtech assumes no liability for any errors in this document, or for the application or design described herein. Semtech reserves the right to make changes to the product or this document at any time without notice. Buyers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. Semtech warrants performance of its products to the specifications applicable at the time of sale, and all sales are made in accordance with Semtech’s standard terms and conditions of sale.

SEMTECH PRODUCTS ARE NOT DESIGNED, INTENDED, AUTHORIZED OR WARRANTED TO BE SUITABLE FOR USE IN LIFE-SUPPORT APPLICATIONS, DEVICES OR SYSTEMS, OR IN NUCLEAR APPLICATIONS IN WHICH THE FAILURE COULD BE REASONABLY EXPECTED TO RESULT IN PERSONAL INJURY, LOSS OF LIFE OR SEVERE PROPERTY OR ENVIRONMENTAL DAMAGE. INCLUSION OF SEMTECH PRODUCTS IN SUCH APPLICATIONS IS UNDERSTOOD TO BE UNDERTAKEN SOLELY AT THE CUSTOMER’S OWN RISK. Should a customer purchase or use Semtech products for any such unauthorized application, the customer shall indemnify and hold Semtech and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs damages and attorney fees which could arise.

The Semtech name and logo are registered trademarks of the Semtech Corporation. All other trademarks and trade names mentioned may be marks and names of Semtech or their respective companies. Semtech reserves the right to make changes to, or discontinue any products described in this document without further notice. Semtech makes no warranty, representation or guarantee, express or implied, regarding the suitability of its products for any particular purpose. All rights reserved.

Wireless Communications

Due to the nature of wireless communications, transmission and reception of data can never be guaranteed. Data may be delayed, corrupted (i.e., have errors) or be totally lost. The Semtech product should not be used in situations where failure to transmit or receive data could result in damage of any kind to the user or any other party, including but not limited to personal injury, death, or loss of property. Semtech accepts no responsibility for damages of any kind resulting from delays or errors in data transmitted or received using the Semtech product, or for failure of the Semtech product to transmit or receive such data.

Safety

Do not operate the Semtech product in areas where blasting is in progress, where explosive atmospheres may be present, near medical equipment, near life support equipment, or near any equipment which may be susceptible to any form of radio interference. In such areas, the Semtech product should be powered off.

Sierra Wireless

Semtech Corporation purchased Sierra Wireless in January 2023. The Sierra Wireless brand is gradually being phased out. During the phase-out period, references to both *Semtech* and *Sierra Wireless* may appear in product documentation.

Qualcomm Licenses

Semtech’s cellular modules are sold subject to certain notices and restrictions regarding patent licenses from Qualcomm Incorporated. These notices and restrictions are available at www.sierrawireless.com/qualcomm-notices.

Copyright © 2025 Semtech Corporation. All rights reserved.

Contact Information

Sales information and technical support, including warranty and returns	Web: https://sierrawireless.com/company/contact-us/ Global toll-free number: 1-877-687-7795 6:00 am to 5:00 pm PST
Corporate and product information	Web: Sierra Web

Revision History

Revision	Date	Changes
1.4	Oct 20, 2025	Minor rewording Use HL7900 in place of HL79XX throughout the documentation Correct AT%LOGCMD and Add 'host connection' section
1.3	Aug 21, 2025	Add a note about IP layer logging command
1.2	Jul 28, 2025	Correct %SETSRMCFG's DBGUARTMODE disable value to 2 Remove spacing in table header definition
1.1	Jun 10, 2025	Rephrase wordings for readability Hide instruction of building STT from source
1.0	Apr 23, 2025	Initial creation

1 Introduction

This guide gives high-level instructions for basic debugging of HL7900 modules running Semtech modem firmware. It covers the collection of trace logs and crash backtraces, which are used by Semtech personnel for analysis and debugging.

2 Download Location

- The tools and external documents mentioned in this document can be downloaded from [The Source](#).

3 Connect the Host to the Development Kit

Note:

Below steps are relevant to the installation of the HL7900 module to the Development Kit board. For other configurations, the steps are substantially the same.

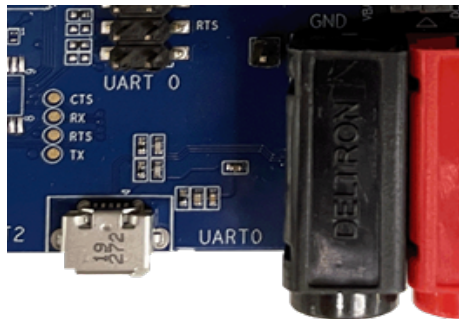


Figure 1: USB-UART0 debug port

The HL7900 module's DEBUG port is located on UART0 and can be connected as below:

1. On Linux, grant read-write access right to the Linux user to access the UART port. On Ubuntu, the following can be done:

```
sudo tee /etc/udev/rules.d/70-persistent-usb.rules <<EOM
# FTDI USB2Serial : 0403:6014 0403:6015
SUBSYSTEMS=="usb", ATTRS{idVendor}=="0403", ATTRS{idProduct}=="6014", MODE="0666"
SUBSYSTEMS=="usb", ATTRS{idVendor}=="0403", ATTRS{idProduct}=="6015", MODE="0666"
EOM
sudo udevadm control --reload-rules ; sudo udevadm trigger
```

Note:

The same is automatically applied by the script `init_config.sh` in the Dev-env installation.

2. For capturing firmware traces from UART0,
 1. Connect UART0 port via a micro-USB cable to PC.
 2. On Windows, open Device Manager to identify the "USB Serial Port" which corresponds to the UART0 port.



- On Linux, open a terminal and identify the the `/dev/ttyUSB*` or `/dev/ttyACM*` port that corresponds to the UART port.

```
e.g. dmesg | grep -i ttyusb | tail
```

```
$ dmesg | grep -i ttyusb | tail
```

```
[ 296.126786] usb 3-2 : FTDI USB Serial Device converter now attached to ttyUSB0
```

4 Enable/Disable Firmware traces

Firmware log traces can be collected from the Debug Port of the HL7900 module for debugging. Each log trace line is a `printf`-like string encapsulated by the HL7900 Firmware and sent over Debug Port to the host system.

A firmware log file is in a binary format and must be sent to Semtech for decoding and analysis.

The Debug Port is a physical UART exposed by the HL7900 module and operates at 3M baud optimal speed via an external serial-to-USB driver (A FTDI USB-to-serial controller is installed on the Development Kit).

The Debug Port must be enabled for outputting firmware traces and disabled after use to restore power saving.

Note: *After debug port is enabled, some default log filters are enabled. The filter levels can be retrieved by `AT%LOGLEVEL=<core>` like `AT%LOGLEVEL="MCU"`, `AT%LOGLEVEL="APP"` and `AT%LOGLEVEL="LTEC"`.*

4.1 Enable Firmware traces with filter

The Debug Port must be enabled for outputting firmware traces. Additionally, firmware traces can be filtered using AT commands `AT+LOGCFG`, `AT%LOGLEVEL` and `AT%LOGCMD`.

Refer to [R2 HL7900 AT Command Reference Guide](#) for more information about these AT commands.

Important: *These AT commands are lengthy and prone to manual mistakes, so they need to be handled with cautions.*

Note: *When necessary, some extended trace filters for a given scenario will be provided by Semtech support.*

- Enable the Debug Port

```
AT%SETSRMCFG="DBGUARTMODE", 1
```

This will enable the default logging on UART0 at 3000000 baud with CTS/RTS flow control.

The default logging is sufficient for most basic scenarios. If desired or instructed by Semtech support, the logging filters can be modified as described in step 2.

- Configure the extended trace filters

```
AT+LOGCFG=1,<module-id>,<severity>
```

```
AT%LOGLEVEL=<subsystem>,<module>,<level>,<persistence>
```

```
AT%LOGCMD="ENABLE",<core>
```

Note: AT%SETSRMCFG and AT%LOGCMD= . . . require a device reboot to take effect.

4.2 Disable Firmware traces

After collecting firmware traces, the Debug Port and trace filters must be disabled to restore power saving.

1. Disable the trace filters

```
AT%LOGCMD="DISABLE", "ALL"
```

2. Disable the Debug Port

```
AT%SETSRMCFG="DBGUARTMODE", 2
```

Note: AT%SETSRMCFG and AT%LOGCMD= . . . require a device reboot to take effect.

4.3 Examples of Firmware trace filters

Note: The following examples may not work with every HL7900 firmware.

Note: <module-id> used in AT+LOGCFG=1, <module-id>, . . . can be determined from AT+LOGCFG?.

1. To investigate TCP/IP/UDP issues,

```
AT+LOGCFG=1, <HTTP_ID>, 6
AT+LOGCFG=1, <SOCKET_ID>, 6
AT+LOGCFG=1, <SEM_LEAP1_ID>, 6
AT+LOGCFG=1, <SEM_LEAP2_ID>, 6
AT+LOGCFG=1, <SEM_LEAP3_ID>, 6
```

2. To investigate PPP DUN issues,

```
AT+LOGCFG=1, <SEM_PPP_ID>, 6
AT+LOGCFG=1, <SEM_SIGNAL_ID>, 6
AT+LOGCFG=1, <SEM_UARTSIGNAL_ID>, 6
```

3. To investigate AT Command issues (Response, URC & etc),

```
AT+LOGCFG=1, <SEM_ATCMD_ID>, 6
AT+LOGCFG=1, <SEM_ATMDM_ID>, 6
AT+LOGCFG=1, <SEM_ATPROXY_ID>, 6
```

```
// optional
```

```
AT+LOGCFG=1, <SEM_LEAP1_ID>, 6
AT+LOGCFG=1, <SEM_LEAP2_ID>, 6
AT+LOGCFG=1, <SEM_LEAP3_ID>, 6
AT+LOGCFG=1, <SEM_LEFRK_ID>, 6
```

```
// optional, for 3GPP/Cellular specific Modem AT
AT%LOGLEVEL="APP", "AT_CMDS", "DEBUG", 1
AT%LOGLEVEL="APP", "AT_CLIENT", "DEBUG", 1
```

4. To investigate LTE network and USIM activation issues,

```
AT%LOGCMD="ENABLE", "LTEC"
AT%LOGLEVEL="LTEC", "USIM", "DEBUG", 1
```

Note: For capturing IP logs, below AT command should be configured followed by a module reset.
 at%setacfg="modem_apps.LteSniff.All2msgv", 1

5 Capturing Firmware traces by STT

5.1 Introduction

Serial Trace Tool (stt) is a generic tool for capturing debug traces from a serial trace port.

It can be run standalone to capture raw trace data sent from a serial trace port provided that the serial port driver was properly installed.

5.2 Capturing traces

5.2.1 Usage menu of stt

A usage menu is displayed when running with `stt -h`, dictated below. Parameters *port-name*, *log-file-path* and *baud-rate* are three common parameters to be specified.

The *baud-rate* must match with the active baud used in the trace port. For instance, HL7900 Debug Port uses the default baud 3000000 (3M baud) via USB-to-Serial chip on the Development Kit.

Usage:

```
stt -p <port-name>
    [-l <log-file-path>] [-s <file-split-size>] [-g <gz-compression-ratio>]
    [-r <baud-rate>] [-q <quiet>] [-f <flush-period>]

or

stt <port-name> [<log-file-path>] [<file-split-size>] [<gz-compression-ratio>] [<baud-rate>]

<port-name>      : the trace port, this is mandatory
                  e.g. use COM4 if this corresponds to 'IMC Port (USB1)' in Device manager
<log-file-path>  : path of the trace log file to save,
                  if value is -, write to stdout without saving
                  if file extension is .gz, GZIP compression is enabled
                  Default = stt-<port-name>.gz
<file-split-size> : if <trace-file> is not stdout, stt will split the logging into
                  a separate log file if last written size is over this value.
                  Range: ( 0, 50K - 1024M) default=50M unit=(B,K,M,G) default-unit=K
<gz-compression-ratio> : GZIP compression ratio, larger number means higher compression ratio.
                  Range: (-1, 0 - 9) default=-1
<baud-rate>      : baud-rate, effective for physical trace port
<flush-period>   : time in seconds to flush log data to log file, 0 to disable, min=5s
<quiet>          : -1 - dump data, 0 - print total, 1 - print one dot every 1024bytes, 2 - print nothing

e.g. Capture trace on COM4 port, split log file into about 5Mbytes with max GZ compression-ratio=9
stt COM4 stt-log.gz 5M 9
```

5.2.2 Steps to capture traces

1. Configure trace filters in the Firmware. Refer to [Section 4](#).
2. To start capturing traces, run below command in a system terminal (e.g. bash or cmd.exe) with your trace port
 - On Linux: `./stt -p ttyUSB0 -r 3000000 -l log2.gz -q1`
 - On Windows: `stt -p COM10 -r 3000000 -l log2.gz -q1`

On Linux, a handy script `getlog.sh` is provided to poll for serial port to appear before calling `stt`. Run below command in a terminal with your trace port

- On Linux: `getlog.sh ttyUSB0 3000000`

3. To stop capturing traces, press Ctrl-C in the terminal.
4. Disable trace filters in the Firmware. Refer to [Section 4.2](#).

6 Getting backtraces

Backtraces are non-volatile trace records of crash information. They are stored at non-volatile filesystem `d:/systemlog` in the device storage of the HL7900 module.

Note:

- Crash data for backtrace are temporarily stored as volatile data chunks at pseudo-filesystem `z:/systemlog`. They are collected and stored as a single file at `d:/systemlog` when MCU system boots.
- Each backtrace is stored with file name: `systemlog_<md5sum>_<id>`. The file of the largest number in `<id>` is the latest backtrace.
- A maximum of five backtraces are stored due to limited device storage. When the maximum capacity is reached, rotation occurs, and the oldest file is overwritten.

6.1 Display and erase backtraces

Backtraces can be listed, retrieved or erased by using AT command **AT+KEXCEPT**.

For example,

```
// Erase all backtraces
AT+KEXCEPT=255

OK

// Debugging with new bracktraces generated

... Some debugging here ...

// Display all available backtraces
AT+KEXCEPT=0

+KEXCEPT: 1,"systemlog_4029365450_5"

DABADABAFFFFFFFFFFFFFFFFFFFFFFFDADABABA000000009229120200000000

...

OK
```

6.2 Upload backtraces to FTP server

Backtraces can be uploaded to a desired FTP server using AT command **AT+KBTFUP**.

For example,

```
// Upload all available backtraces
AT+KBTFUP="your.ftpserver.com","user","password",21

OK
```

Note: Refer to [HL7900 AT Command Reference Guide](#) for more information about these AT commands.

7 Reference Documents

For more details, below references can be consulted.

1. HL7900 AT Command Reference Guide, reference number: 41114811
2. HL7900 Getting Started Guide, reference number: 2174364
3. HL7900 Product Technical Specification, reference number: 41114801

END OF DOCUMENT