

# QNX Installation Notes

## Gcom STREAMS Protocol Suite Version 1.51p1.66



### CAUTION

Gcom STREAMS Protocol Suite (SPS) Version 1.51p1.66 is designed to run on QNX 4.24 or 5.25. Not other QNX version is supported.

### Procedure Summary

1. Install synchronous serial adapter.
2. [Install Gcom SPS software](#).
3. [Start the QNX driver](#).
4. [Edit hardware/software test files](#) to run properly on QNX operating system.
5. [Run hardware/software tests](#) to ensure Gcom hardware/software is operational.
6. Contact Gcom Support for help configuring protocol stacks.

### Install Gcom SPS Software

1. Download the compressed archive of Gcom SPS software from at <ftp://ftp.gcom.com/pub/blind/qnx/GcomSPS-1.51p1.66.tar.Z>.
2. Copy archive to the root directory of the system that will run the Gcom SPS software.
3. To uncompress the archive, enter: **gunzip GcomSPS-1.51p1.66.tar.Z**
4. To un-tar the archive and copy all Gcom SPS software files to the correct file system locations, enter: **tar xf GcomSPS-1.51p1.66.tar**

### Start the QNX Driver

Gcom SPS software runs as a QNX driver. To start the driver, enter: **/usr/bin/Dev.gcom &**

**Dev.gcom** starts a family of threads. Typical use: Run the driver at system startup and never terminate the driver process.



### CAUTION

If you choose to terminate the driver, make sure you first terminate all other **Gcom\_** commands. For example: To terminate **Gcom\_monitor**, enter: **Gcom\_dlpi -E** Then kill all the threads.

### Edit Hardware/Software Test Files

**Note:** No matter what protocol you ultimately plan to use, test that Gcom hardware/software is operational using X.25 protocol test files.

In the **/usr/lib/gcom/x25** directory, edit all **start** files to make **Gcom\_monitor** run asynchronously in the correct manner on the QNX operating system. For example, edit **start.onchip** in the following manner:

Edit this code in start.onchip	To this code in start.onchip
<pre>/usr/bin/Gcom_monitor -N -E -B -q -econfigs.onchip</pre>	<pre>/usr/bin/Gcom_monitor -N -E -q -econfigs.onchip &amp;</pre>

That is, delete the **-B** and add the **&**.

## Run Hardware/Software Tests

**Note:** No matter what protocol you ultimately plan to use, test that Gcom hardware/software using X.25 protocol test files.

### In-memory Test

The in-memory test:

- Loads the protocol software into memory.
- Sends data to a logical port that loops the data back within the memory of the computer.

This verifies the software is installed properly and can manipulate data without hardware interaction.

**Note:** This test does not verify proper installation of the adapter.

Synchronous serial adapter/X.25 protocol sample results:

```
Protocol : x25 Test : inmem
Connection Established to address 1111.
Sent 1000 messages, received 1000 messages in 0 seconds
Connection Established to address 2222.
Sent 1000 messages, received 1000 messages in 0 seconds

UPA          CDI_State Frames-out  Frames-in  wd mf af      receive-errs  xmt-errs
           -----
           1 *DRIVER-DOWN*          0           0  0  0  0          0  0  0          0  0
           2 INPUT_ALLOWED      3069        3069  0  0  0          0  0  0          0  0

UPA MODULE DLPI_STATE LPA MUX BIND  LAP_TYPE PRIM  SEC      XMIT      RCV
-----
  1 frame  infoxfer   1  14  01      lapb  01  03      3005      3005

MNR UPA LPA  NPI_State  Type  Chan  Data-out  Data-in  Address  PID
-----
  1 - -  IDLE    *    0    0    0    *      4812
```

1. Type: `cd /usr/lib/gcom/x25`
2. Type: `./start.inmem`
3. Type: `./test.inmem`

The test program sends and receives 1000 messages via the Gcom X.25 protocol module in the kernel. You may press a key at any time to regain a prompt. The terminal prints a status message when transmission is complete. This process should take no more than nine seconds. When the test is complete, the terminal displays a message similar to the following:

```
Connection Established to address 1234.
Sent 1000 messages, received 1000 messages in 9 seconds.
Sent 1111 msgs per second, received 1111 messages per second.
Sent 628375 bytes 558555 baud; received 628375 bytes 558555 baud.
```

4. To unload Gcom monitor software, type: `Gcom_dlpi -X`

## On-chip Test

The on-chip test is similar to the in-memory test, except that the message looping occurs on the adapter card's onboard serial chip. This test, which enables the chip's back-to-back mode to tie the transmitter pin to the receiver pin, verifies:

- The adapter card is properly installed.
- Interrupts are handled properly.
- Data flows to and from the adapter card without corruption.

Synchronous serial adapter/X.25 protocol sample results:

```
Protocol : x25 Test : onchip
Connection Established to address 1234.
Sent 1000 messages, received 1000 messages in 5 seconds
Sent 200 msgs per second, received 200 messages per second
Sent 64607 bytes 103371 baud; received 64607 bytes 103371 baud
UPA          CDI_State Frames-out  Frames-in  wd mf af      receive-errs  xmit-errs
-----
 1  INPUT_ALLOWED      1569        1569    0 0 0         0   0   0         0   0
UPA MODULE DLPI_STATE LPA MUX BIND  LAP_TYPE PRIM  SEC      XMIT      RCV
-----
 1 frame  infoxfer    1 16  01      lapb  01  03      1508     1508
No NPI ungs are in use
```

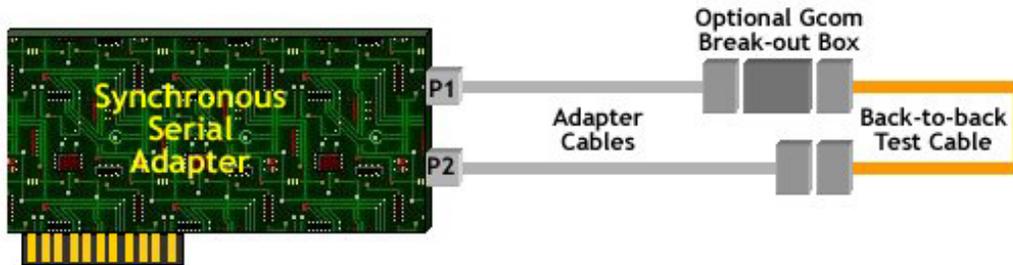
1. Type: `cd /usr/lib/gcom/x25`
2. Type: `./start.onchip`
3. Type: `./test.onchip` (Press **Enter** to regain a prompt.)
4. Type: `./watch`  
The **watch** script, which prints every 10 seconds, lets you monitor the progress of the test as it sends and receives messages on the chip. The **Frames-out** and **Frames-in** counts should steadily rise as the test progresses.
5. To stop the **watch** script, press **Ctrl-C**.
6. To unload Gcom monitor software, type: `Gcom_dlpi -X`

## Back-to-back Test

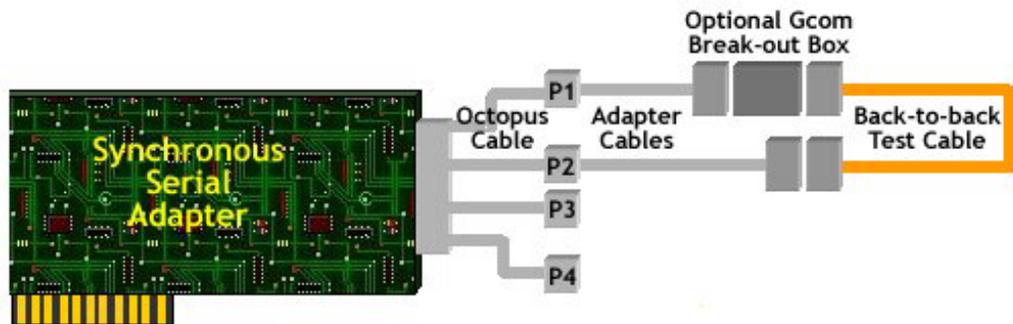
The back-to-back test the ports on the adapter card using a special back-to-back cable (color coded orange). Gcom includes this cable in the Gcom NPK (New Project Kit).

Connect the back-to-back orange test cable to the 2-port or 4-port cable connections as illustrated below. (Using the Gcom Break-out Box is optional, but it does give a visual indication of the protocol in action.)

### Two-port Cable Hookup



### Four-port Quad Cable Hookup



Synchronous serial adapter/X.25 protocol sample results:

```

Protocol : x25 Test : btob

Connection Established to address 1111.
Sent 1000 messages, received 1000 messages in 18 seconds
Sent 55 msgs per second, received 55 messages per second
Sent 128000 bytes 56888 baud; received 128000 bytes 56888 baud
Connection Estab
Sent 1000 messages, received 1000 messages in 18 seconds
Sent 55 msgs per second, received 55 messages per second
Sent 128000 bytes 56888 baud; received 128000 bytes 56888 baud

UPA          CDI_State Frames-out  Frames-in  wd mf af      receive-errs  xmit-errs
-----
1  INPUT_ALLOWED      3044       3043      0 0 0         0   0   0         0   0
2  INPUT_ALLOWED      3043       3043      0 0 0         0   0   0         0   0

UPA MODULE DLPI_STATE LPA  MUX BIND  LAP_TYPE PRIM  SEC      XMIT      RCV
-----
1  frame  infoxfer   1  20  01      lapb   01  03      3017     3017
2  frame  infoxfer   2  22  01      lapb   01  03      3017     3017

MNR UPA LPA  NPI_State  Type  Chan  Data-out  Data-in  Address      PID
-----
1  -  -      IDLE      *    0    0         0     0 *          4891

```

1. Type: `cd /usr/lib/gcom/x25`
2. Type: `./start.btob`
3. Type: `./test.btob` (Press **Enter** at any time to regain a prompt.)
4. Type: `./watch`

The **watch** script, which prints every 10 seconds, lets you monitor the progress of the test as it sends and receives messages on the chip. The **Frames-out** and **Frames-in** counts should steadily rise as the test progresses.

5. To stop the **watch** script, press **Ctrl-C**.
6. To unload Gcom monitor software, type: `Gcom_dlpi -X`

## Common Testing Problems

<b>Problem</b>	<b>Description/Solution</b>
Error Message: Permission denied	All users need write permission to the <code>/usr/spool/gcom</code> directory so that any user running Gcom processes can create the necessary log files.  If you receive this error, ask your system administrator to provide you with the proper permissions access.
Error Message: Monitor already running	You will receive this error if you perform tests using the command line interface and fail to unload the Gcom monitor between tests. After the completion of each test, be sure to type: <b>Gcom_dlp_i -X</b>
Prompt does not reappear	If you perform tests using the command line interface, the command prompt does not reappear while tests are running until you press the <b>ENTER</b> key. The prompt will reappear permanently after the test is completed or aborted with the <b>Gcom_dlp_i-X</b> command.
Other errors	If you encounter errors not mentioned here, check with your system administrator to make sure your system is running properly. If the problem is not with your system, contact Gcom Support.