

THE DRAWINGS CONTAINED IN THIS RECOMMENDATION HAVE BEEN DONE IN AUTOCAD.

2.5.4.1 *Initialization*

The STE will perform an MLP initialization by first resetting MV(S), MV(T), and MV(R) to zero and then initializing each of its SLPs. Upon successful initialization of at least one of the SLPs, the STE shall perform the multilink resetting procedure as described in § 2.5.4.2. An SLP initialization is performed according to § 2.4.4.1 of this Recommendation.

Note – An SLP that cannot be initialized should be declared out of service and appropriate recovery action should be taken.

2.5.4.2 *Multilink resetting procedure*

The multilink resetting procedure provides the mechanism for synchronizing the sending and receiving MLPs in both STEs when deemed necessary by either STE. Exact cases when a MLP reset procedure will be invoked are for further study. Following a successful multilink resetting procedure, the multilink sequence numbering in each direction begins with the value 0.

Appendix I provides examples for the multilink resetting procedures when initiated by either a single STE or by both STEs simultaneously.

A multilink frame with $R = 1$ is used to request multilink reset, and a multilink frame with $C = 1$ confirms that the multilink reset process has been completed. An MLP resets MV(S) and MV(T) to zero on transfer of a multilink frame with $R = 1$ and resets the MV(R) to zero on receipt of a multilink frame with $R = 1$.

When the MLP initiates the resetting procedure, it removes all of the unacknowledged multilink frames that are held in that MLP and its associated SLPs, and retains control of those frames. Hereafter, the initiating MLP does not transmit a multilink frame with $R = C = 0$ until the reset process is completed. (One method to remove multilink frames in the SLP is to disconnect the link of that SLP.) The initiating MLP then resets its multilink send state variable MV(S) and its transmitted multilink frame acknowledged state variable MV(T) to zero. The initiating MLP then transmits a multilink frame with $R = 1$ as a reset request on one of its SLPs and starts Timer MT3. The value of the MN(S) field in the $R = 1$ frame may be any value, since when $R = 1$ the MN(S) field is ignored by the receiving MLP. The initiating MLP continues to receive and process multilink frames from the remote MLP, in accordance with the procedures as described in § 2.5.4.4 until it receives a multilink frame with $R = 1$ from the remote MLP.

An MLP which has received a multilink frame with $R = 1$ (reset request) in the normal communication status from an initiating MLP starts the operation as described above; the MLP should receive no multilink frames with $R = C = 0$ until the reset process is completed. Any such frame received is discarded. When the MLP has already initiated its own multilink resetting procedure and has transferred the multilink frame with $R = 1$ to one of its SLPs for transmission, that MLP does not repeat the above operation upon receipt of a multilink frame with $R = 1$ from the remote MLP.

Receipt of a frame with $R = 1$ (reset request) causes the receiving MLP to deliver to the packet layer those packets already received and to identify those multilink frames transmitted but unacknowledged. The packet layer may be informed of the packet loss at the original value of MV(R) and at any subsequent value(s) of MV(R) for which there has been no multilink frame received up to and including the highest numbered multilink frame received. The receiving MLP then resets its multilink receive state variable MV(R) to zero.

After an MLP transmits a multilink frame with $R = 1$ on one of its SLPs, it shall receive

confirmation of successful transfer from that SLP as one of the conditions before transmitting a multilink frame with $C = 1$; when the initiating MLP then received a multilink frame with $R = 1$, and has completed the variable resetting operation above, the initiating MLP transmits a multilink frame with $C = 1$ (reset confirmation) to the remote MLP. When an MLP has

- 1) received a multilink frame with $R = 1$,
- 2) sent a multilink frame with $R = 1$ on one of its SLPs, and
- 3) completed the variable resetting operation above,

that MLP then transmits a multilink frame with $C = 1$ (reset confirmation) to the initiating MLP as soon as possible, given that confirmation of the transfer of the $R = 1$ multilink frame has been received from that SLP. The $C = 1$ multilink frame is a reply to the multilink frame with $R = 1$. The value of the MN(S) field in the above $C = 1$ frame may be any value, since with $C = 1$ the MN(S) field is ignored by the receiving MLP. The multilink sequence number MN(S) received in each direction following multilink reset will begin with the value zero.

When an MLP uses the same or only one SLP to transmit the multilink frame with $C = 1$, the MLP can transmit the multilink frame with $C = 1$ immediately after the multilink frame with $R = 1$ without waiting for SLP indication of transfer completion. An MLP may use two different SLPs as long as one is used for transmitting the multilink frame with $R = 1$ and the other is used for transmitting the multilink frame with $C = 1$ following receipt of the SLP indication of successful transmission of the $R = 1$ multilink frame. A multilink frame with $R = C = 1$ is never used and will be discarded if received.

When an MLP receives the multilink frame with $C = 1$, the MLP stops its Timer MT3. The successful transmission of the multilink frame with $C = 1$ to the remote MLP and the reception of a multilink frame with $C = 1$ from the remote MLP completes the resetting procedure. The first multilink frame transmitted with $R = C = 0$ shall have a multilink sequence number MN(S) value of zero. (The originating MLP, having successfully delivered a multilink frame with $C = 1$ to the remote MLP, and having received a multilink frame with $C = 1$, could immediately transmit multilink frames with $R = C = 0$. However, to insure that the multilink frames with $R = C = 0$ are not discarded because they arrive at the remote MLP prior to the SLP acknowledgement of the reception of the $C = 1$ multilink frame, the MLP should use the same SLP as that which acknowledged receipt of the multilink frame with $C = 1$.)

When the initiating MLP receives a multilink frame with $C = 1$ without having received a multilink frame with $R = 1$, it will retransmit the multilink frame with $R = 1$ and restart its Timer MT3.

When an MLP additionally receives one or more multilink frames with $R = 1$ between receiving a multilink frame with $R = 1$ and transmitting a multilink frame with $C = 1$, the MLP shall discard the extra multilink frames with $R = 1$. When an MLP receives a multilink frame with $C = 1$, which is not a reply to a multilink frame with $R = 1$, the MLP shall discard the multilink frame with $C = 1$.

After an MLP transmits a multilink frame with $C = 1$ on one of its SLPs, the MLP may receive a multilink frame with $R = 1$ from the remote MLP. The MLP shall regard the multilink frame with $R = 1$ as a new reset request and shall start the multilink resetting procedure from the beginning.

When Timer MT3 runs out, the MLP restarts the multilink resetting procedure from the beginning. The value of Timer MT3 shall be large enough to include the transmission, retransmission and propagation delays in the SLPs, and the operation time of the MLP that receives a multilink frame with $R = 1$ and responds with a multilink frame with $C = 1$.

2.5.4.3 *Transmitting multilink frames*

2.5.4.3.1 *General*

The transmitting STE MLP shall be responsible for controlling the flow of packets from the packet level into multilink frames and then to the SLPs for transmission to the receiving STE MLP.

The functions of the transmitting STE MLP shall be to:

- 1) accept packets from the packer layer;
- 2) allocate multilink control fields, containing the appropriate sequence number $MN(S)$, to the packets;
- 3) assure that $MN(S)$ is not assigned outside the MLP transmit window (MW);
- 4) pass the resultant multilink frames to the SLPs for transmission;
- 5) accept indications of successful transmission acknowledgements from the SLPs;
- 6) monitor and recover from transmission failures or difficulties that occur at the SLP sublayer; and
- 7) accept flow control indications from the SLPs and take appropriate actions.

2.5.4.3.2 *Transmission of multilink frames*

When the transmitting MLP accepts a packet from the packet layer, it shall place the packet in a multilink frame, set the $MN(S)$ equal to $MV(S)$, assure that $MN(S)$ is not assigned outside the transmit window (MW), set V, S, R and C to 0, and then increment $MV(S)$ by 1.

In the following, incrementing send and receive state variables is in reference to a continuously repeated sequence series, i.e., 4095 is 1 higher than 4094, and 0 is 1 higher than 4095 for modulo 4096 series.

If the $MN(S)$ is less than $MV(T) + MW$, and the remote STE has not indicated a busy condition on all available links, the transmitting MLP may then assign the new multilink frame to an available link. The transmitting MLP shall always assign the lowest $MN(S)$ unassigned multilink frame first. Also, the transmitting MLP may assign a multilink frame to more than one link. When the SLP successfully completes the transmission of a multilink frame(s)

by receiving an acknowledgement from the remote SLP, it shall indicate this to the transmitting MLP. The transmitting MLP may then discard the acknowledged multilink frame(s). As the transmitting STE receives new indications of acknowledgements from the SLPs, MV(T) shall be advanced to denote the lowest numbered multilink frame not yet acknowledged.

Whenever an SLP indicates that it has attempted to transmit a multilink frame N2 times, the MLP will then assign the multilink frame to the same or one or more other links, unless the MN(S) has been acknowledged on some previous link. The MLP shall always assign the lowest MN(S) frame first.

Note 1 – If an MLP implementation is such that a multilink frame is transmitted on more than one link (e.g., to increase the probability of successful delivery) there is a possibility that one of these multilink frames (i.e., a duplicate) may be delivered to the remote MLP after an earlier one has been acknowledged [the earlier multilink frame would have resulted in the receiving remote MLP having incremented its MV(R) and the transmitting MLP having incremented its MV(T)]. To ensure that an old duplicate multilink frame is not mistaken for a new frame by the receiving remote MLP, it is required that the transmitting MLP shall never send a new multilink frame with MN(S) equal to $MN(S)' - MW - MX$, where $MN(S)'$ is associated with a duplicate multilink frame that is being transmitted on other SLPs, until all SLPs have either successfully transferred the multilink frame or retransmitted the frame their maximum number of times. Alternatively, the incrementing of MV(T) may be withheld until all SLPs have either successfully transferred the multilink frame or retransmitted the frame their maximum number of times. These and other alternatives are for further study.

Flow control is achieved by the window size parameter MW, and through busy conditions being indicated by the remote SLPs.

The MLP will not assign a multilink frame with an MN(S) greater than $MV(T) + MW - 1$. At the point where the next multilink frame to be assigned has a $MN(S) = MV(T) + MW$, the MLP shall hold this and subsequent multilink frames until an indication of acknowledgement that advances MV(T) is received from the SLPs.

The remote MLP may exercise flow control of the MLP by indicating a busy condition over one or more remote STE SLPs. The number of SLPs made busy will determine the degree of MLP flow control realized. When the MLP receives an indication of a remote SLP busy condition from one or more of its SLPs, the MLP may reassign any unacknowledged multilink frames that were assigned to those SLPs. The MLP will assign the multilink frames containing the lowest MN(S) to an available SLP as specified above.

In the event of a circuit failure, an SLP reset or SLP disconnection, all multilink frames unacknowledged on an SLP link shall be retransmitted on an operational SLP(s) which is(are) not in the busy condition.

Note 2 – The action to be taken on the receipt of an RNR frame by the SLP whose unacknowledged multilink frames have been removed for further study.

Note 3 – The means of detecting transmitting MLP malfunctions (e.g., sending more than MW multilink frames) and the actions to be taken are for further study.

2.5.4.4 *Receiving multilink frames*

Any multilink frame less than two octets in length shall be discarded by the receiving STE.

Note 1 – The procedures to be followed by the receiving STE when V and/or S is equal to 1 are for further study.

When the STE receives multilink frames from one of its SLPs, the STE will compare the multilink sequence number $MN(S)$ of the received multilink frame to its multilink receive state variable $MV(R)$, and act on the frame as follows:

- a) If the received $MN(S)$ is equal to the current value of $MV(R)$, i.e., is the next expected in-sequence multilink frame, the MLP delivers the packet to the packet layer.
- b) If the $MN(S)$ is greater than the current value of $MV(R)$ but less than $[MV(R) + MW + MX]$, the MLP keeps the received multilink frame until condition a) is met, or discards it if it is a duplicate.
- c) If the $MN(S)$ is other than that in a) and b) above, the multilink frame is discarded.

Note 2 – In case c above the recovery from the desynchronization greater than MX between the local and the remote MLP, i.e., the value of $MN(S)$ assigned to new multilink frames at the remote MLP is higher than $MV(R) + MW + MX$ at the local MLP, is for further study.

On receipt of a multilink frame, $MV(R)$ is incremented in the following way:

- i) If $MN(S)$ is equal to the current value of $MV(R)$, the $MV(R)$ is incremented by the number of consecutive in-sequence multilink frame received. If additional multilink frames are awaiting delivery pending receipt of a multilink frame with $MN(S)$ equal to $MV(R)$, then Timer MT1 (see § 2.5.5.1) is restarted; otherwise MT1 is stopped.
- ii) If $MN(S)$ is greater than the current value of $MV(R)$ but less than $MV(R) + MW$, $MV(R)$ remains unchanged. Timer MT1 is started, if not already running.
- iii) If $MN(S)$ is $MV(R) + MW$ but $< MV(R) + MW + MX$, $MV(R)$ is incremented to $MN(S) - MW + 1$ and then the packet layer may be informed of the packet loss at the original value of $MV(R)$. As $MV(R)$ is being incremented, if the multilink frame with $MN(S) = MV(R)$ has not yet been received, the packet layer may be informed of the packet loss also; if the multilink frame with $MN(S) = MV(R)$ has been received, it is delivered to the packet layer. After $MV(R)$ reaches $MN(S) - MW + 1$, it may then be incremented further as above until the first unacknowledged $MN(S)$ is encountered (see Figure 4/X.75).
- iv) If the $MN(S)$ is other than that in i), ii) and iii) above, $MV(R)$ remains unchanged.

If Timer MT1 runs out, $MV(R)$ is incremented to $MN(S)$ of the next multilink frame awaiting delivery to the packet layer and then the packet layer may be informed of the packet loss at the original $MV(R)$. The procedure follows i) and a) above as long as there are consecutive in-sequence multilink frames which have been received.

When flow control of the other MLP is desired, one or more SLP(s) may be made to indicate a busy condition. The number of remote SLPs made busy determines the degree of flow control realized.

If the MLP can exhaust its receive buffer capacity before resequencing can be completed, Timer MT2 (see § 2.5.5.2 below) may be implemented. Whenever a busy condition is indicated by the MLP on all its SLPs, and multilink frames at the MLP are awaiting resequencing, Timer MT2 shall be started. When the busy condition is cleared on one or more SLPs by the MLP, Timer MT2 shall be stopped.

If Timer MT2 runs out, the multilink frame with $MN(S) = MV(R)$ is blocked and shall be considered lost. $MV(R)$ shall be incremented to the next sequence number not yet received, and the packets contained in multilink frames with intervening multilink sequence numbers are delivered to the packet layer. Timer MT2 shall be restarted if the busy condition remains in effect on all SLPs and more multilink frames are awaiting resequencing.

Figure 4/X.75 - T0702230-87

2.5.4.5 Retransmission of multilink frames

If an SLP has retransmitted a multilink frame $MN1$ times, the STE will then assign the multilink frame to the same or one or more other links, unless the $MN(S)$ has been acknowledged on some previous link. The STE shall always reassign the lowest $MN(S)$ frame first. The first SLP transmits the frame $N2$ times, regardless of the value of $MN1$.

Note – The procedures associated with the reassigning of multilink frames from a link of poor quality (e.g., before $N2$ transmissions) to other links are for further study.

2.5.4.6 *Taking an SLP out of service*

An SLP may be taken out of service for maintenance, traffic, or performance considerations.

An SLP is taken out of service by disconnecting at the physical layer or the data link layer. Any outstanding multilink frames will be treated as in § 2.5.4.1. The usual procedure would be to flow control the remote SLP by an RNR, and then to disconnect logically the local SLP (see § 2.4.4.3 above).

If Timer T1 has run out N2 times and the SLP resetting procedure is unsuccessful, then the SLP will enter the disconnected phase, taking the SLP out of service (see §§ 2.4.5.8 and 2.4.7.2 above).

Note – In the case when all SLPs are out of service, the recovery mechanism is based on initiating the MLP reset procedure. Additional recovery procedures are for further study.

2.5.5 *List of multilink system parameters*

2.5.5.1 *Lost-frame timer MT1*

Timer MT1 is used at a receiving STE to provide a means to identify during low traffic periods that the multilink frame with MN(S) equal to MV(R) is lost.

2.5.5.2 *Group busy timer MT2*

Timer MT2 is provided at a receiving STE to identify a “blocked” multilink frame condition (e.g., a buffer exhaust situation) that occurs before required resequencing can be accomplished. MT2 is started when all SLPs are busy and there are multilink frames awaiting resequencing. If MT2 runs out before the “blocked” multilink frame MV(R) is received, the “blocked” multilink frame(s) is(are) declared lost. MV(R) is incremented to the value of the next in-sequence multilink frame to be received, and any packets intervening multilink frames are delivered to the packet layer.

Note – MT2 may be set to infinity; e.g., when the receiving STE always has sufficient storage capacity.

2.5.5.3 *MLP reset confirmation timer MT3*

Timer MT3 is used by the MLP to provide a means of identifying that the remote MLP multilink frame with the C bit set to 1 that is expected following the transmission of the MLP multilink frame with R bit set to 1 has not been received.

2.5.5.4 *Retransmission attempts MN1*

MN1 has a value between zero and the smallest N2 over all SLPs inclusive. If a multilink frame is to be retransmitted at the SLP sublayer, MN1 retries indicates when action may be taken at the MLP sublayer.

3 **Packet layer procedures between signalling terminals**

General principles

Section 3 of this Recommendation relates to the transfer of packets at the STE–X/STE–Y (X/Y) interface. The procedures apply to packets which are successfully transferred across the X/Y

interface.

Each packet to be transferred across the X/Y interface shall be contained within the link layer information field which will delimit its length, and only one packet shall be contained in the information field of an I frame.

Note – Some networks require the data field of packets to contain an integral number of octets. The arrangements for interworking with such networks is subject to bilateral agreement between Administrations. The transmission by a DTE of data fields not containing an integral number of octets to the network may cause a loss of data integrity.

To enable simultaneous virtual calls and/or permanent virtual circuits, logical channels are used. Each virtual call and permanent virtual circuit is assigned a logical channel group number (in the range 0 to 15 inclusive) and a logical channel number (in a range of 0 to 255 inclusive). For virtual calls, a logical channel group number and a logical channel number are assigned during the call set-up phase. The range of logical channel and logical channel groups that are available for assignment to virtual calls is agreed bilaterally for a period of time. For permanent virtual circuits using the static method, a logical channel group number and a logical channel number are assigned at the time of establishment (see Recommendation X.181). Procedures for a dynamic method are for further study.

The combination of logical channel number 0 and logical channel group number 0 will not be used for virtual calls and permanent virtual circuits.

In the case that multiple STE X/Y interfaces are used between two networks, virtual calls may be distributed over the available STEs. STE selection may be performed once by the originating and each transit network for a call request. The procedure for selecting the particular X/Y interface is network dependent. During the existence of a particular virtual call, each packet related to that call uses the STEs selected at call set-up.

For permanent virtual circuit, each packet related to that circuit uses the STEs selected at establishment time of the permanent virtual circuit. In the case that multiple X/Y interfaces are used between two networks, bilateral agreement is necessary selecting the specific STE X/Y interface to be used.

In the case that multiple STE X/Y interfaces are used between two networks, the networks may apply network utilities and their parameters either in common or independently to the STE X/Y interfaces.

For virtual calls, it is assumed that the gathering of information required for charging and accounting should normally be the responsibility of the calling Administration (see Recommendation D.10). Other arrangements for gathering information are for further study. For permanent virtual circuit, responsibility of gathering information required for charging and accounting should normally be the source Administration (see Recommendation X.181).

The group of logical channels to be assigned for permanent virtual circuits has to be agreed bilaterally between Administrations.

3.1 *Procedures for virtual call set-up and clearing*

Virtual calls will be set up and cleared according to the procedures described hereunder. The procedures for calls set-up and clearing are only applicable when a logical channel is in the *packet layer ready* state (r1). In all other *r* states these procedures are not applicable.

3.1.1 *Ready state*

If there is no call or call attempt in existence and if call set-up is possible, the logical channel is in the *ready* state (p1), within the *packet layer ready* state (r1).

3.1.2 *Call request packet*

An STE indicates a call request by transferring a *call request* packet which specifies a logical channel in the *ready* state (p1) across the X/Y interface. The logical channel selected by the calling STE is then in the STE *call request* state (p2/3). If this state persists for more than T31, the calling STE will clear the call. The value of T31 is 200 seconds (see Annex D).

Note – In the *call request* packet, bit 7 of the general format identifier (see § 4.1.1) may be used in conjunction with the delivery confirmation procedure (see § 3.3.4). The bit 7 is conveyed transparently through an STE.

3.1.3 *Call connected packet*

The called STE will indicate acceptance of the call by the called DTE by transferring across the X/Y interface a *call connected* packet specifying the same logical channel as that of the *call request* packet. This places the specified logical channel in the *flow control ready* state (d1) within the *data transfer* state (p4). The procedure applying to the *data transfer* state is specified in § 3.3 below.

Note – In the *call connected* packet, bit 7 of the general format identifier (see § 4.1.1) may be used in conjunction with the delivery confirmation procedure (see § 3.3.4). This bit 7 is conveyed transparently through an STE.

3.1.4 *xe ""§Call collision*

Call collision occurs if STE–X receives a *call request* packet when the logical channel specified is in state p2 or if the STE–Y receives a *call request* packet when the logical channel specified is in state p3. In these cases, both calls shall be cleared. The clearing cause field shall be coded “Network congestion”.

In order to reduce the occurrence of this situation, inverse order testing of logical channels will be used. The *call request* packet of one STE will use the logical channel in the *ready* state with the lowest number; the *call request* packet of the other STE will use the logical channel in the *ready* state with the highest number. Which STE will use the lowest number and which the highest number will be agreed bilaterally.

3.1.5 *xe ""§Clear request packet*

An STE may request clearing of a logical channel in any state by transferring across the X/Y interface a *clear request* packet specifying the logical channel. If the STE *clear request* state persists for more than T33, the actions taken by the STE are given in Annex D. The value of T33 is 180 seconds.

The clearing cause field will be coded according to the reason for clearing. Each STE shall be capable of generating the distinct codes for all of the call progress signals specified in Recommendation X.96 for the packet-switched data transmission service.

3.1.6 *xe ""§Clear confirmation packet*

When an STE–X or STE–Y (STE X/Y) has received a *clear request* packet, it will free the

logical channel, whatever the state of the logical channel except the STE X/Y *clear request* state (p6 or p7 respectively), and transfer across the X/Y interface a *clear confirmation* packet specifying the same logical channel. The logical channel is placed in the *ready* state (p1) within the *packet layer ready* state (r1). The receipt of a *clear confirmation* packet cannot be interpreted as an indication of the remote DTE being cleared.

3.1.7 *Clear collision*

If a logical channel is in the STE X/Y *clear request* state (p6 or p7 respectively) and the STE X/Y receives a *clear request* packet specifying the same logical channel, this STE will consider the clearing completed and will not transmit a *clear confirmation* packet. This logical channel is now in the *ready* state (p1) within the *packet layer ready* state (r1).

3.2 *Procedures for permanent virtual circuit service*

Figures B-1/X.75 and B-3/X.75 show the state diagrams which give a definition of events at the packet layer X/Y interface for logical channels assigned for permanent virtual circuits.

For permanent virtual circuits there is no call set-up or clearing. The procedures for the control of packets between STEs while in the *data transfer* state are contained in § 3.3.

In case of momentary failure within the network, the STE will reset the permanent virtual circuit as described in § 3.4.2, with the cause “Network congestion”, and then will continue to handle data traffic.

If the network has a temporary inability to handle data traffic, the STE shall reset the permanent virtual circuit with the cause “Network out of order”. When the network is again able to handle data traffic, the STE should reset the permanent virtual circuit with the cause “Network operational”.

3.3 *Procedure for data and interrupt transfer*

The data transfer procedure described below applies independently to each logical channel existing at the X/Y interface.

Normal network operation dictates that user data in *data* packets and interrupt data are all passed transparently, unaltered through the network. The order of bits within these packets is preserved. A packet sequence received by an STE is always delivered as a complete packet sequence.

3.3.1 *States for data transfer*

Data, interrupt, flow control and *reset* packets may be transmitted and received by an STE in the *data transfer* state (p4) of the *packet layer ready* state (r1) of a logical channel at the X/Y interface. Only in this state, do the flow control and reset procedures described in § 3.4 below apply to data transmission on that logical channel to and from the STE. In all other *r* or *p* states the data and interrupt transfer, flow control, and reset procedures are not applicable.

3.3.2 *Numbering of data packets*

Each data packet transmitted at the X/Y interface for each direction of transmission in a virtual call or permanent virtual circuit is sequentially numbered. This sequential numbering is performed regardless of the layer of data [value of the qualifier (Q) bit].

The sequence numbering scheme of the packets is performed modulo 8 or 128. This modulo is common to all logical channels at the X/Y interface. The packet sequence numbers cycle through the entire range 0 to 7 or 0 to 127 respectively. The selection of modulo 8 or 128 is done by bilateral agreement.

Only *data* packets contain this sequence number called the packet send sequence number P(S).

The first data packet to be transmitted across the X/Y interface for a given direction of data transmission when the logical channel has just entered the *flow control ready* state (d1), has a packet send sequence number equal to 0.

If an STE receives the first *data* packet with a packet send sequence number not equal to 0 after entering the *flow control ready* state (d1), it will reset the virtual call or permanent virtual circuit indicating the cause “Network congestion”.

3.3.3 *Data field length of data packets*

The standard maximum data field length is 128 octets (1024 bits) and is provided by all Administrations. In addition for virtual calls, optional maximum data field lengths may be provided on a per call basis by bilateral agreement between Administrations in conjunction with an optional network utility defined in § 5.3.5 (see Note). For permanent virtual circuits, optional maximum data field length may be provided on a “per permanent virtual circuit” basis by bilateral agreement between Administrations and could be selected at establishment time. The value selected, in conjunction with the window size selected in § 3.4.1.1 has to satisfy the throughput class agreed between networks and end users at establishment time for a specific permanent virtual circuit. The attainable throughput at the STE X/Y interface is limited by the line characteristics and the traffic characteristics of other logical channels at the STE X/Y interface.

The data field length may contain any number of bits from 0 up to the agreed maximum data field length.

If an STE receives a *data* packet having a data field exceeding the maximum data field length, it will reset the virtual call or the permanent virtual circuit indicating the cause “Network congestion”.

Note – Optional maximum data field lengths may be selected from the following list: 16, 32, 64, 256, 512 and 1024 octets. Maximum data field lengths of 2048 and 4096 octets are for further study.

3.3.4 *Delivery confirmation, more data and qualifier bits*

The setting of these “”§ Delivery confirmation bit (or D bit) is used to indicate whether or not an end-to-end acknowledgement of delivery is required for data being transmitted, this information being provided by means of the packet receive sequence number P(R) (see § 3.4.1.2).

A packet sequencing method is provided to enable coherent transmission of data longer than the maximum data field length of *data* packets.

Each complete packet sequence consists of any number (including 0) of full *data* packets (full means that the data field contains the bit number of the maximum data field length) with $M = 1$ and $D = 0$, followed by one other packet of any length up to (and including) the maximum with either $M = 0$ and $D = 0$ or 1, or $M = 1$ and $D = 1$. If an STE receives a packet which is not full, and which has the D bit set to 0 but the M bit set to 1, it will reset the virtual call or the permanent virtual circuit; the resetting cause shall be “Network congestion”.

A complete packet sequence may be one of two levels as indicated by the *Qualifier* bit (or Q bit).

The value of the Q bit should not change within a complete packet sequence. If an STE detects that the value of this bit has changed within a packet sequence, it may reset the virtual call or the permanent virtual circuit; the resetting cause shall be “Network congestion”.

Note – The value of the Q bit in a *data* packet, which follows a *data* packet with either *M* = 0 or both the *M* and *D* bits set to 1, may be set independently of the value of the Q bit in the previous packet.

3.3.5 *Interrupt procedure*

The interrupt procedure allows a DTE to transmit data to the remote DTE, without following the flow control procedure applying to *data* packets between STEs (see § 3.4 below). The interrupt procedure can only apply in the *flow control ready* state (d1) within the *data transfer* state (p4).

The interrupt procedure has no effect on the transfer and flow control procedures applying to the *data* packets on the virtual call or the permanent virtual circuit.

If an STE receives an *interrupt* packet with a user data field longer than 32 octets, the STE should reset the virtual call or the permanent virtual circuit.

An STE conveys an interrupt by transferring across the X/Y interface an *interrupt* packet. The other STE will convey the interrupt confirmation by transferring an *interrupt confirmation* packet.

The receipt of an *interrupt confirmation* packet indicates that the interrupt has been confirmed by the remote DTE by means of a *DTE interrupt confirmation* packet.

An *interrupt* packet is conveyed across the X/Y interface at or before the point in the stream of *data* packet at which it was generated by the DTE.

An STE receiving a further *interrupt* packet in the time between receiving one *interrupt* packet and transferring the *interrupt confirmation*, may either discard this *interrupt* packet or reset the virtual call or the permanent virtual circuit.

3.4 *Procedures for flow control and for reset*

The procedures for flow control of *data* packets and for reset only apply to the *data transfer* state (p4) and are specified below.

3.4.1 *Procedure for flow control*

At the X/Y interface of each logical channel used for a virtual call or a permanent virtual circuit, the transmission of *data* packets is controlled separately for each direction and is based on authorizations from the receiver.

3.4.1.1 *Window description*

At the X/Y interface of each logical channel used for a virtual call or a permanent virtual circuit, a window is defined for each direction of data transmission as the ordered set of *W* consecutive packet send sequence numbers of the *data* packets authorized to cross the interface.

The lowest sequence number in the window is referred to as the lower window edge "lower window edge". When a virtual call or a permanent virtual circuit at the X/Y interface has just

been established or reset, the window related to each direction of data transmission has a lower window edge equal to 0. The packet send sequence number of the first *data* packet not authorized to cross the interface is the value of the lower window edge plus W (modulo 8 or 128).

The maximum value of the window size for each direction of transmission at the X/Y interface is common to all the logical channels and is agreed for a period of time bilaterally. This value does not exceed 7 or 127 (modulo 8 or 128).

For a particular virtual call or a permanent virtual circuit two window sizes may be selected, one for each direction of transmission. These window sizes may be less than or equal to the above-mentioned maximum. For virtual calls, the two sizes are selected by reference to a utility (see § 5.3.4) in the network utility field of the *call request* packet and the *call connected* packet, and, in some cases, by reference also to a correspondence table relating window size to throughput class. This table is agreed for a period of time between Administrations. For permanent virtual circuits two window sizes are selected at the establishment time and agreed between Administrations. The values selected in conjunction with the data field length selected in § 3.3.3 has to satisfy the throughput class agreed between networks and end users at establishment time for a specific permanent virtual circuit. The attainable throughput at the STE X/Y interface is limited by the line characteristics and the traffic characteristics of other logical channels at the STE X/Y interface.

3.4.1.2 Flow control principles

A number modulo 8 or 128 referred to as a packet receive sequence number P(R), conveys across the X/Y interface information from the receiver for the transmission of *data* packets. When transmitted across the X/Y interface, a P(R) becomes the lower window edge. In this way, additional *data* packets may be authorized by the receiver to cross the X/Y interface.

When the sequence number P(S) of the next *data* packet to be transmitted by the STE is within the window, the STE is authorized to transmit this *data* packet to the other STE, which may then accept it. When the sequence number P(S) of the next *data* packet to be transmitted by the STE is outside the window, the STE shall not transmit a *data* packet to the other STE. Otherwise, the other STE will consider the receipt of this *data* packet as a procedure error and will reset the virtual call or the permanent virtual circuit.

The packet receive sequence number, P(R), is conveyed in *data*, *receive ready* (RR) and *receive not ready* (RNR) packets, and implies that the STE transmitting the P(R) has accepted at least all *data* packets numbered up to and including $P(R) - 1$.

The value of a P(R) received by the STE must be within the range starting from the last P(R) received by the STE up to and including the packet send sequence number of the next *data* packet to be transmitted by the STE. Otherwise, the STE will consider the receipt of this P(R) as a procedure error and will reset the virtual call or the permanent virtual circuit.

When the D bit is set to 0 in a *data* packet $[P(S) = p]$, the significance of the P(R) [i.e., $P(R) = p + 1$] corresponding to that *data* packet is a local updating of the window across the packet layer interface.

When the D bit is set to 1 in a *data* packet $[P(S) = p]$, the significance of the P(R) received corresponding to the *data* packet [i.e., $P(R) = p + 1$] is an indication that a P(R) has been received from the remote DTE for all data bits in the *data* packet in which the D bit had originally been set to 1 [i.e., $P(S) = p$].

Note 1 – The STE is required to send a P(R) corresponding to a *data* packet with the D bit set to 1 as soon as possible after it receives the P(R) from the remote DTE. An RNR packet may be used in this case if necessary.

Note 2 – In the case where a P(R) for a *data* packet with the D bit set to 1 is outstanding, local updating of the window will be deferred for subsequent *data* packets with the D bit set to 0. Some STEs may also defer updating of the window for previous *data* packets (within the window) with the D bit set to 0.

3.4.1.3 *STE receive ready (RR) packet*

RR packets are used by the STE to indicate that it is ready to receive the *W data* packets within the window starting with P(R), where P(R) is indicated in the *RR* packet.

3.4.1.4 *STE receive not ready (RNR) packet*

RNR packets are used by the STE to indicate a temporary inability to accept additional *data* packets for the virtual call or the permanent virtual circuit. An STE receiving an *RNR* packet shall stop transmitting *data* packets on the indicated logical channel but the window is updated by the P(R) indicated in the *RNR* packet.

The receive not ready situation indicated by the transmission of an *RNR* packet is cleared by the transmission in the same direction of an *RR* packet or by a reset procedure being initiated.

The transmission of an *RR* after an *RNR* at the packet layer is not to be taken as a demand for retransmission of packets which have already been transmitted.

3.4.2 *Procedure for reset*

The reset procedure is used to reinitialize the virtual call or the permanent virtual circuit. The reset procedure only applies in the *data transfer* state (p4) of the X/Y interface. In any other state of the interface the reset procedure is not applicable.

There are three states within the *data transfer* state (p4). They are *flow control ready* (d1), *STE-X reset request* (d2) and *STE-Y reset request* (d3). When entering state p4, the logical channel is placed in state d1.

When a virtual call or a permanent virtual circuit at the X/Y interface has just been reset, the window related to each direction of data transmission has a lower window edge equal to 0, and the numbering of subsequent *data* packets to cross the X/Y interface for each direction of data transmission shall start from 0.

3.4.2.1 *xe ""\$Reset request packet*

The STE shall indicate a request for reset by transmitting a *reset request* packet specifying the logical channel. This places the logical channel in the *reset request* state (d2 or d3).

In this state, the STE will discard *data*, *interrupt*, *RR* and *RNR* packets.

3.4.2.2 *Reset collision*

Reset collision occurs when both STEs simultaneously transfer a *reset request* packet. In this case both STEs shall consider that resetting is complete and shall not transfer a *reset confirmation* packet. The logical channel is then in the *flow control ready* state (d1).

3.4.2.3 *Reset confirmation packet*

When the logical channel is in the *reset request* state, the requested STE will confirm reset

by transmitting to the requesting STE a *reset confirmation* packet. This places the logical channel in the *flow control ready* state (d1).

The *reset confirmation* packet can only be interpreted universally as having local significance; however, within some Administrations' networks, reset confirmation may have end-to-end significance. If the *reset request* state persists for more than T32, the actions taken by the STE are given in Annex D. The value of T32 is 180 seconds.

3.4.2.4 *Effect of reset procedure on data and interrupt packets*

Data and *interrupt* packets, transmitted by an STE before a reset procedure is initiated at its X/Y interface, will either be delivered before the corresponding reset procedure is initiated at the remote DTE/DCE interface, or discarded.

The first *data* and *interrupt* packets transmitted by an STE after a reset procedure is completed at its interface will be the first packets delivered after the corresponding reset procedure is completed at the remote DTE/DCE interface.

Data and *interrupt* packets transmitted by an STE after a reset procedure has been initiated by the other STE will be discarded by the latter STE until the reset procedure has been completed at the X/Y interface.

3.5 *Procedure for restart*

The restart procedure is used to clear simultaneously all the virtual calls and/or reset all the permanent virtual circuits at the X/Y interface.

There are three states of the X/Y interface concerned with the restart procedure. They are *packet layer ready* (r1), *STE-X restart request* (r2) and *STE-Y restart request* (r3). When entering state r1, all logical channels are placed in state p1.

3.5.1 *Restart by the STE*

The STE may at any time request a restart by transferring across the X/Y interface a *restart request* packet. The interface for each logical channel is then in the *request* state (r2 or r3).

In this state of the X/Y interface, the STE will discard all packet types except *restart request* and *restart confirmation* packets.

On receipt of a *restart request* packet, an STE shall clear all virtual calls and reset all permanent virtual circuits and shall place logical channels used for virtual calls in the *ready* state (p1) and the logical channels used for permanent virtual circuits in the *flow control ready* state (d1). The STE shall return a *restart confirmation* packet unless a collision has occurred.

The *restart confirmation* packet can only be interpreted universally as having local significance. If the *restart request* state persists for more than T30, the actions taken by the STE are given in Annex D. The value of T30 is 180 seconds.

3.5.2 *Restart collision*

Restart collision can occur when both STEs simultaneously transfer *restart request* packets. Under these circumstances, both STEs will consider that the restart is completed and will not expect a *restart confirmation* packet, neither will they transfer a *restart confirmation* packet.

3.6 *Relationship between layers*

Changes of operational states of the physical and link layer of the X/Y interface do not implicitly change the state of each logical channel at the packet layer. Such changes, when they occur, are explicitly indicated at the packet layer by the use of restart, clear or reset procedures as appropriate.

However, in some cases of trouble at the link layer, it may be appropriate to initiate the restart procedure, and accept no more new virtual calls or no more *data* packets on permanent virtual circuits.

A failure on the physical and/or link layer is defined as a condition in which the STE cannot transmit and receive any frames because of abnormal conditions caused by, for instance, a line fault between STEs.

When a failure on the physical and/or link layer is detected, virtual calls will be cleared and permanent virtual circuits will be declared out of order. The STE will transmit to the remote end in the network:

- 1) a reset with the cause “Network out of order” and the appropriate diagnostic for each permanent virtual circuit; and
- 2) a clear with the cause “Network congestion” and the appropriate diagnostic for each existing virtual call.

During the failure:

- 1) the STE will clear any virtual call with the cause “Network congestion” and an appropriate diagnostic;
- 2) for any *data* or *interrupt* packet received from the remote DTE on a permanent virtual circuit, the STE will reset the permanent virtual circuit with the cause “Network out of order” and an appropriate diagnostic;
- 3) a *reset request* packet received from the remote end on a permanent virtual circuit will be confirmed to the remote end by either a *reset confirmation* or *reset request* packet.

The appropriate diagnostic value depends on whether the failure was unexpected or the result of planned maintenance action; the values are No. 115 and No. 122 respectively (see also Note 3 of Annex E).

When the failure is recovered on the physical and link layers, the restart procedure will be actioned with the cause “Network operation” and a reset with the cause “Network operational” will be transmitted to both ends of each permanent virtual circuit going through the X/Y interface.

In other out-of-order conditions on the physical and/or link layer, the STE will clear virtual calls and reset permanent virtual circuits.

4 **xe ""§Packet formats for virtual calls and permanent virtual circuits**

4.1 *General*

The formats of Recommendation X.75 packets are based on the general structure of packets in Recommendation X.25. It is anticipated that modification in Recommendation X.25 control packet formats will also be adopted in this Recommendation.

The possible extension of packet formats by the addition of new fields is for further study.

Bits of an octet are numbered 8 to 1 where bit 1 is the low order bit and is transmitted first. Octets of a packet are consecutively numbered starting from 1 and are transmitted in this order.

4.1.1 *xe ""§General format identifier*

The general format identifier field is a four-bit binary coded field which is provided to indicate the general format of the rest of the header. The general format identifier field is located in bit positions 8, 7, 6 and 5 of octet 1 and 5 is the low order bit (see Table 11/X.75).

TABLE 11/X.75

General format identifier

	Octet 1 Bits			
	8	7	6	5
General format identifier				

Data packets	Sequencing numbers scheme modulo 8	X	X	0	1
	Sequencing numbering scheme modulo 128	X	X	1	0
Call set-up packets	Sequencing numbering scheme modulo 8	0	X	0	1
	Sequencing numbering scheme modulo 128	0	X	1	0
Clearing, flow control, interrupt, reset and restart packets	Sequencing numbering scheme modulo 8	0	0	0	1
	Sequencing numbering scheme modulo 128	0	0	1	0
General format identifier extension		U	U	1	1
Reserved format for other applications		U	U	0	0

Note – A bit which is indicated as X may be set to either 0 or 1 as specified in the text and in Figures 3/X.75, 4/X.75, 7/X.75 and 8/X.75. A bit which is indicated as U is unspecified.

Bit 8 of the general format identifier is used for the qualifier (Q) in *data* packets and is set to 0 in all other packet types.

Bit 7 of the general format identifier is used in *data* and in *call set-up* packets in conjunctions with the *delivery confirmation* (D) procedure, and is set to 0 in all other packet types.

Bits 5 and 6 are encoded for four possible indications. Two of the codes are used to distinguish packets using modulo 8 sequence numbering scheme from packets using modulo 128 sequence numbering scheme. The third code is used to indicate an extension to an extended family of general format identifier codes and extended formats which are a subject for further study. The fourth code is unassigned.

4.1.2 *Logical channel group number*

The logical channel group number appears in every packet except in *restart* packets (see § 4.5 below) in bit positions 4, 3, 2 and 1 of octet 1. This field is binary coded and bit 1 is the low order bit of the logical channel group number.

For each logical channel, this number has local significance at the X/Y interface.

4.1.3 *Logical channel number*

The logical channel number appears in every packet except in *restart* packets (see § 4.5 below) in all bit positions of octet 2. This field is binary coded and bit 1 is the low order bit of the logical channel number.

For each logical channel, this number has local significance at the X/Y interface.

4.1.4 *Packet type identifier*

Each packet shall be identifier in octet 3 of the packet according to Table 12/X.75.

TABLE 12/X.75

Packet type identifier	
	Octet 3 Bits

Packet Type	8	7	6	5	4	3	2	1
<i>Call set-up clearing</i>								
Call request	0	0	0	0	1	0	1	1
Call connected	0	0	0	0	1	1	1	1
Clear request	0	0	0	1	0	0	1	1
Clear confirmation	0	0	0	1	0	1	1	1
<i>Data and interrupt</i>								
Data	X	X	X	X	X	X	X	0
Interrupt	0	0	1	0	0	0	1	1
Interrupt confirmation	0	0	1	0	0	1	1	1
<i>Flow control and reset</i>								

Receive ready (modulo 128)	0	0	0	0	0	0	0	1
Receive ready (modulo 8)	X	X	X	0	0	0	0	1
Ready not ready (modulo 128)	0	0	0	0	0	1	0	1
Ready not ready (modulo 8)	X	X	X	0	0	1	0	1
Reset request	0	0	0	1	1	0	1	1
Reset confirmation	0	0	0	1	1	1	1	1
<i>Restart</i>								
Restart request	1	1	1	1	1	0	1	1
Restart confirmation	1	1	1	1	1	1	1	1

Note – A bit which is indicated as X may be set to either 0 to 1 as specified in the text and in Figures 5/X.75 to 20/X.75.

4.2 *xe ""§Call set-up and clearing packets*

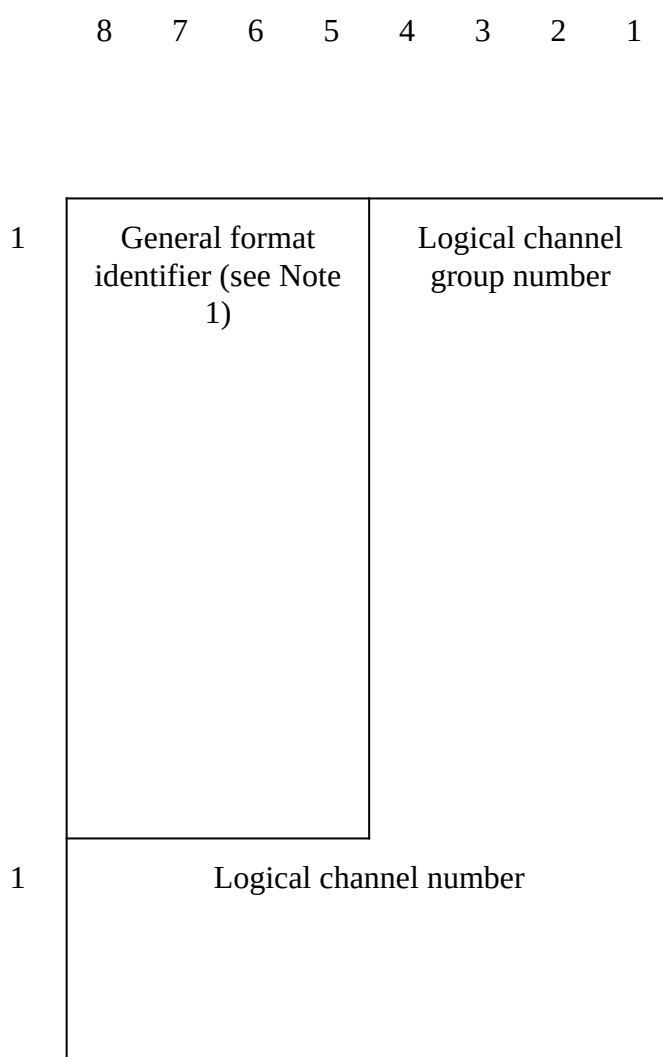
The following describes the nature of addresses present in the call set-up and clearing packets.

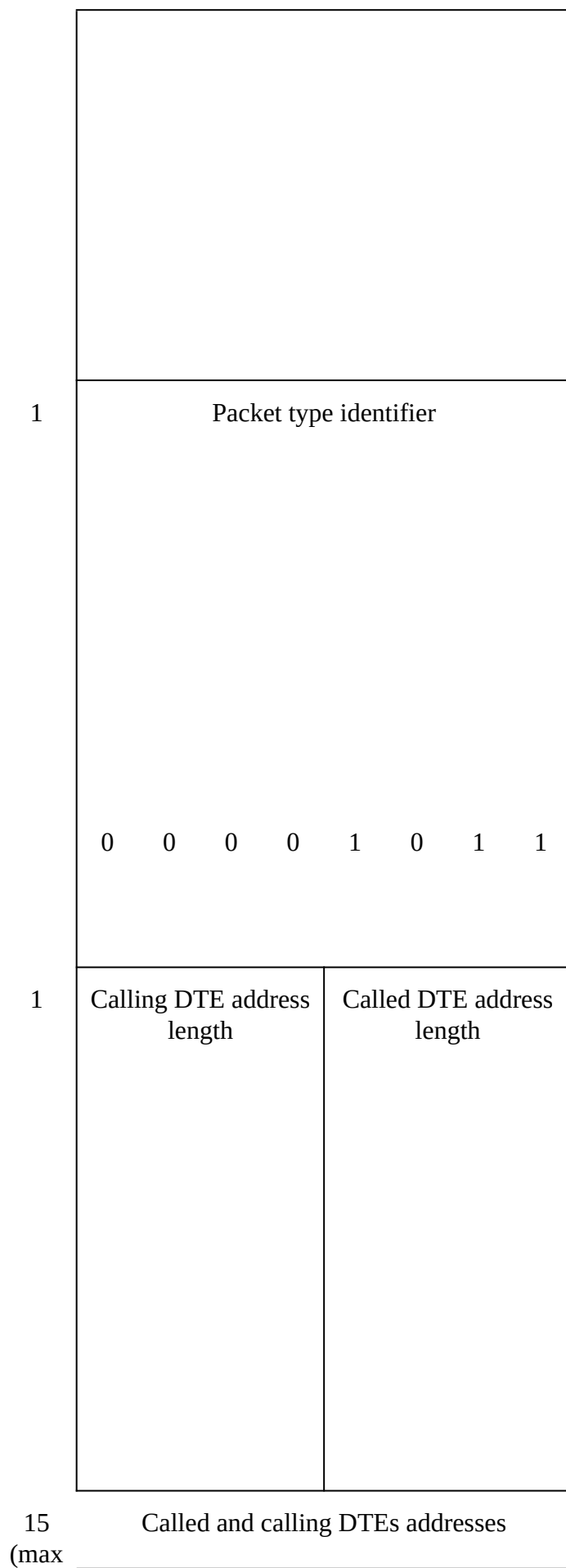
If the STE X/Y interface is between two PSPDNs, or between a PSPDN and an ISDN then the addresses will be in the international format given in Recommendation X.121, including escape digits where required. If the STE X/Y interface is between two ISDNs, then the addresses will be in the international format given in Recommendation E.164, including escape digits where required. Additional guidance is given in Recommendations X.31, X.122 and E.166.

4.2.1 *Call request packet*

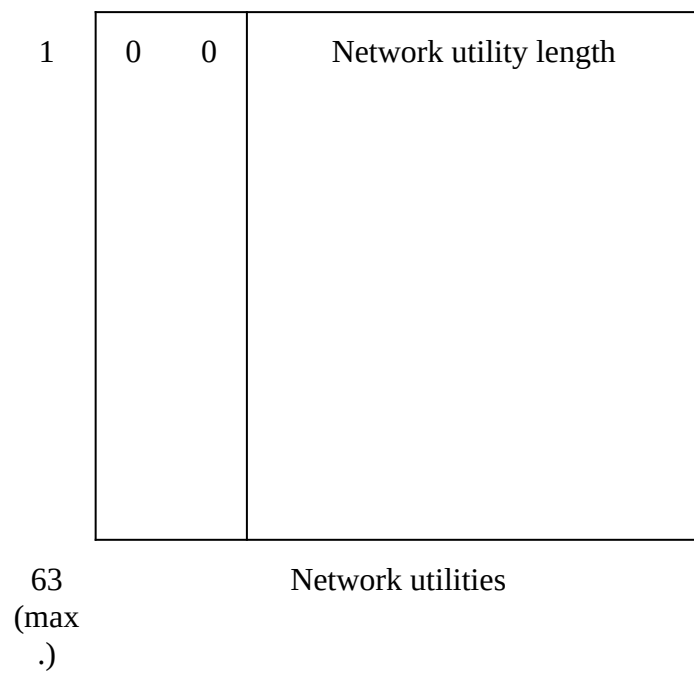
Figure 5/X.75 illustrates the format of a *call request* packet. In this figure the user facility length field, user facilities field, and call user data field are as defined in Recommendation X.25.

Bits

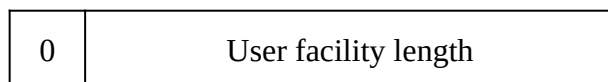


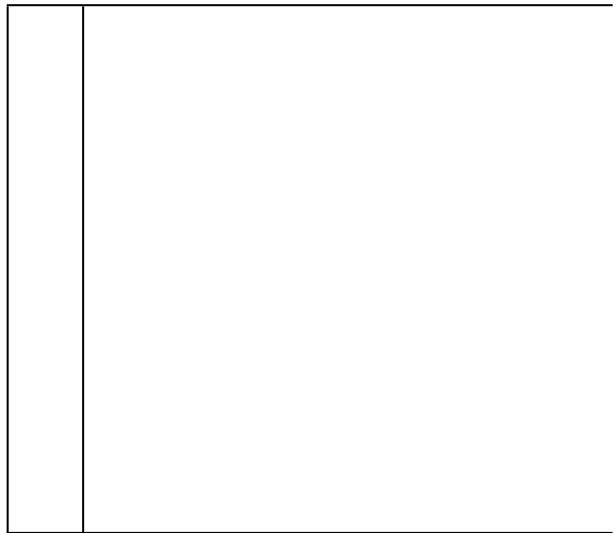


.)



1





109
(max
.)

User facilities

16
(max
.) or
128
(max
.)

Call user data (see Note 2)

Note 1 – Coded 0D01 (modulo 8) or 0D10 (modulo

128). D is the delivery confirmation bit.

Note 2 – More than 16 octets of call user data will only be present when the *fast select* optional user facility is requested.

FIGURE 5/X.75

Call request packet format

MONTAGE: Number of octets

4.2.1.1 *General format identifier*

Bit 7 can be set to either 0 or 1.

4.2.1.2 *xe ""\$Address length field*

Octet 4 consists of field length indicators for the called and calling DTE address. Bits 4, 3, 2 and 1 indicate the length of the called DTE address in semi-octets. Bits 8, 7, 6 and 5 indicate the length of the calling DTE address in semi-octets. Each address length indicator is binary coded and bit 1 or 5 is the low order bit of the indicator.

4.2.1.3 *xe ""\$Address field*

Octet 5 and the following octets consist of the called DTE address followed by the calling DTE address as specified in § 4.2 above.

Each digit of an address is coded in a semi-octet in binary coded decimal with bit 5 or 1 being the low order bit of the digit.

Starting from the high order digit, the address is coded in octet 5 and consecutive octets with two digits per octet. In each octet, the higher order digit is coded in bits 8, 7, 6 and 5.

The address field shall be rounded up to an integral number of octets by inserting 0s in bits 4, 3, 2 and 1 of the last octet of the field when necessary.

4.2.1.4 *xe ""\$Network utility length field*

Bits 6 through 1 of the octet following the address field indicate the length of the network utility field in octets.

The network utility length field indicator is binary coded and bit 1 is the low order bit.

Bits 8 and 7 of this octet are unassigned and set to 0.

4.2.1.5 *xe ""\$Network utility field*

The network utility field contains an integral number of octets. The length of this field depends on the utilities present. The maximum length of this field is 63 octets.

The coding of the network utility field is defined in § 5 below.

4.2.1.6 *xe ""\$User facility length field*

Bits 7 through 1 of the octet following the network utility field indicate the length of the user facility field in octets. The user facility length indicator is binary coded and bit 1 is the low order bit.

Bit 8 of this octet is set to 0.

4.2.1.7 *xe ""\$User facility field*

The user facility field contains an integral number of octets. The length of this field depends on the facilities present. The maximum length of this field is 109 octets. The coding of the user facility field is dependent on the facilities being requested as defined in Recommendation X.25 (Table 29/X.25 and Annex G/X.25).

4.2.1.8 *xe ""§Call user data field*

Following the user facility field, user data may be present. In the absence of the *fast select* optional user facility, the call user data field may contain any number of bits from 0 to 128 (16 octets). When the *fast select* optional facility is requested, the call user data may contain any number of bits from 0 to 1024 (128 octets). The contents of the field are passed unchanged.

Note – Some networks require the call user data field contains an integral number of octets (see § 3 Note).

4.2.2 *Call connected packet*

Figure 6/X.75 illustrates the format of a *call connected* packet. Similarly to the *call request* packet, the *call connected* packet contains:

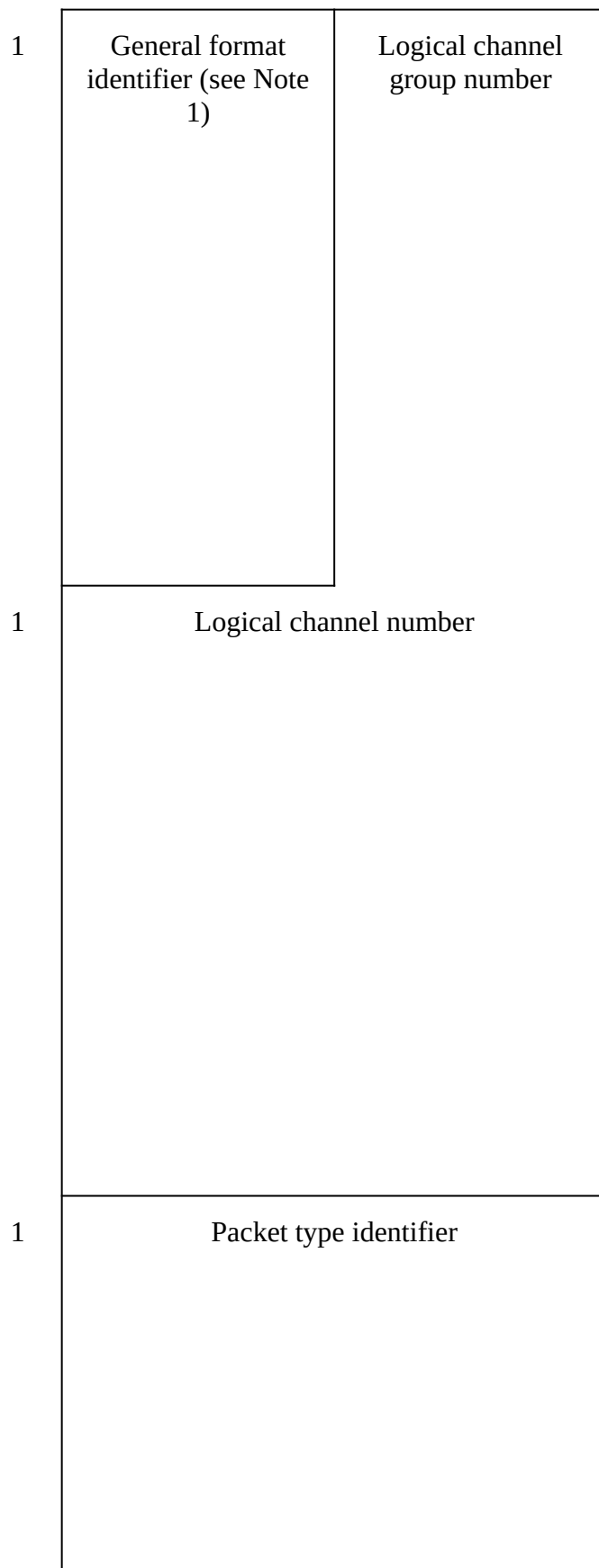
- an address length field,
- an address field,
- a network utility length field,
- a network utility field,
- a user facility length field,
- a user facility field, and
- a called user data field.

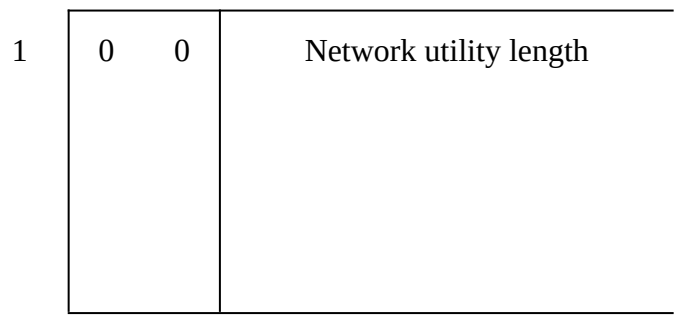
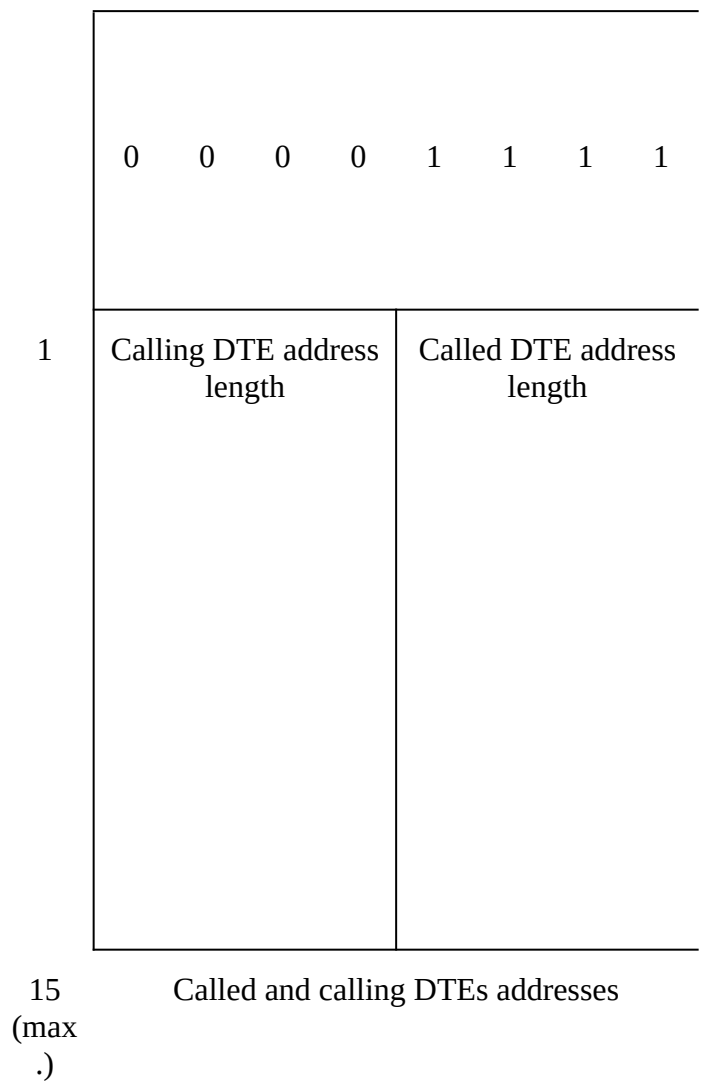
The coding of these fields is the same as that in *call request* packet (see § 4.2.1 above). Bit 7 of the general format identifier can be set to either 0 or 1. The address field may be empty. However, in the case of call redirection, the address field shall contain the address of the DTE to which the call was finally directed, and the utility field should contain the *called line address modified notification* utility (see § 5.3.10).

The called user data field may only be included for calls in which the *fast select* optional user facility has been requested with no restriction on response and may contain any number of bits from 0 up to 1024 (128 octets). The contents of the field are passed unchanged.

Bits

8 7 6 5 4 3 2 1





128 (max .)	Call user data (see Note 2)
-------------------	-----------------------------

Note 1 – Coded 0D01 (modulo 8) or 0D10 (modulo 128). D is the delivery confirmation bit.

Note 2 – This field will only be included where the called user data is returned in response to a *call request* packet in which the *fast select* optional user facility has been requested with no restriction on response.

FIGURE 6/X.75

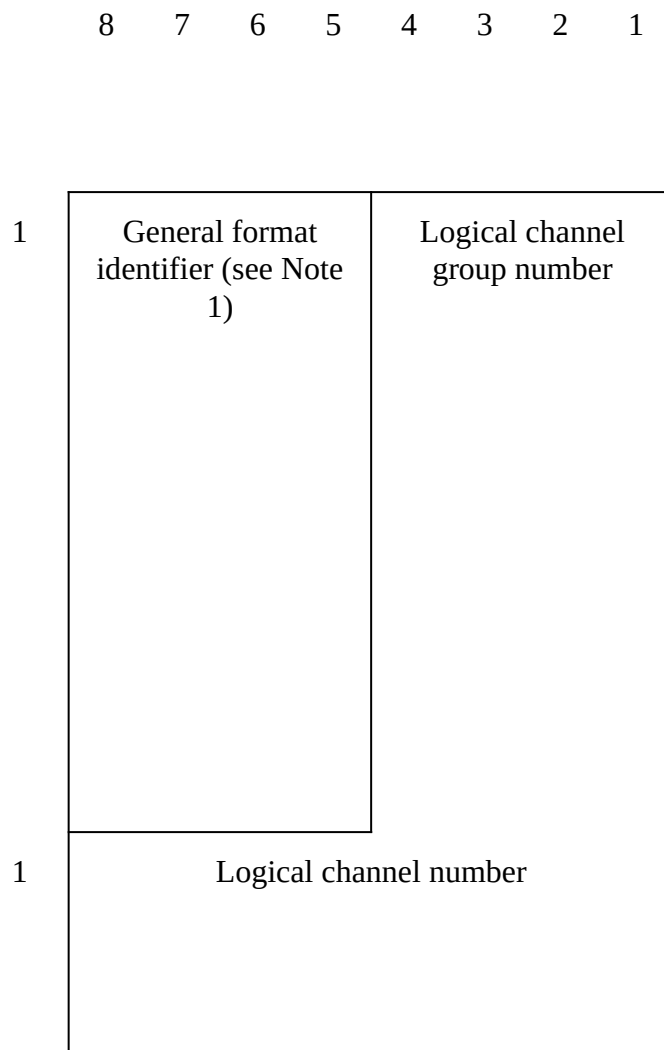
Call connected packet format

MONTAGE: Number of octets

4.2.3 Clear request packet

Figure 7/X.75 illustrates the format of a *clear request* packet.

Bits



1

Packet type identifier

0 0 0 1 0 0 1 1

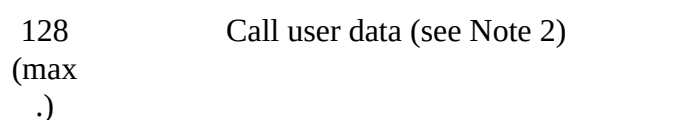
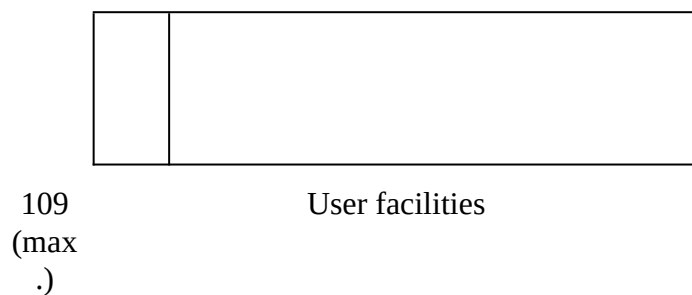
1

Clearing -cause

1	Diagnostic code	
	Calling DTE address length	Called DTE address length
15 (max .)	Called and calling DTEs addresses	

1	0	0	Network utility length
63 (max .)	Network utilities		

1	0	User facility length
---	---	----------------------



Note 1 – Coded 0001 (modulo 8) or 0010 (modulo 128).
Note 2 – This field will only be included where the clear user data is returned when *fast select* optional user facility has been requested.
Note 3 – Used only in the extended format (see § 4.2.3.3).
 FIGURE 7/X.75

Clear request packet format

MONTAGE: Number of octets (See Note 3)

4.2.3.1 Clearing cause field

Octet 4 is the clearing cause field and contains the reason for the clearing of the call.

The coding of the clearing cause field in a *clear request* packet is given in Table 13/X.75.

An STE receiving a clearing cause "clearing cause" other than that given in Table 13/X.75 will either pass this cause unchanged or change the cause to "Network congestion".

TABLE 13/X.75

Coding of clearing cause field in a clear request packet

Clearing cause	Octet 4 Bits							
	8	7	6	5	4	3	2	1
DTE originated	0	0	0	0	0	0	0	0
DTE originated (see Note 1)	1	X	X	X	X	X	X	X
Number busy	0	0	0	0	0	0	0	1

Out of order	0	0	0	0	1	0	0	1
Remote procedure error	0	0	0	1	0	0	0	1
Reverse charging acceptance not subscribed	0	0	0	1	1	0	0	1
Incompatible destination	0	0	1	0	0	0	0	1
Fast select acceptance not subscribed	0	0	1	0	1	0	0	1
Ship absent (see Note 2)	0	0	1	1	1	0	0	1
Invalid facility request	0	0	0	0	0	0	1	1
Access barred	0	0	0	0	1	0	1	1
Network congestion	0	0	0	0	0	1	0	1

Not obtainable	0	0	0	0	1	1	0	1
RPOA out of order (see Note 3)	0	0	0	1	0	1	0	1

Note 1 – When bit 8 is set to 1, the bits represented by Xs are those included by the remote DTE in the clearing or restarting clause field of the X.25 *clear* or *restart request* packet.

Note 2 – Used in conjunction with Mobile Maritime service.

Note 3 – May be received by the STE only if the optional *RPOA selection* utility was used by the STE.

4.2.3.2 *Diagnostic code field*

Octet 5 is the diagnostic code field and may contain additional information on the reason for the clearing of the call.

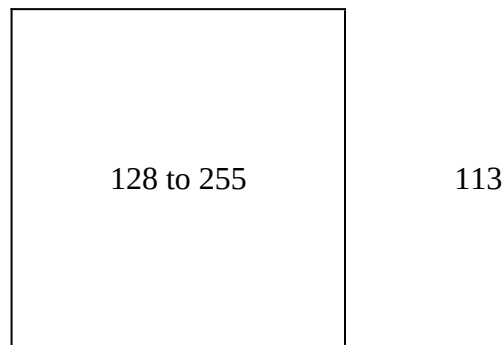
If the associated clearing cause field (octet 4) indicates any valid cause (see Table 13/X.75) except “Network congestion”, the contents of this field will be passed unchanged. If the clearing cause field indicates “Network congestion” and the original clear or restart request was generated as the result of an event detected other than at the local STE–X/Y interface, then the value of the diagnostic code passed will be as shown in Table 14/X.75.

The diagnostic codes in *clear request* packets generated as the result of event detected at the local STE–X/Y interface are listed in Annex E.

TABLE 14/X.75

Diagnostic code mapping for clear request packet

Decimal value originally generated	Decimal value passed
0	same
1 to 111	114
112 to 127	same



4.2.3.3 *Extended format*

The following fields may follow the diagnostic code field in the extended format:

- an address length field,
- an address field,
- a network utility length field,
- a network utility field,
- a user facility length field,
- a user facility field, and
- a clear user data field.

4.2.3.3.1 *Address length field*

This single octet field consists of field length indicators for the called and calling DTE addresses. Bits 4, 3, 2 and 1 indicate the length of the called DTE address in semi-octets. Bits 8, 7, 6 and 5 indicate the length of the calling DTE address in semi-octets. Each address length indicator is binary coded and bit 1 or 5 is the low order bit of the indicator.

The address length field is always present when the network utility length field is present.

4.2.3.3.2 *Address field*

In the case that the clear request is issued, by a DTE to which a call has been redirected, as a direct response to the *call request* packet the address shall contain the address of the DTE to which the call was finally directed. Other use of this field is for further study.

Note – In the case of call redirection or call distribution within a hunt group, the utility field of *clear request* packet should include the *called line address modified notification* utility (see § 5.3.10).

4.2.3.3.3 *Network utility length field*

Bits 6 through 1 of the octet following the address field indicate the length of the network utility field in octets.

The network utility length field is binary coded and bit 1 is the low order bit.

Bits 8 and 7 of this octet is set to 0.

The network utility length field is always present when the user facility length is present.

4.2.3.3.4 *Network utility field*

The network utility field contains an integral number of octets. The length of this field depends on the utilities present. The maximum length of the field is 63 octets.

The coding of the network utility field is defined in § 5 below.

4.2.3.3.5 *User facility length field*

Bits 7 through 1 of the octet following the network utility field indicate the length of the user facility field in octets. The user facility length indicator is binary coded and bit 1 is the low order bit of the indicator.

Bit 8 of this octet is set to 0.

The user facility length field is always present when the user data field is present.

4.2.3.3.6 *User facility field*

The user facility field contains an integral number of octets. The length of this field depends on the facilities present. The maximum length of this field is 109 octets. The coding of the user facility field is dependent on the facilities being requested as defined in Recommendation X.25 (Table 29/X.25 and Annex G/X.25).

4.2.3.3.7 *Clear user data field*

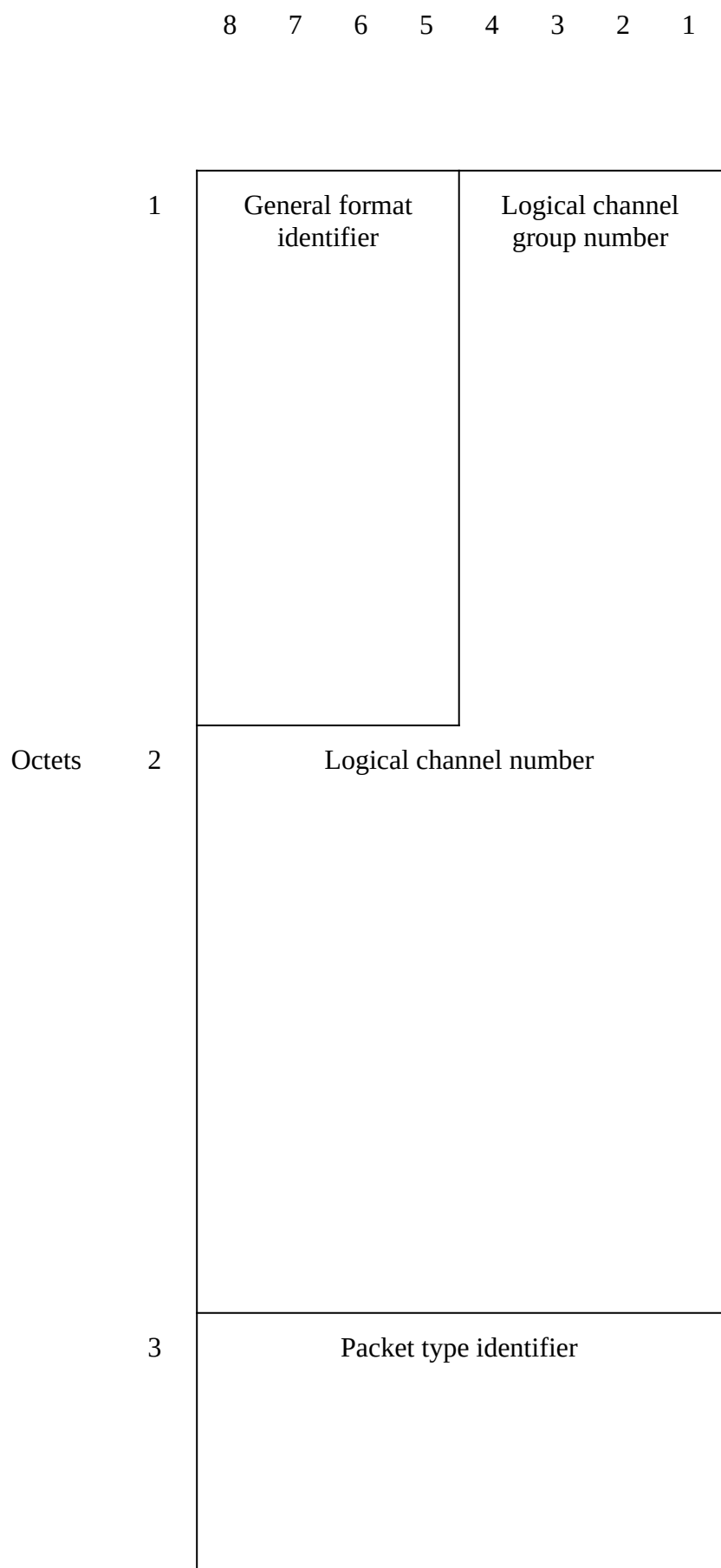
For calls in which the *fast select* optional user facility has been requested, clear user data may be present, following the user facility field. The clear user data field may contain any number of bits from 0 up to 1024 (128 octets). The contents of the field are passed unchanged.

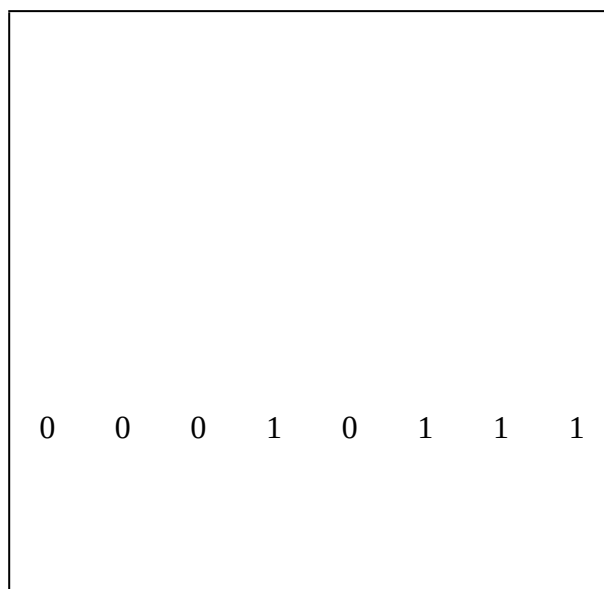
Note – Some networks require the clear user data field to contain an integral number of octets (see § 3 Note).

4.2.4 *xe ""§Clear confirmation packet*

Figure 8/X.75 illustrates the format of the *clear confirmation* packet.

Bits





Note – Coded 0001 (modulo 8) or 0010 (modulo 128).
FIGURE 8/X.75

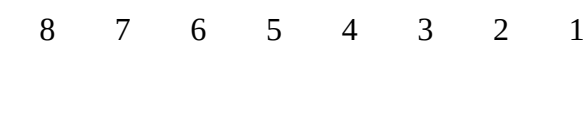
Clear confirmation packet format

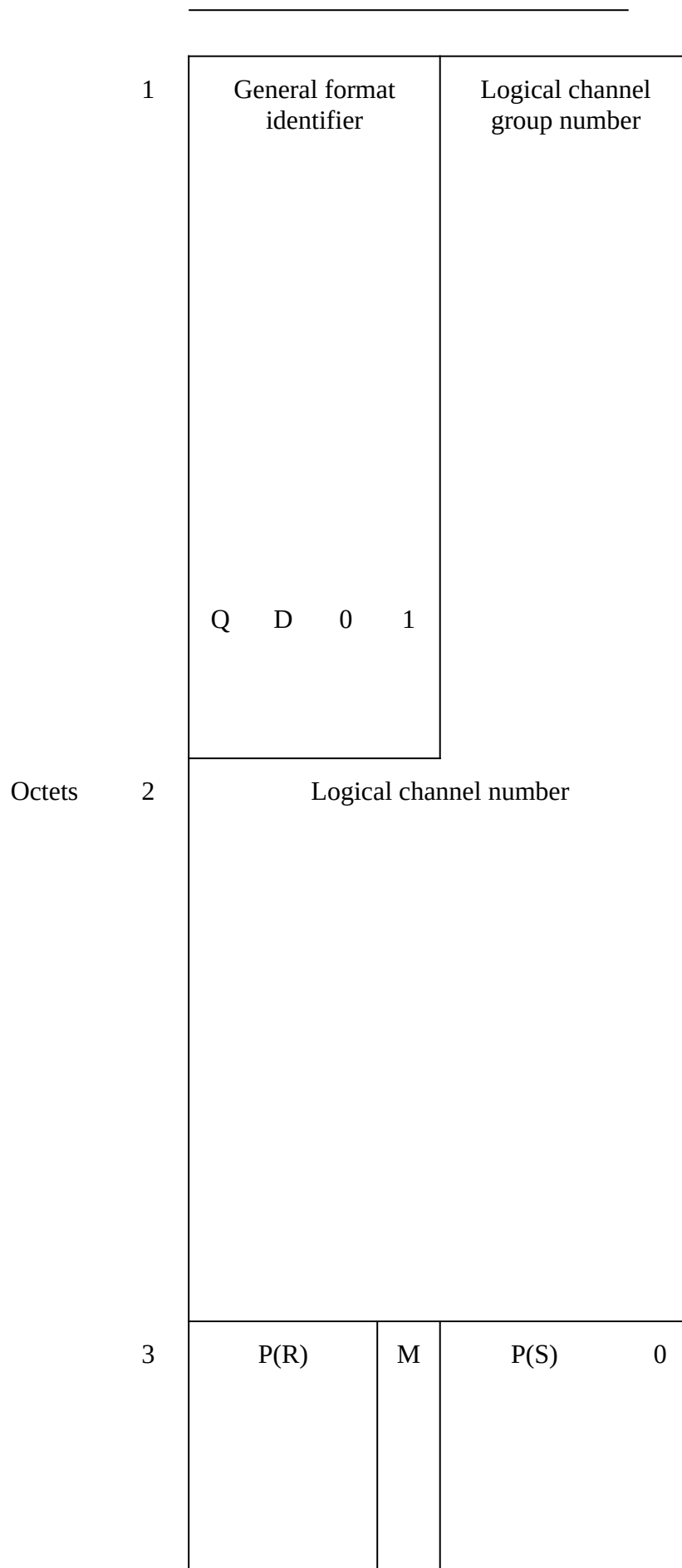
4.3 *Data and interrupt packets*

4.3.1 *Data packet*

Figures 9/X.75 and 10/X.75 illustrate the format of the *data* packets in the case of modulo 8 and modulo 128 respectively.

Bits





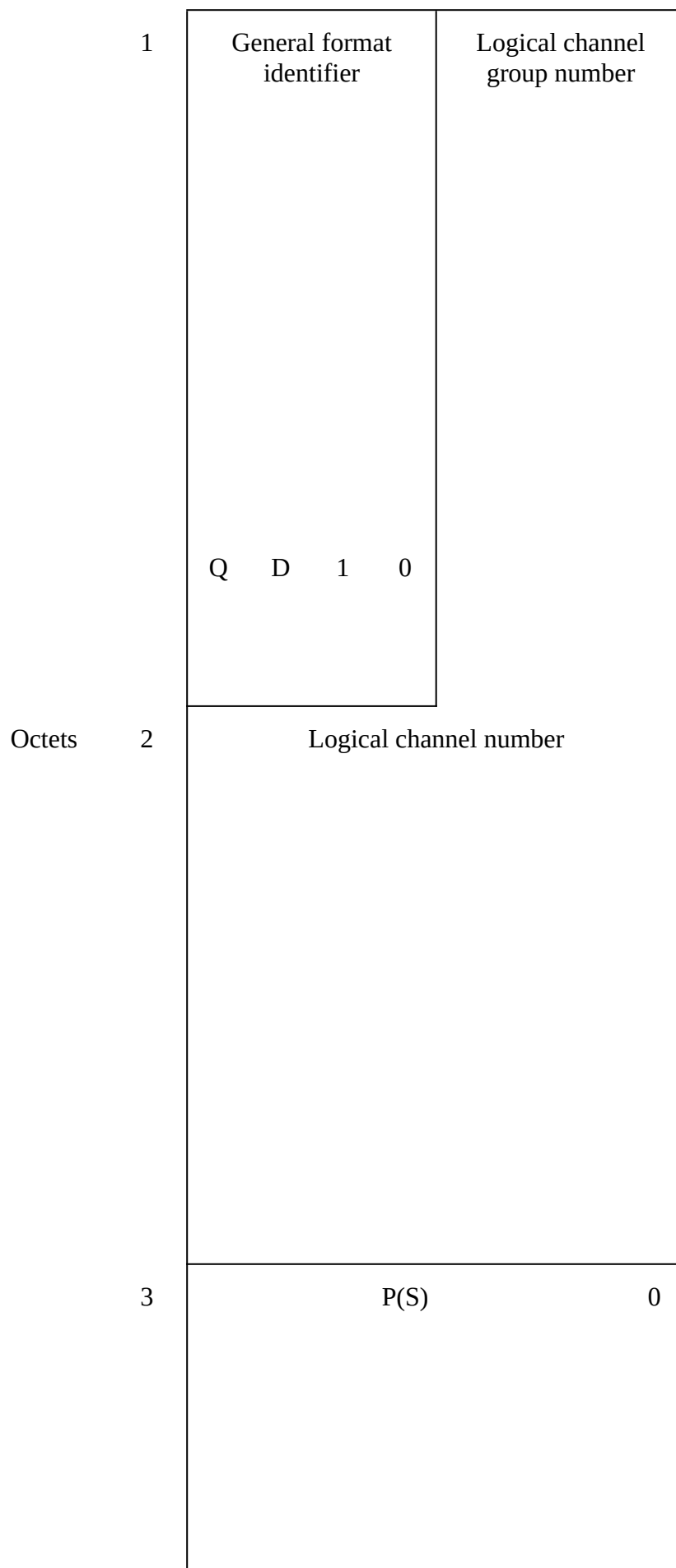
User data

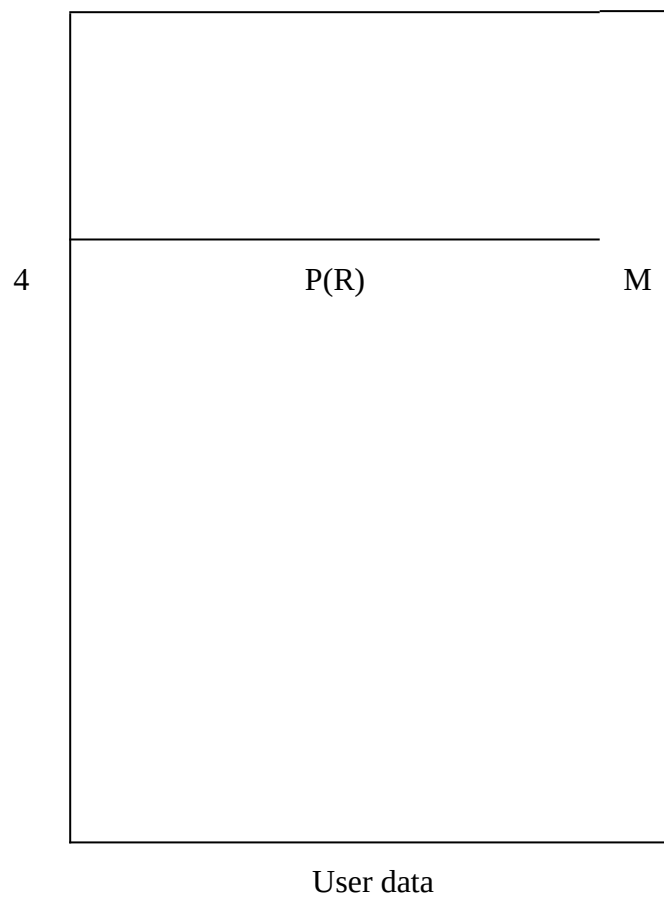
D Delivery confirmation
M More data indication
Q Qualifier
FIGURE 9/X.75

Data packet format (modulo 8)

Bits

8 7 6 5 4 3 2 1





D Delivery confirmation
M More data bit
Q Qualifier bit
FIGURE 10/X.75

Data packet format (modulo 128)

4.3.1.1 *xe ""§Qualifier (Q) bit*

Bit 8 in octet 1 is used for the *qualifier* (Q) bit.

4.3.1.2 *xe ""§Delivery confirmation (D) bit*

Bit 7 in octet 1 is the *delivery confirmation* (D) bit.

4.3.1.3 *xe ""§Packet receive sequence number*

In Figure 9/X.75 bits 8, 7 and 6 of the octet 3 are used for indicating the packet receive sequence number P(R). P(R) is binary coded and bit 6 is the low order bit. In Figure 10/X.75, bits 2 through 8 of octet 4 are used for the packet send sequence number and bit 2 is the low order bit.

4.3.1.4 *xe ""§More data bit*

In Figure 9/X.75, bit 5 in octet 3 is used for the *more data* mark (M bit). In Figure 10/X.75, bit 1 in octet 4 is used for the *more data* mark (M bit) (0 for no more data and 1 for more data).

4.3.1.5 *xe ""§Packet send sequence number*

In Figure 9/X.75, bits 4, 3 and 2 of octet 3 are used for indicating the packet send sequence number P(S). P(S) is binary coded and bit 2 is the low order bit. In Figure 10/X.75, bits 2 through 8 of octet 3 are used for the packet send sequence number and bit 2 is the low order bit.

4.3.1.6 *User data field*

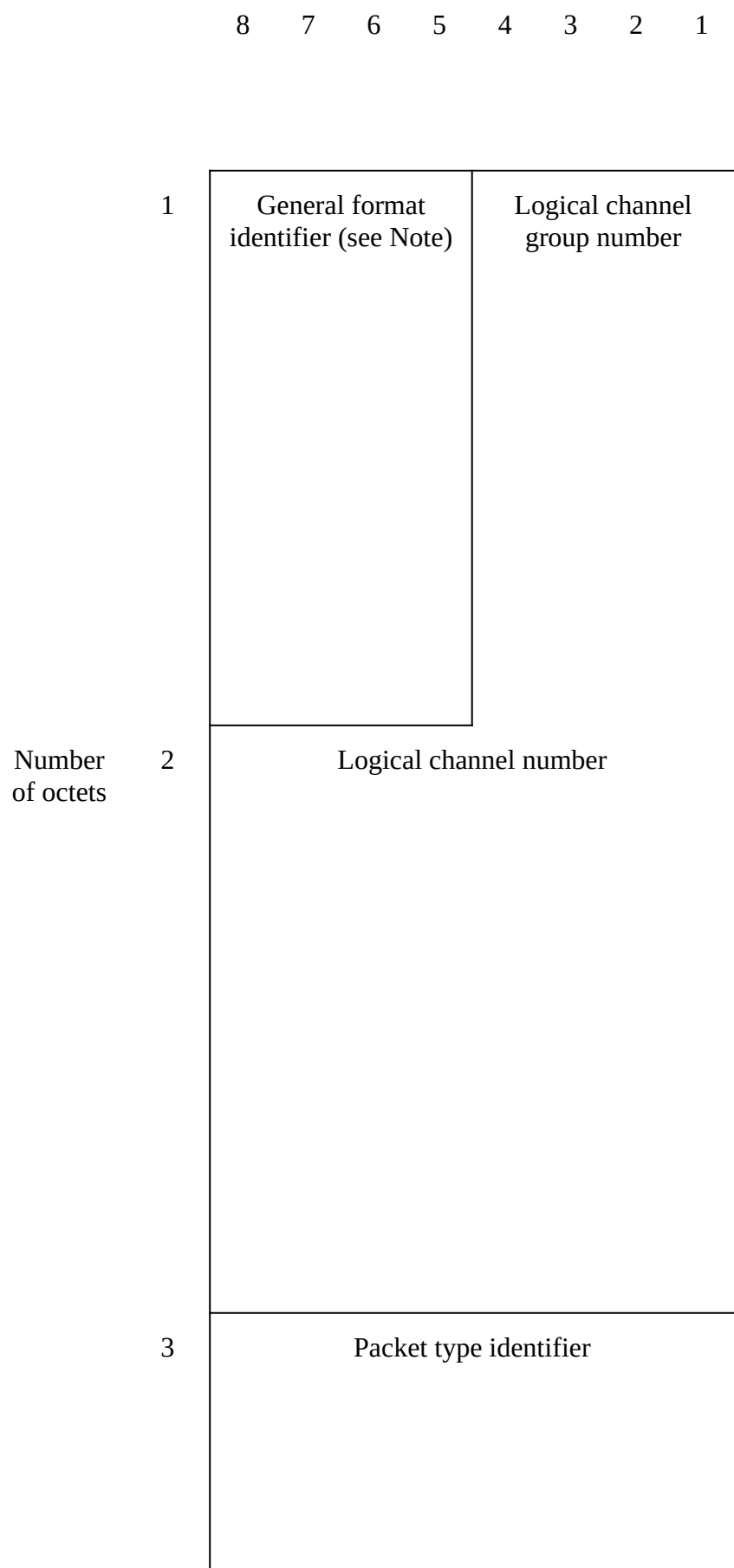
The bits following octet 3 (modulo 8) or octet 4 (modulo 128) contain user data.

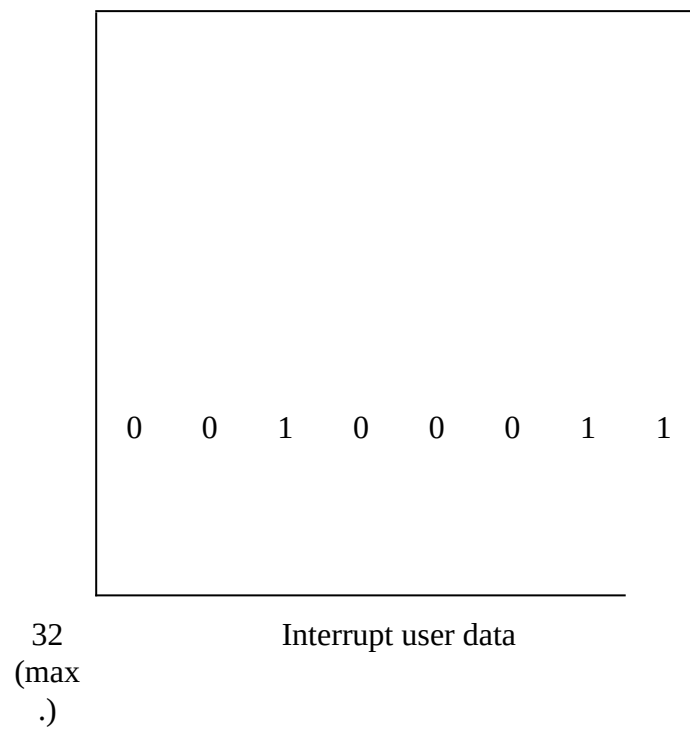
Note – Some networks require the user data field to contain an integral number of octets (see § 3 Note).

4.3.2 *xe ""§Interrupt packet*

Figure 11/X.75 illustrates the format of the *interrupt* packet.

Bits





Note – Coded 0001 (modulo 8) or 0010 (modulo 128).
FIGURE 11/X.75

Interrupt packet format

