# Gryphon Hardware Information: LIN card

## External D-9 connector pinout

| Pin # | Channel | Assignment |
|-------|---------|------------|
| 01 | 1 | VIN+ |
| 02 | 1 | No Connect |
| 03 | 1 | LIN BUS |
| 04 | 1 | VIN- / GND |
| 05 | 1 | No Connect |
| 06 | 1 | No Connect |
| 07 | 1 | No Connect |
| 08 | 1 | No Connect |
| 09 | 1 | No Connect |

## BUS termination requirements

To be included.

## Event definitions

The following values may be transmitted as FT_EVENT type GCProtocol frames.
The event value will be in the *event* field of such a frame.
Note: Please see the include file dev_lin.h for numeric values of the following symbols.

| Event name | Event meaning |
|------------|---------------|
| GLINRESET | The processor on the LIN card has been reset |
| GLINBADID | An invalid ID has been received |
| GLINSYNC | Received synch field is not 0x55 |
| GLINWAKEUP | A wakeup signal was received |
| GLINSLEEP | A sleep command was received or bus activity has timed out |
| GLINWAKEUPFAILED | The master failed to respond to the wakeup sequence |
| GLINNEWBITRATE | The bit rate has changed |

Please also see generic events for non card-specific events.

## Card-specific IOCTL definitions

These values are used with the CMD_CARD_IOCTL command.
Please see the include files dev_lin.h and dev_ubp.h for numeric values of IOCTLS and data field bit assignments.

| IOCTL name | Function | Data field length/purpose |
|------------|----------|---------------------------|
| GLINGETBITRATE | Get current bitrate | 4 data bytes |
| GLINSETBITRATE | Set new bitrate | 4 data bytes |

| | | |
|---|---|---|
| GLINGETBRKSPACE | Get current dominant (low) break time in bits | 1 data byte |
| GLINSETBRKSPACE | Set new dominant (low) break time in bits | 1 data byte |
| GLINGETBRKMARK | Get current recessive (high) break time in bits | 1 data byte |
| GLINSETBRKMARK | Set new recessive (high) break time in bits | 1 data byte |
| GLINGETIDDELAY | Get current delay, in bits, between synch field and ID | 1 data byte |
| GLINSETIDDELAY | Set new delay, in bits, between synch field and ID | 1 data byte |
| GLINGETRESPDELAY | Get current delay, in bits, between ID and Data | 1 data byte |
| GLINSETRESPDELAY | Set new delay, in bits, between ID and Data | 1 data byte |
| GLINGETINTERBYTE | Get current Data interbyte delay in bits | 1 data byte |
| GLINSETINTERBYTE | Set new Data interbyte delay in bits | 1 data byte |
| GLINGETWAKEUPDELAY | Get current wakeup delay | 1 data byte |
| GLINSETWAKEUPDELAY | Set new wakeup delay | 1 data byte |
| GLINGETWAKEUPTIMEOUT | Get current wakeup timeout | 1 data byte |
| GLINSETWAKEUPTIMEOUT | Set new wakeup delay | 1 data byte |
| GLINGETWUTIMOUT3BR | Get current wakeup timeout after 3 breaks | 1 data byte |
| GLINSETWUTIMOUT3BR | Set new wakeup timeout after 3 breaks | 1 data byte |
| GLINSENDWAKEUP | Send wakeup signal | no data bytes |
| GLINGETMODE | Get current mode of operation | 1 data byte<br>0 Master or Slave depending on usage<br><br>1 Currently Slave<br><br>2 Currently Master<br><br>3 Slave; may not be Master<br><br>4 Master; must be master |
| GLINSETMODE | Set new mode of operation | 1 data byte |

| | | |
|---|---|---|
| GLINGETSLEW | Get the LIN bus slew rate | 1 data byte<br>    0 Slow slew rate<br>    1 Normal slew rate<br>    2 High slew rate |
| GLINSETSLEW | Set the LIN bus slew rate | 1 data byte |
| GLINADDSCHED | Add a LIN schedule | Bytes 0 thru 3<br>    The number of bytes to follow<br><br>Bytes 4 thru 35<br>    The name of the schedule<br><br>Repeated for each message in the schedule:<br>    4 bytes:  delay time in tenths of milliseconds<br><br>    1 byte:  Number of IDs to follow<br><br>    1 byte:  Low 4 bits: number of data bytes to follow;<br>        bit 6 set if sporadic;  bit 7 set if event<br>        driven<br><br>    1 or more bytes:  ID(s)<br><br>    0 to 8 bytes:  data bytes |
| GLINGETSCHED | Get a LIN schedule | Bytes 0 thru 3<br>    The size of the schedule<br><br>Bytes 4 thru 35<br>    The name of the schedule<br><br>Remainder<br>    returned data for the schedule (see<br>        GLINADDSCHED) |
| GLINGETSCHEDSIZE | Get the size of a LIN schedule | Bytes 0 thru 3<br>    The size of the schedule suitable for use in<br>        GLINGETSCHED<br><br>Bytes 4 thru 35<br>    The name of the schedule |
| GLINDELSCHED | Delete a LIN schedule | Bytes 0 thru 31<br>    The name of the schedule to be deleted<br><br>    if the name is empty (the first byte of the name is a<br>        NULL), all of the schedules are deleted |
| GLINACTSCHED | Activate a LIN schedule | Bytes 0 thru 31<br>    The name of the schedule to be activated |
| GLINDEACTSCHED | Deactivate the currently active LIN schedule | 0 data bytes |
| GLINGETACTSCHED | Get the active LIN schedule | Bytes 0 thru 31<br>    The name of the currently active schedule (returned) |

| GLINGETNUMSCHEDS | Get the number of LIN schedules | 2 data bytes |
|---|---|---|
| GLINGETSCHEDNAMES | Get a list of the LIN schedules | Variable number of data bytes (equals 32 times the number of schedules) |
| GLINSETFLAGS | Set the flag bit(s) for one or more ID's | Byte 0<br>    Starting frame ID (0 through 63)   (If zero, all flags will be cleared first. If non-zero, existing flags will be left intact.)<br><br>Byte 1<br>    The number of frame IDs for which to set the flags<br><br>Bytes 2 thru n<br>    Flag bits as defined below |
| GRESETHC08 | Reset Card (HC080) | 0 data bytes |
| GTESTHC08COP | Test the COP (Computer Operating Properly) module | 0 data bytes |

**notes:**

GRESETHC08 causes the driver to perform a hard reset of the HC08 on the LIN card. This causes all of the information about which slaves are being emulated as well as the slave data itself to be lost. A GLINRESET event is reported while the card's firmware is restarting.

If the main loop of the card's firmware takes more than 16.83 milliseconds between any two passes, the HC08's watchdog expires and resets the processor. A GLINRESET event is reported while the card's firmware is restarting.

GTESTHC08COP prevents the watchdog timer from being reset in the firmware's main loop. If everything is working properly, a GLINRESET event will be reported within 16.38 milliseconds when the watchdog timer expires.

There are up to 4 flag bits per Frame ID that may be used by the driver. Currently only two bits are defined.

Bit 7 - Checksum flag.    Set for Enhanced Checksum:   Reset for Classic Checksum.

Bit 6 - Event flag.    Set if this is an event frame. If so, the unconditional frame's ID code is present in the low order six bits. To allow the Gryphon to emulate slaves that share the same event frame, this bit may also be set in the unconditional frame's entry, allowing a string of unconditional frames to be defined for one event frame.

---

## Card-specific functionality

The LIN card computes the checksum when the slave task sends the *Data* portion of a message to the network. Normally, no checksum should follow the *Data* field of a Gryphon FT_DATA packet sent to the card. If an *Extra Data* byte is present in addition to two or more *Data* bytes, the *Extra Data* byte is used as the checksum when the slave task transmits its data. The explicit checksum byte is always present as the last byte of the *Data* portion of the message

when the slave task transmits it. If the explicit checksum is correct, no indication is present on any receiving Gryphon node. However, if the explicit checksum is incorrect, all receiving Gryphon LIN nodes show the invalid checksum in the *Extra Data* portion of the message.

Any ID checkbits present in the header are discarded by the firmware before it generates them from the lower 6 bits of the header (ID). Thus, it is impossible to send an invalid ID code from the Gryphon.

When receiving messages from the network, the checksums are verified. If the checksum is valid, the data minus the checksum is present in the FT_DATA packet in the *Data* field. If the checksum is invalid, it (the invalid checksum) is placed in the FT_DATA packet in the *Extra Data* field.

By sending just the ID (header) in an FT_DATA packet to the Gryphon, the Gryphon's mode is set to Master, if allowed, and the ID is sent on the LIN bus.

If both the ID and data are sent to the card, the data is stored in the card, indexed by the ID. The firmware will, in the future, automatically send the stored data in response to receiving the corresponding ID.
If the Remote bit in the mode element of the data header is not set, the Gryphon's mode is set to Master, if allowed, and the ID is sent on the LIN bus which causes the data to be sent as well. (The ID is transmitted on the LIN bus followed by the data which was just stored in the card.)
If the Remote bit in the mode element of the data header is set, no further action is taken. (Neither the ID nor the data is transmitted on the LIN bus.)

To clear the data in the card for a given ID, send a data message message to the card with the desired ID, no *Data* bytes and one byte of *Extra Data*. The *Extra Data* byte must have its least signficant bit set. The rest of the bits in the byte should be reset to allow for future use. This message is trapped by the LIN driver which sends a special flag to the card to cause the data to be deleted for that ID.

To clear the data in the card for all of the IDs, send a data message message to the card with any ID, no *Data* bytes and one byte of *Extra Data*. The *Extra Data* byte must have its next to least signficant bit set (set to 2). The rest of the bits in the byte should be reset to allow for future use. This message is trapped by the LIN driver which sends a special flag to the card to cause the data to be deleted for all of the IDs.

The *Transmit/Receive* flag in the Gryphon Data Header indicates whether the Gryphon transmitted or received the header.

The *Local/Remote* flag in the Gryphon Data Header indicates whether the Gryphon transmitted or received the data portion of the message.