

Version 1.1

Jonathan Naylor 26.12.96

This is a description of the messages to be passed between the X.25 Packet Layer and the X.25 device driver. They are designed to allow for the easy setting of the LAPB mode from within the Packet Layer.

The X.25 device driver will be coded normally as per the Linux device driver standards. Most X.25 device drivers will be moderately similar to the already existing Ethernet device drivers. However unlike those drivers, the X.25 device driver has a state associated with it, and this information needs to be passed to and from the Packet Layer for proper operation.

All messages are held in sk_buff's just like real data to be transmitted over the LAPB link. The first byte of the skbuff indicates the meaning of the rest of the skbuff, if any more information does exist.

Packet Layer to Device Driver

First Byte = 0x00 (X25_IFACE_DATA)

This indicates that the rest of the skbuff contains data to be transmitted over the LAPB link. The LAPB link should already exist before any data is passed down.

First Byte = 0x01 (X25_IFACE_CONNECT)

Establish the LAPB link. If the link is already established then the connect confirmation message should be returned as soon as possible.

First Byte = 0x02 (X25_IFACE_DISCONNECT)

Terminate the LAPB link. If it is already disconnected then the disconnect confirmation message should be returned as soon as possible.

First Byte = 0x03 (X25_IFACE_PARAMS)

LAPB parameters. To be defined.

Device Driver to Packet Layer

First Byte = 0x00 (X25_IFACE_DATA)

This indicates that the rest of the skbuff contains data that has been received over the LAPB link.

First Byte = 0x01 (X25_IFACE_CONNECT)

LAPB link has been established. The same message is used for both a LAPB link connect_confirmation and a connect_indication.

First Byte = 0x02 (X25_IFACE_DISCONNECT)

LAPB link has been terminated. This same message is used for both a LAPB link disconnect_confirmation and a disconnect_indication.

First Byte = 0x03 (X25_IFACE_PARAMS)

LAPB parameters. To be defined.

Requirements for the device driver

Packets should not be reordered or dropped when delivering between the Packet Layer and the device driver.

To avoid packets from being reordered or dropped when delivering from the device driver to the Packet Layer, the device driver should not call "netif_rx" to deliver the received packets. Instead, it should call "netif_receive_skb_core" from softirq context to deliver them.