INTERNATIONAL STANDARD

ISO/IEC 13871

First edition
1995-12-15

Information technology — Telecommunications and information exchange between systems — Private telecommunications networks — Digital channel aggregation

Technologies de l'information — Télécommunications et échange d'information entre systèmes — Réseaux privés de télécommunications — Agrégation de canal numérique

Reference number
ISO/IEC 13871:1995(E)

Contents **page**

Foreword iv

Introduction v

1 Scope 1

2 Normative References 2

3 Definitions 2

4 Symbols and Abbreviations 3

5 Selection of ISO/IEC 13871 or ITU-T H.244 aggregation procedures 3

6 Basic Principles 5

6.1 Modes of operation 5
6.2 Frame Structure 6
6.2.1 Bit Ordering 7
6.2.2 Frame Alignment Word (FAW) 7
6.2.3 Frame Count (FC) 7
6.2.4 Information Channel (IC) 7
6.2.5 Cyclic Redundancy Check (CRC) 8
6.2.6 Distribution of Overhead Octets 8
6.3 Overall Frame alignment 9
6.4 Description of CRC4 procedure 10
6.4.1 Computation of the CRC4 bits 10
6.4.1.1 Multiplication and division process 10
6.4.1.2 Encoding procedure 10
6.4.1.3 Decoding procedure 10
6.4.2 Consequent actions 10
6.4.2.1 Enabling and Disabling the CRC procedures 10
6.4.2.2 Action on bit E 11
6.4.2.3 Monitoring for error performance 11

7 Description of the Information Channel 11

7.1 Information Channel Frame 11
7.1.1 Rate Multiplier and Subrate Multiplier Calculation and Operation 15
7.2 Information Channel (IC) synchronization 19
7.3 Transmission and Recognition of an Information Channel Frame 19

8 Procedures 20

8.1 Call Setup 20
8.1.1 Initial Channel Setup 21
8.1.2 Additional Channel Setup 23

8.1.3 Delay Equalization 25
8.1.4 Timeout Conditions 25
8.2 Adding Bandwidth to an Existing Call 26
8.2.1 Negotiation (Calling Endpoint Initiated) 26
8.2.2 Negotiation (Answering Endpoint Initiated) 26
8.2.3 Additional Channel Setup 27
8.2.4 Timeout Conditions 28
8.3 Deleting Bandwidth from an Existing Call 28
8.3.1 Calling Endpoint Initiated 29
8.3.2 Answering Endpoint Initiated 30
8.3.3 Timeout Conditions 30
8.4 Collision Resolution 30
8.5 Call Disconnection 30
8.5.1 Calling Endpoint Initiated 30
8.5.2 Answering Endpoint Initiated 30
8.5.3 Timeout Conditions 31
8.6 Remote Loopback 31
8.6.1 Timeout Conditions 32
8.7 Error Conditions 32
8.7.1 Loss of Channel (Other Than Call Setup) 32
8.7.2 Loss of Delay Equalization 33

9 Transparent Mode Operation 33

9.1 Non Aggregating Unit Calling Aggregating Unit 33
9.2 Aggregating Unit Calling Non Aggregating Unit 34

Annexes

A RMULT and SUBMULT Values 35

A.1 Bearer Channel Rate (BCR) = 56 kbit/s 35
A.2 Bearer Channel Rate (BCR)= 64 kbit/s 37

B Manufacturer's Identifiers 39

C Phone Number Length Extension 40

D State Machine Description of Channel Aggregation Control 41

D.1 Reference Architecture 41
D.2 State Definitions 42
D.2.1 DEQ Negotiation Control 42
D.2.2 DEQ MULTIFRAME CONTROL States 44
D.3 Messages and Events 44
D.3.1 Information Channel Messages 45
D.3.2 DEQ Negotiation Control to Call Control Messages 46
D.3.3 DEQ Negotiation Control to DEQ Multiframe Control Primitives 50
D.4.Timers 53
D.5 State Transition Diagrams 54
D.5.1 DEQ Negotiation Control 54
D.5.2 DEQ Multiframe Control 58
D.6. SDL Diagrams 59
D.6.1 DEQ Negotiation Control 60
D.6.2 DEQ Multiframe Control 87

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

International Standard ISO/IEC 13871 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 6, *Telecommunications and information exchange between systems*.

Annexes A to C form an integral part of this International Standard. Annex D is for information only.

Introduction

This International Standard is one of a set of International Standards designed to facilitate the global interconnection of applications across both private telecommunications networks and public ISDNs. This is achieved by ensuring compatibility and interworking of services offered by the private and public networks.

In line with the above aims, this International Standard defines a service offering flexible bandwidth allocation that provides unrestricted information transfer between special terminal adapters called Channel Aggregation Units. A Channel Aggregation Unit can provide a high bandwidth capability to an attached terminal application by combining multiple 56 kbit/s or 64 kbit/s digital communications channels that are available across public and/or private networks. It also ensures that bit sequence integrity of the aggregated bandwidth is maintained across the network(s). The number of individual communications channels aggregated may range from one up to the practical limits imposed by the Channel Aggregation Unit.

The service defined in this International Standard can be used for a number of applications, including

- o LAN interconnection
- o image transfer
- o bulk file transfer
- o video conferencing.

ITU-T Recommendation H.244 defines the means by which the appropriate choice of channel aggregation procedures may be made, in accordance with the requirements of the application being served. This choice is made from two available sets of procedures, one of which is also defined by ITU-T Recommendation H.244, the other of which is defined in this International Standard.

This page intentionally left blank

Information technology - Telecommunications and information exchange between systems - Private telecommunications networks - Digital channel aggregation

1 Scope

This International Standard defines a set of procedures collectively called 'Digital Channel Aggregation', which are used in the provision of an aggregated bearer service. Digital channel aggregation involves a means for the combination of multiple switched or unswitched 56 kbit/s or 64 kbit/s digital communications channels across one or more public and/or private networks into higher bandwidth digital bi-directional channels between Channel Aggregation Units (CAUs) serving terminal applications at network endpoints. Figure 1 below illustrates a typical channel aggregation scenario.

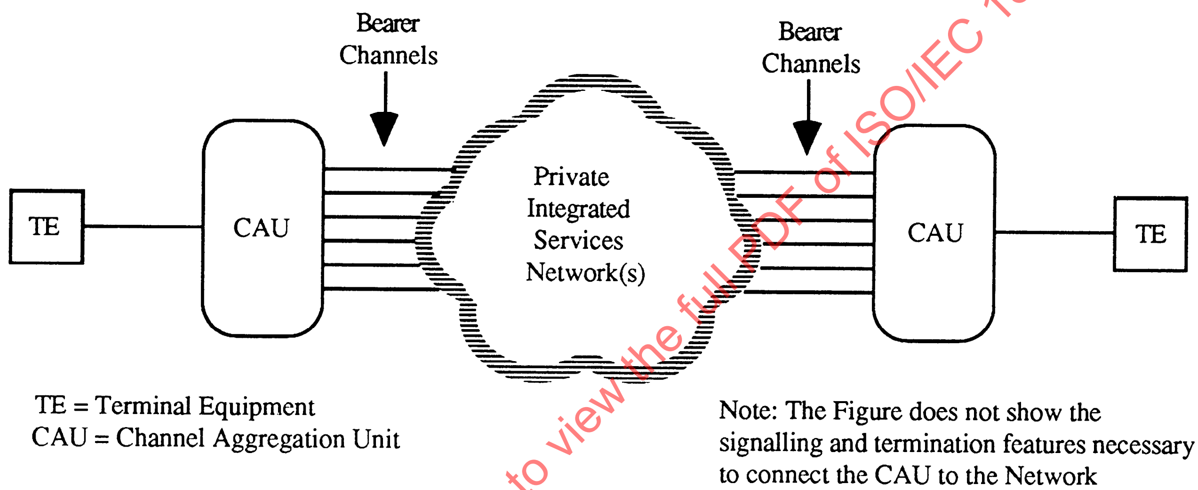


Figure 1 - Channel Aggregation Scenario

In defining the channel aggregation procedures, this International Standard sets out the means used by a CAU to

- o establish the parameters to be used for the aggregated connection
- o synchronise and align multiple communications channels to maintain the bit sequence integrity of the aggregated bandwidth across the network(s)
- o optionally monitor data transfer throughout a call to detect failure modes
- o institute failure recovery procedures
- o optionally dynamically vary the bandwidth on demand during a call.

The channel aggregation procedures above are defined at the interface between the CAU and the network in terms of the frame structure and information messages applied to individual communications channels. Not defined by this International Standard are

- o the call control procedures used to establish and disconnect the individual communications channels that make up the higher bandwidth connections
- o the higher level protocols or applications that use the channel aggregation services
- o the frame structure on the individual communications channels at the physical interface between the CAU and the network as required for 'normal' 56 kbit/s or 64 kbit/s bearer services as opposed to aggregated bearer services
- o the electrical characteristics of the individual communications channels at the physical interface between the CAU and the network.

2 Normative References

The following standards contain provisions which, through reference in this text, constitute provisions of this International Standard. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this International Standard are encouraged to investigate the possibility of applying the most recent editions of the standards indicated below. Members of IEC and ISO maintain registers of currently valid International Standards.

ISO/IEC 11571:1994, *Information technology - Telecommunications and information exchange between systems - Numbering and sub-addressing in private integrated services networks.*
ITU-T Recommendation G.704 (1991), *Synchronous frame structures used at primary and secondary hierarchical levels.*
ITU-T Recommendation H.221 (1994), *Frame structure for a 64 to 1920 kbit/s channel in audiovisual teleservices.*
ITU-T Draft Recommendation H.244 (1995), *Synchronized aggregation of ISDN-B channels.*
ITU-T Recommendation H.320 (1994), *Narrow-band visual telephone systems and terminal equipment.*
ITU-T Recommendation I.112 (1992), *Vocabulary of terms for ISDNs.*
ITU-T Recommendation T.35 (1991), *Procedure for the allocation of CCITT defined codes for non-standard facilities.*

3 Definitions

For the purposes of this International Standard, the following definitions apply.

3.1 Answering Endpoint: The endpoint that receives the call.

3.1 Bearer Channel: The network provided channel used to carry the data. The framing structure is overlaid on the Bearer Channel to provide the data transport and inverse multiplexing capability. Examples of Bearer Channels are:

- o BRI or PRI B-channels
- o T1 DS0s
- o Switched 56/64 kbit/s connections

3.3 Bearer Channel Rate: The data rate at which the Bearer Channel is clocked.

3.4 Calling Endpoint: The endpoint that initiates the call.

3.5 Channel Aggregation Unit: Any piece of equipment adhering to this International Standard. Note that any reference to other forms of channel aggregation (e.g., H.244) will be stated explicitly.

3.6 CRC Encoder: The function that calculates the CRC for inclusion in the transmitting direction.

3.7 Information Message: The 16 octet message sent and received over the Information Channel in the multiframe structure or over the entire bandwidth of the Master Channel during initial parameter negotiation. This message is transmitted continuously. In this International Standard, the use of the term Information Message, when discussing receipt of a message, indicates a message which has been recognized after the use of some form of detection processing (e.g., 2 out of 3 majority voting).

3.8 Master Channel: The channel used to communicate control information between each endpoint. This includes the initial parameter negotiation for call setup, channel delete, channel addition, remote loopback, call disconnection, etc. See 8.1.1 for further information.

3.9 Overall System Delay: The end-to-end transit delay (length of time it takes for the data to get from the transmitter to the receiver) including following:

- o Delay introduced by the transmitting endpoint (equipment implementing this algorithm).
- o Delay introduced by the network connecting the two endpoints.
- o Delay introduced by the receiving endpoint (equipment implementing this algorithm).

4 Symbols and Abbreviations

ALIGN	-	Alignment Octet
AU	-	Answering Unit
AVT	-	Audio Visual Terminal
BCR	-	Bearer Channel Rate
CAU	-	Channel Aggregation Unit
CID	-	Channel Identifier
CRC	-	Cyclic Redundancy Check
CSU	-	Channel Service Unit
CU	-	Calling Unit
DEQ	-	Delay Equalisation
DN	-	Directory Number
DS0	-	Transmission Standard for bit rate of 64 kbit/s
DS1	-	Transmission Standard for bit rate of 1.544 Mbit/s
DSU	-	Digital Service Unit
EBR	-	Effective Bearer Rate
FAW	-	Frame Alignment Word
FC	-	Frame Count
GID	-	Group Identifier
IC	-	Information Channel
ISO	-	International Organisation for Standardisation
ISDN	-	Integrated Services Digital Network
MF	-	Multiframe
MFG	-	Manufacturing Identifier
PISN	-	Private Integrated Services Network
PMDL	-	Physical Media dependent Layer
PTN	-	Private Telecommunications Network
Res	-	Reserved Bits
REV	-	Revision Level
RI	-	Remote Indicator
RLIND	-	Remote Loopback Indication
RLREQ -	Remote Loopback Request	
RMULT-	Rate Multiplier	
SI	-	Stream Identifier
SUBMULT	-	Subrate Multiplier
T1	-	Transmission Standard for bit rate of 1.544 Mbit/s
TAa	-	Timer in Answering Endpoint - timer a
TCa	-	Timer in Calling Endpoint - timer a
TXa	-	Timer in Either Endpoint - timer a
XFLAG -	Transfer Flag	

5 Selection of ISO/IEC 13871 or ITU-T H.244 aggregation procedures

ITU-T Recommendation H.244 defines both a set of channel aggregation procedures and a means for determining the correct choice of procedures, i.e. either ISO/IEC 13871 or ITU-T H.244, according to the application being served.

A channel aggregation unit which is equipped to operate in both ISO/IEC 13871 and ITU-T H.244 aggregation modes shall select the correct mode of operation according to the process depicted in Figure 2 (reproduced from Figure 9/H.244).

If the CAU is not preset to ISO/IEC 13871 (Condition 1), and if H.221 framing is detected on the single-channel side (Condition 2), and if this framed signal includes either {SM-comp} or {6B-H0-comp} in its capset (Condition 3) - see H.244 - then the CAU shall commence transmissionas for H.244 aggregation, and shall continue this if H.221 framing is also received on the master channel from theremote CAU. However, if the signal from the remote CAU is according to 7.1.1 of this International Standard, then transmission shall immediately be switched to be likewise in accordance with 7.1.1.

If any of the Conditions 1, 2, 3 are not met, then the CAU shall commence transmission as set out in 7.1.1.

In the following, it is assumed that Conditions 1, 2, 3 above have not been met and that the CAU has commenced transmission in ISO/IEC 13871 mode.

NOTE -The significance of the test point (4) in Figure 2 is as follows; if the incoming signal is H.221 framed but neither {SM-comp} nor {6B-H0-comp} is included in the capability set, then the remote end-point is a self-aggregating type of H.320 terminal and ISO/IEC 13871 aggregation is not possible.

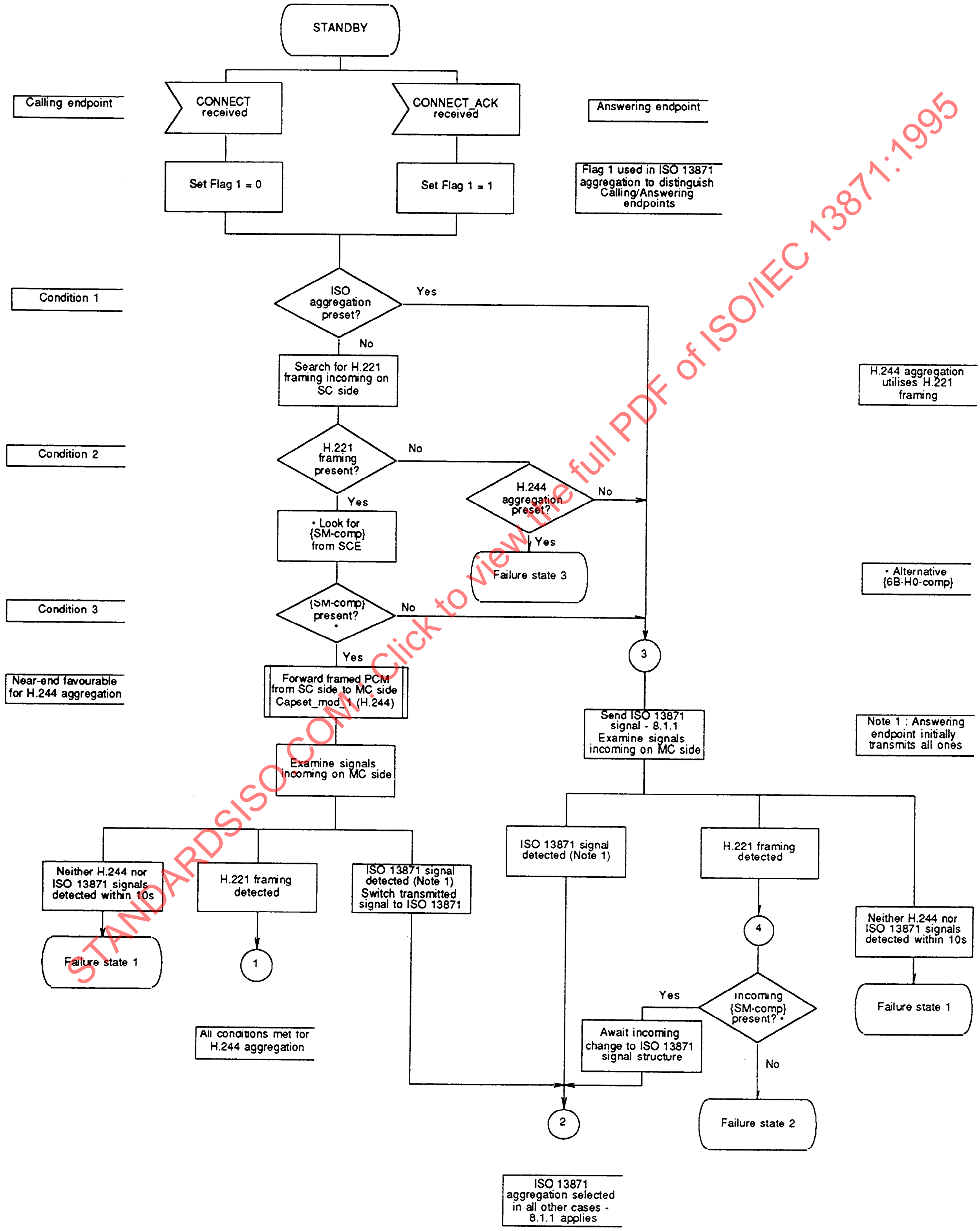


Figure 2 - Selection of ISO/IEC 13871 or ITU-T H.244 aggregation

6 Basic Principles

This International Standard describes a means for providing a high speed serial data stream by use of multiple, independent 56/64 kbit/s channels. These channels are individually connected over the switched digital network (public, private, etc.). At the transmitting end, user data is placed in a framing structure in each bearer channel and transmitted over multiple, independent channels. At the receiving end, all channels are phase aligned and synchronized (using the framing structure) to recreate the original data stream. This framing and synchronization is transparent to the attached application.

Due to the network routing the channels used for the wideband connection independently of each other, the data in each channel might be individually delayed relative to the data in other channels. The frame structure described herein for each 56/64 kbit/s bearer channel provides for the alignment of data octets from the individual channels to their original sequence before reforming the individual channels into a composite serial data stream at the terminating end. Overall transit delay for the end-to-end connection is equal to the longest transit delay from all the channels plus a constant delay due to the frame alignment that depends on implementation.

Once aligned, data transfer may be constantly monitored throughout the call. The failure of a channel, for reasons such as call disconnection, phase slip or high error rate, may be automatically detected. Various fault isolation and recovery procedures are defined in response to these scenarios.

Individual switched 56/64 calls can be combined to form a single transparent N x 56/64 channel.

NOTES

1 This algorithm will maintain bit and octet sequence integrity in that it will deliver user bits and octets to the remote endpoint in the same order and timing relationships as were received on the originating endpoint (except in cases of lost frame alignment). For applications that require octet timing to be maintained across the network (i.e., an octet of user data clocked into the transmitting endpoint must be clocked out as an octet of user data at the receiving endpoint), it is recommended that the first octet of user data transmitted across the network (i.e., the first octet transmitted after delay equalization is achieved) be placed in the first octet of a Channel Aggregation frame and not split across two octets of a Channel Aggregation frame. It is also recommended that only Modes 1, or 3 restricted to multiples of 64 kbit/s be used for this purpose (otherwise octet timing may be lost).

2 Although currently a maximum of 63 channels is specified, this limit is for further study.

6.1 Modes of operation

Five modes of operation are supported. The following two modes are required:

Mode 1:

This mode supports user data rates that are multiples of the bearer rate. It provides the user data rate with the full available bandwidth, but does **not** provide an in-band monitoring function. The overhead octets are removed after the call is phase aligned. Error conditions on one or more channels that disturb overall system synchronization are not recognized automatically after the call is in active state. Recovery from these error conditions during the active state requires manual or external intervention. This intervention is outside the scope of these procedures.

Transparent Mode:

In this mode, incoming (outgoing) channels are "cut-through" to the applications served by the Channel Aggregation Unit. The Channel Aggregation Unit performs no delay equalization or parameter negotiation. This mode is useful when delay equalization is performed by some other means (or is not required) and one endpoint is **not** a Channel Aggregation Unit.

A Channel Aggregation Unit implementing this International Standard **shall** implement Mode 1 and the Transparent Mode. Thus, Mode 1 and the Transparent Mode **shall** be the common modes of operation for all implementations claiming compatibility with this International Standard. Clause 9 describes the Transparent Mode in detail.

The following three modes of operation are optional.

Mode 0:

This mode provides initial parameter negotiation and Directory Number exchange over the master channel, then enters transparent mode without delay equalization. It does **not** provide an inband monitoring function. This mode is useful when the calling endpoint requires Directory Numbers, but the delay equalization is performed by some other means (e.g., attached video codec).

Mode 2:

This mode supports user data rates that are multiples of 63/64 of the bearer rate. An in-band monitor function provides a continuous check for delay equalization and an end-to-end bit error rate test. (Error rate testing is accomplished by performing a cyclic redundancy check calculation on an octet sequence before transmission and testing the same sequence for errors on the receive end.) The user data rate is that bandwidth remaining after the insertion of overhead octets (i.e., 98.4375% or 63/64 of the total network bandwidth).

Mode 3:

This mode supports user data rates that are integral multiples of 8 kbit/s, including N x 56 and N x 64 kbit/sec. All channels use the same bearer channel rate. An in-band monitor function provides a continuous check for delay equalization and an end-to-end bit error rate test. (Error rate testing is accomplished by performing a cyclic redundancy check calculation on an octet sequence before transmission and testing the same sequence for errors on the receive end.) The overhead octets required for monitoring are provided by adding bandwidth (most likely an additional bearer channel), thereby preserving the full user data rate. The overhead octets are included in each bearer channel.

6.2 Frame Structure

Each 56/64 kbit/s channel is organized into a sequence of frames. A frame consists of 256 octets. The octets are numbered 1 to 256. To establish the framing structure and provide for the exchange of information with the far end, each frame carries four octets of overhead (Octet 64, Octet 128, Octet 192 and Octet 256).

Overhead octets contain only seven bits of information so that a common structure can be used for 56 and 64 kbit/s bearer channels. In s 5.2.2 through 5.2.5, bit 8 is depicted as having the value 1. For 64 kbit/s bearer channels, bit 8 is set to "1". For 56 kbit/s bearer channels, the following procedures are followed.

- o At the transmitting endpoint, procedures such as CRC calculation are performed with all octets (data and overhead) having an 8th bit set to "1". If 56 kbit/s baseband bearer channels are used, bit 8 is stripped off of each octet before it is transmitted. Thus, only 7 bits are transmitted in each 125 µsec interval. If 56 kbit/s bearer channels are provided via 64 kbit/s facilities, 8 bits are transmitted in each 125 µsec interval, and the 8th bit remains set to "1" in all octets.
- o If the receiving endpoint uses 56 kbit/s baseband bearer channels, the incoming bit stream is serialized and searched for bits 1-7 of the Frame Alignment Word (FAW). When FAW is detected, an 8th bit set to "1" is added to each septet to form octets. All procedures such as CRC are then based on the resulting octets.
- o If the receiving endpoint uses 56 kbit/s bearer channels provided via 64 kbit/s facilities, the 8th bit is stripped upon receipt of each octet. The resulting 56 kbit/s bit stream is serialized, and searched for bits 1-7 of the FAW. When FAW is detected, an 8th bit set to "1" is added to each septet to form octets. All procedures such as CRC are then based on the resulting octets.

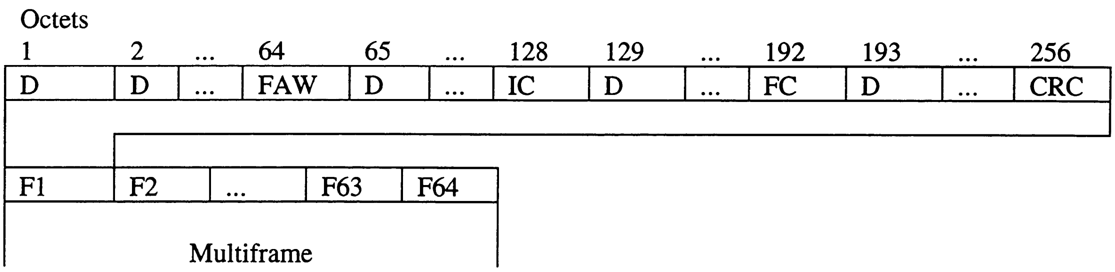
  NOTE 1 - The receiving endpoint may initially search for the entire FAW on the incoming 64 kbit/s bit stream, assuming octet alignment. However, if the incoming channel aggregation octets are not aligned with the network octets, FAW will not be detected and the procedure described above will have to be followed.

The receiving endpoint may need to serialize the incoming bit stream for 64 kbit/s bearer channels if one endpoint is not provided with network octet timing. To simplify procedures at the receiving endpoint, the transmitting endpoint **shall** align the channel aggregation octets with network octets if octet timing is provided.

NOTE 2 - Equipment that requires that the channel aggregation frames be synchronized with network provided octet timing will not be able to communicate with equipment connected to networks that do not provide octet timing.

A multiframe (MF) contains 64 consecutive frames numbered 1 to 64. The total duration of a multiframe is 2.048 seconds. Note that the duration of the multiframe must be twice as long as the desired amount of delay compensation. Hence, the frame structure described herein supports a maximum relative delay (between channels) of 1.024 seconds.

The following figure illustrates the framing structure.



FAW: Frame Alignment Word

IC: Information Channel

FC: Frame Count

CRC: Cyclic Redundancy Check

D: Data Octet

F1 - F64: Frames

Figure 3 - Framing Structure

6.2.1 Bit Ordering

In this International Standard, the bits of each octet are numbered b1 to b8 indicating the order of bit transmission. Numeric values are transmitted most significant bit first.

6.2.2 Frame Alignment Word (FAW)

Octet 64 in every frame contains the frame alignment word (FAW). The assignment of bits 1-8 of the FAW is shown below:

	b1	b2	b3	b4	b5	b6	b7	b8
FAW	0	0	0	1	1	0	1	1

As described in 6.2, for 56 kbit/s base bearer channels, only the first 7 bits are transmitted. Therefore, the FAW is a 7 bit word. On the incoming side, the receiver must search for a 7 bit Frame Alignment Word in order to synchronize

6.2.3 Frame Count (FC)

Octet 192 in every frame contains a Frame Count (FC) value. The Frame Count is used to measure the relative delay variance between the individual channels of an N x 56/64 call. This 6-bit, modulo 64 counter is incremented once every frame and rolls over on a multiframe boundary. The first frame of a multiframe **shall** contain a Frame count of zero.

The assignment of bits 1-8 of the FC is shown below. Note that bit 8 of the Frame Count is always set to binary 1.

	b1	b2	b3	b4	b5	b6	b7	b8
FC	1	Frame Count						1

Where 0 2 Frame Count 2 63.

6.2.4 Information Channel (IC)

Octet 128 in every frame contains the information channel (IC). The IC provides for the in band exchange of information with the far end device. A detailed description can be found in clause 7.

For initial parameter negotiation on the master channel, the Information Channel uses the entire bandwidth of the channel.

NOTE - Bit 8 of the Information Channel (b8) is always set to 1. See Figure 5 in Clause 7.

6.2.5 Cyclic Redundancy Check (CRC)

Octet 256 in every frame contains a Cyclic Redundancy Check (CRC). The assignment of bits 1-8 of the CRC octet is shown below. Note that bit 8 of the CRC octet is always set to binary 1.

	b1	b2	b3	b4	b5	b6	b7	b8
CRC	1	A	E	CRC4				1

A: Alignment Bit -- indicates that the remote end has frame alignment when A=1

E: Error Bit -- indicates that the remote end has detected a CRC4 error when E=1

CRC4: Contains the 4 bit CRC calculated for current block (see 6.4)

Use of the CRC4 is optional. A complete description of the CRC4 procedure, including use of the E bit, is given in 6.4. A description of the (A) bit is given in 5.3 below.

When CRC4 is not used, the transmitter **shall** set the E bit to 0 and the CRC4 bits to 1. When a unit receives E bit set to 0 and CRC4 set to all 1's, it **shall** disable the CRC procedure in the receive direction and **shall** ignore subsequent E bits and CRC4 bits. When an endpoint that implements the CRC procedure receives a CRC4 not set to all 1's it **shall** enable the CRC procedures.

6.2.6 Distribution of Overhead Octets

The overhead octets **shall** be distributed in the channels by the transmitting endpoint such that an overhead octet in channel "i+1" **shall** be offset from the corresponding overhead octet in channel "i" by 125 µsec. For example, if the FAW octet in channel 1 (as identified by the Channel Identifier) is transmitted starting at time t=0, the FAW octet in channel 2 **shall** be transmitted starting at time t = 125 µsec, the FAW octet in channel 3 **shall** be transmitted starting at time t = 250 µsec and in general, the FAW octet in channel "m" **shall** be transmitted starting at time t = (m-1)*125 µsec. The other overhead octets are transmitted with the same constraints. The objective of this requirement is to distribute the overhead octets as evenly as possible across all channels. This requirement is independent of the framing structure of the underlying physical medium.

The following example illustrates the distribution of overhead octets when calls are made using a T1 network access and using Mode 3 operation. This example is for T1 frames only and does not imply restriction of this procedure only to T1 transmission systems. In addition, for illustration purposes, this example uses contiguous channel assignment and does not imply restriction of this procedure to contiguous channel assignment. This example also assumes that the Channel Aggregation framing structure aligns with the DS1 framing structure. The Channel Aggregation framing structure is not required to align with the DS1 framing structure.

NOTE 1 - If the Layer 1 used is not clocked with 8 kHz framing (e.g., Layer 1 is V.35 connected to a DSU/CSU), then the DS1 frame can not be used as framing to find FAW. In this case, the incoming bit stream can be serialized before searching for framing.

Based on the requirement for distribution of overhead octets, the overhead octets are distributed within a frame in such a way as to displace only one octet within the 24 octet frame of a T1 system. Figure 4 shows how the overhead octets are dispersed within a sample 6 x 64 kbit/s call. The columns represent individual channels identified by the Channel Identifier. In this example the Channel Identifier is the same as the DS0 number used for the channel. This is not a requirement. An "FAW" or an "IC" indicates an overhead octet. The rows in the chart are instances of T1 frames.

Column 7 shows an N+1 channel used to replace the capacity lost due to overhead octets in Mode 3 operation so as to preserve the full user data rate. In Mode 2, column 7 is not used and the user data rate is that bandwidth remaining after the insertion of overhead octets. In Mode 1, no overhead octets are used and column 7 is not used (the entire bandwidth is available for data). In this example, the user data bytes are numbered in order of transmission up to 48. The rest of the data bytes are ordered in the same manner (across and down) *An asterisk (*) indicates an unused octet in which all bits are set to one.*

Channel Aggregation
Channels

		1	2	3	4	5	6	7		24/32
	1	FAW	D1	D2	D3	D4	D5	D6		
T1	2	D7	FAW	D8	D9	D10	D11	D12		
	3	D13	D14	FAW	D15	D16	D17	D18		
F	4	D19	D20	D21	FAW	D22	D23	D24		
R	5	D25	D26	D27	D28	FAW	D29	D30		
A	6	D31	D32	D33	D34	D35	FAW	D36		
M	7	D37	D38	D39	D40	D41	D42	FAW		
E	8	D43	D44	D45	D46	D47	D48	*		
S		o								
		o								
		o								
	65	IC	D	D	D	D	D	D		
	66	D	IC	D	D	D	D	D		
	67	D	D	IC	D	D	D	D		
	68	D	D	D	IC	D	D	D		
	69	D	D	D	D	IC	D	D		
	70	D	D	D	D	D	IC	D		
	71	D	D	D	D	D	D	IC		
	72	D	D	D	D	D	D	*		

Figure 4 - Overhead octets dispersed within a sample 6 x 64 kbit/s call (Mode 3 Operation).

NOTE 2 - Figure 4 shows the Channel Aggregation channel number. There is no relation between the Channel Aggregation channel number and the DS0 number.

6.3 Overall Frame alignment

In Modes 1, 2 and 3, the FAW in each channel is used at call setup time to align the individual channels. In Modes 2 and 3, the FAW in each channel is monitored continuously to determine if frame alignment has been lost and to recover frame alignment.

Loss of Frame Alignment:

An endpoint **shall** detect loss of frame alignment when it receives three consecutive frame alignment words (FAW) in error or three consecutive Frame Counts in error. Upon detection of loss of frame alignment, the endpoint **shall** set the Alignment bit (A) in the CRC octet to 0 in the transmit direction.

Frame Alignment Recovery:

An endpoint **shall** start the recovery of frame alignment immediately after detecting loss of frame alignment and at channel connection time. Frame alignment is defined to have been recovered when the endpoint detects the following sequence:

- the correct frame alignment word (FAW);
- the absence of the FAW 128 octets later, detected by verifying that Bit 1 = 1 in the FC octet;
- the presence of the correct frame alignment word (FAW) in the next frame
- the presence of CID = 0 when setting up the initial Master Channel.

To speed the initial establishment of frame alignment, an endpoint **shall** transmit an all ones pattern in the Data bits of the frame on initial establishment of the connection until it receives the remote Alignment (A) bit set to one for two consecutive CRC octets.

An endpoint **shall** begin transmitting an all ones pattern in the Data bits or the frame whenever it receives an A bit set to zero in two consecutive CRC octets. Receipt of an A bit set to zero in two consecutive CRC octets indicates that the remote endpoint has lost frame alignment.

6.4 Description of CRC4 Procedure

The CRC4 procedure can be used to provide an end-to-end quality monitor of each 56/64 kbit/s channel. The four CRC bits (bit positions 4 to 7 of octet 256) contain the CRC computed at the source location. In addition, bit 3 of the same octet, referred to as the E bit, is used to transmit an error indication about the received data in the opposite direction (i.e., whether or not the previous frame was received with errors).

The CRC4 used in these procedures is defined in ITU-T Recommendation G.704.

6.4.1 Computation of the CRC4 Bits

The CRC4 is computed from the whole 56/64 kbit/s channel from the previous 255 octets of the frame before the CRC4 octet. The CRC4 is therefore computed over 255 octets starting with the first octet of the frame (i.e., the octet immediately following the CRC4 octet of the previous frame) and ending with the octet just preceding the CRC4 octet (i.e., Octet 255). The CRC4 octet is not included in the CRC4 calculation.

6.4.1.1 Multiplication and Division Process

A given CRC4 located in Octet 256 is the remainder after multiplication by X^4 and then division (modulo 2) by the generator polynomial $X^4 + X + 1$ of the polynomial representation of the 255 octets preceding the CRC octet in the frame.

When representing contents of a frame as a polynomial, the first bit in the frame **shall** be taken as being the most significant bit. Similarly Bit 4 of the CRC octet is defined to be the most significant bit of the remainder and Bit 7 is the least significant bit of the remainder.

6.4.1.2 Encoding Procedure

Octets 1 through 255 of the frame are acted upon by the multiplication and division process defined in 5.4.1.1. The remainder, resulting from the multiplication and division process, is inserted into the CRC4 field of Octet 256 (the CRC octet) of the frame. The CRC is always computed over 8 bit octets. When using 56 kbit/s bearer channels, bit 8 of each octet **shall** be forced to "1" for consistency.

NOTE - As described in 6.2, for 56 kbit/s bearer channels, an 8th bit set to 1 is stuffed at the end of each septet before CRC encoding. This 8th bit of each octet will be stripped after CRC calculation and before transmission if 56 kbit/s baseband bearer channels are used

6.4.1.3 Decoding Procedure

Octets 1 through 255 of a received frame are acted upon by the multiplication and division process, defined in 6.4.1.1. When using 56 kbit/s bearer channels, an 8th bit set to "1" shall be added to each septet (as derived from FAW alignment) to form octets for consistency. The remainder, resulting from the multiplication and division process is compared on a bit-by-bit basis with the CRC4 received in Octet 256 of the frame. If the decoded, calculated remainder equals the CRC4 received in Octet 256, then the checked frame is considered error-free. If they are different, then the checked frame is considered in error.

NOTE - As described in 6.2, CRC calculation for 56 kbit/s bearer channels is performed after insertion of an 8th bit set to 1 at the end of each septet.

6.4.2 Consequent Actions

6.4.2.1 Enabling and Disabling the CRC Procedures

An endpoint **shall** indicate that CRC procedures are being used by starting the CRC Encoding process on the transmitting end, thereby not transmitting the CRC bits as all 1's. If an endpoint supports the CRC procedure on this channel, it **shall** enable the CRC Decoding process when it receives a CRC Octet with the CRC bits set to a value other than all 1's; otherwise, it **shall** transmit the E bit set to 0 and CRC set to all 1's.

An endpoint **shall** indicate that CRC Encoding procedures are being disabled by setting all CRC bits to binary 1 and setting the E bit to 0. An endpoint **shall** disable the Decoding process when it receives three CRC octets with CRC set to all 1's and an E bit set to 0.

The above procedure can induce a short period of CRC errors while the endpoints start and stop CRC procedures. The actions taken in the following two subclauses should take this into account.
This procedure allows an endpoint to enable the CRC procedures on a channel by channel basis.

6.4.2.2 Action on Bit E

The E bit of the CRC octet is set to 1 in the transmitting direction when a CRC error is detected in the receive direction (see 6.4.1.3). Otherwise, it is set to zero.

6.4.2.3 Monitoring for Error Performance

The quality of the received 56/64 kbit/s channel can be monitored by counting the number of CRC frames in error within a period of one second. The quality of the transmitted 56/64 kbit/s channel can be monitored by counting the number of received E bits set to 1.

The threshold for declaring a channel out of service is left for further study and is implementation dependent.

7 Description of the Information Channel

The information channel is used to communicate control information between two endpoints. In all modes, the Information Channel occupies the entire bandwidth of the Master Channel during initial parameter negotiation. In modes 2 and 3, the Information Channel occupies octet 128 of the Channel Aggregation Frame in all bearer channels during delay equalization and throughout the duration of the call. In mode 1, the Information Channel occupies octet 128 of the Channel Aggregation Frame in all bearer channels only during the initial delay equalization. In mode 0, the Information Channel is not present after initial parameter negotiation.

7.1 Information Channel Frame

Information channel frames are framed with a unique ALIGNment pattern (01111111) in the first octet. In addition, the first bit (b1) of the subsequent octets is set to one. Figure 5 shows the structure of the Information Channel (IC) frame. IC octets only contain seven bits of information so that a common structure can be used for 56 and 64 kbit/s bearer channels. In Figure 5, bit 8 of each octet is depicted as having the value 1. For 64 kbit/s bearer channels, bit 8 is set to 1. For 56 kbit/s bearer channels provided via 64 kbit/s facilities, bit 8 is used for network signaling and is set to 1 for off-hook. For the case of 56 kbit/s baseband bearer channels, bit 8 is not present and an overhead "octet" transmitted over the bearer channel contains only 7 bits. 6.2 describes procedures for using 56 kbit/s bearer channels.

	b1	b2	b3	b4	b5	b6	b7	b8
	ALIGN							
Octet 1	0	1	1	1	1	1	1	1
2	1	Channel ID						1
3	1	Group ID						1
4	1	Operating Mode			Res			1
5	1	RMULT						1
6	1	SUBMULT			BCR	Res	MFG	1
7	1	RI	RL Req	RL Ind	REV			1
8	1	Subaddress						1
9	1	Transfer Flag						1
10	1	1	1	Digit - 1				1
11	1	1	1	Digit - 2				1
12	1	1	1	Digit - 3				1
13	1	1	1	Digit - 4				1
14	1	1	1	Digit - 5				1
15	1	1	1	Digit - 6				1
16	1	1	1	Digit - 7				1

Figure 5 - Information Channel Frame

NOTE 1 - Annex D provides details of the additional capability to be used when phone numbers longer than 7 digits are required.

ALIGN: Alignment octet (Octet 1, Bits 1-8)

The alignment octet provides a framing mechanism for the Information channel frame. It takes a constant value of "01111111".

NOTE 2 - For 56 kbit/s baseband bearer channels, only the first seven bits of the ALIGN octet are transmitted (i.e., 0111111).

CID: Channel identifier (Octet 2, Bits 2-7)

When simultaneously dialing N calls, the network delay in call setup for each call might not be uniform. Hence the incoming call arrival sequence at the Answering Endpoint cannot be assumed to be equal to the call setup sequence at the Calling Endpoint. The CID identifies each individual channel in the transmit direction for each side of the connection and is used to sequence the time slots in proper order at the receiving end. In each endpoint, the CID value transmitted in each channel is independent of the CID value received from the other endpoint. The Calling and Answering endpoint **shall** assign CID = 1 to the initial channel and **shall** assign CID in sequential ascending order corresponding to the relative order of the remaining channels. A value of 0 in the CID field indicates negotiation of parameters.

The CID is a 6-bit binary encoded number in the range $0 \leq CID \leq 63$.

GID: Group identifier (Octet 3, Bits 2-7)

The call Group Identifier is used to identify the group of bearer channels associated with a particular call. The value of this parameter **shall** be agreed upon by both ends at the time of establishing the master channel. The GID negotiated on the master channel **shall** be used in all subsequent bearer channels for this call. The GID is assigned by the Answering Endpoint and is unique to this call at the Answering Endpoint. The Calling Endpoint uses a combination of the GID and a local identifier to uniquely identify a call. The value of 0 is reserved to indicate an unknown GID during parameter negotiation and channel initiation.

This field contains a binary encoded number in the range $0 \leq GID \leq 63$

MODE: Operating mode (Octet 4, Bits 2-4)

This field indicates the mode of operation desired for this call. The four modes of operation are described in 6.1 of this International Standard.

Bits 2 3 4	
0 0 0	operating Mode 0
0 0 1	operating Mode 1
0 1 0	operating Mode 2
0 1 1	operating Mode 3
1 0 0	Reserved
1 0 1	Reserved
1 1 0	Reserved
1 1 1	Reserved for negotiation.

NOTE 3 - There is no coding for the Transparent Mode because in the Transparent Mode there is no Information Channel.

Res: Reserved (Octet 4, Bits 5-7):

These bits are reserved and **shall** be set to 1 on transmission and **shall** be ignored on receipt.

RMULT: Rate Multiplier (Octet 5, Bits 2-7)

The Rate Multiplier field RMULT together with the Subrate Multiplier field SUBMULT contains information that uniquely defines the application rate for a given call. The application rate is determined from these fields in slightly different ways depending on the Operating Mode given in the MODE field of the Information Message (Octet 4, Bits 2-4). See 7.1.1 for details.

RMULT takes a value in the range $0 \leq \text{RMULT} \leq 63$.

SUBMULT: Subrate Multiplier (Octet 6, Bits 2 - 4):

The Rate Multiplier field RMULT together with the Subrate Multiplier field SUBMULT contains information that uniquely defines the application rate for a given call. For Modes 0, 1 and 2, the SUBMULT field **shall** be set to all binary ones on transmit and ignored on receipt. See 7.1.1 for details.

For Mode 3, SUBMULT takes a value in the range $0 \leq \text{SUBMULT} \leq 6$ for BCR = 56 kbit/s and $0 \leq \text{SUBMULT} \leq 7$ for 64 kbit/s

BCR: Bearer Channel Rate (Octet 6, Bit 5)

This field indicates the data rate to use when determining how to use the bearer channel. A 56 kbit/s BCR means that the channels used are based on 56 kbit/s multiples and a 64 kbit/s BCR means that the channels used are based on 64 kbit/s multiples.

This field takes the following values

Bit 5	
1	64 kbit/s base
0	56 kbit/s base

Res: Reserved (Octet 6, Bit 6):

This bit is reserved and **shall** be set to 1 on transmission and **shall** be ignored on receipt.

MFG: Manufacturing ID Flag (Octet 6, Bit 7):

When this bit is set to 1, the Digits field contains a 7 binary encoded digit Manufacturer's ID. Annex C contains the Manufacturer ID assignments. When this bit is set to 0, the Digits field does not contain the Manufacturers ID and is used according to other procedures.

The Manufacturer's ID is exchanged between equipment at call initialization time. Equipment can use the ID to identify the equipment on the other end of the connection. Manufacturers can use this information to identify their own equipment on the other end of the connection in order to add proprietary extensions to the protocol.

RI: Remote indicator (Octet 7, Bit 2)

The RI field indicates that delay equalization has been accomplished.

Bit 2
 1 call is delay equalized
 0 call is not delay equalized.

RL REQ: Remote loopback Request (Octet 7, Bit 3)

This bit is used to request that the remote endpoint loop back toward the local endpoint. The receipt of an RL REQ bit set to one indicates a request to enable remote loopback. RL REQ set to zero indicates a request to disable remote loopback. **When remote loopback is not requested, this bit shall be set to 0.**

Bit 3
 1 request for remote loopback
 0 turn off remote loopback.

RL IND: Remote Loopback Indication (Octet 7, Bit 4)

This bit is used to indicate to the remote endpoint that the local endpoint is looping the delay equalized payload from all channels back to the remote endpoint. This bit is used as a response to the receipt of RL REQ =1 or to indicate an unsolicited loopback. **When not in loopback, this bit shall be set to 0.**

Bit 4
 1 Sending endpoint is in loopback state
 0 Sending endpoint is not in loopback state.

REV: REVision level (Octet 7, Bits 5-7)

This field can be used to check version compatibility between the local and far end units. If this field is not used it **shall** be set to all binary ones on the transmitting end and ignored on the receiving end. All endpoints developed to Issue 1 of this International Standard **shall** use a Revision Level of 0.

This is a binary encoded number in the range $0 \leq REV \leq 7$.

If one CAU transmits a higher value of REV than the other CAU then only those procedures defined for the lower value of REV shall be used by both CAUs.

Subaddress: Subaddress (Octet 8, Bits 2-7)

This field is used to indicate the subaddress of a call. It takes a value in the range from 0 to 63. A value of 0 indicates that the subaddress is not used.

A CAU may be capable of handling multiple simultaneous calls; the subaddress provides a means of distinguishing an individual call amongst the multiple calls that may exist.

XFLAG: Transfer Flag (Octet 9, Bits 2-7)

When CID = 0, the transfer flag is used during initial parameter negotiation for signaling the transmission of additional phone numbers (7 digits each) to the calling unit. During Channel Deletion procedures, the XFLAG field is used to identify the channel to be deleted.

See 8.1 for description of the phone number transfer procedures and 8.3 for channel deletion procedures.

This field is binary encoded and takes values in the range $0 \leq XFLAG \leq 63$

When CID is not equal to 0, this field and bits 2-7 of octets 10 - 16 can be used for User Information transfer and are not defined in this International Standard.

DIGITS: Phone Number Dial Digits (Octets 10 - 16, Bits 4-7)

When CID = 0 and MFG = 0 the Digits fields are used for phone number negotiation during initial parameter negotiation. There are seven Digits fields, each containing a number from the list below. When sending less than 7 digits, the endpoint **shall** left justify the number (i.e., if there are M digits, then these digits fill the first M Digits fields) and **shall** fill the unused Digits fields with the EON value.

See 8.1 and 8.2 for phone number delivery procedures.

When CID is not equal to 0, the XFLAG field and bits 2-7 of octets 10 - 16 can be used for User Information transfer and are not defined in this International Standard.

When CID = 0 and MFG = 1, the Digits field contains a binary encoded 7 digit number specifying the Manufacturer's ID of the manufacturer of the transmitting endpoint.

Bits 4 5 6 7	Digits
0 0 0 0	0
0 0 0 1	1
0 0 1 0	2
0 0 1 1	3
0 1 0 0	4
0 1 0 1	5
0 1 1 0	6
0 1 1 1	7
1 0 0 0	8
1 0 0 1	9
1 0 1 0	*
1 0 1 1	#
1 1 0 0	EON
1 1 0 1	Reserved
1 1 1 0	Cause Indicator
1 1 1 1	PAD

Cause: Cause Codes (Octets 10 and 11, Bits 2 - 7)

When the first digits field has the value "1110", it indicates that the following field is a Cause Value. The Cause Value is a binary encoded value in the range $0 \leq \text{Cause} \leq 63$. The following cause codes have been defined:

Value	Meaning
0	Reserved
1	No resource available
2	Channel error rate exceeded threshold
3	Normal Clearing
4	Bandwidth addition request
5	Mode not supported
6	Rate not supported
7	Information Channel framing lost
8	FAW framing lost
9	Delay Equalization unsuccessful
10	Unknown command/protocol error
11	Temporary failure, in recovery
12	Failure, no recovery
13	Bandwidth deletion request

Cause Codes 14 - 63 are reserved for further study.

Cause Codes can be used whenever negotiations take place (i.e., CID = 0 in the master channel), and Manufacturer's ID and Phone Numbers are not being transferred. Also Cause Codes can be transmitted by forcing an "Add a Channel" or "Delete a Channel" procedure with no change in the RMULT or SUBMULT field.

NOTE - ISDN subaddresses may be included in the digits field.

7.1.1 Rate Multiplier and Subrate Multiplier Calculation and Operation

Given the Bearer Channel Rate (56 or 64 kbit/s), the Application Data Rate can be derived from the RMULT and SUBMULT values or the RMULT and SUBMULT values can be derived from the application data rate (A) as follows: In the following equations, all data rates are in bits/second.

Mode 0, 1:

E1. $A = \text{RMULT} * \text{BCR}$

E2. $\text{RMULT} = \frac{A}{\text{BCR}}$

Mode 2:

E3. $A = RMULT * BCR * \left(\frac{63}{64}\right)$

E4. $RMULT = \frac{(A * 64)}{(BCR * 63)}$

Mode 3:

E5. $A = (RMULT * BCR) + (SUBMULT * 8000)$

E6. $RMULT = \left\lfloor \frac{A}{BCR} \right\rfloor$ (rounded down to the next lowest integer)

E7. $SUBMULT = \frac{(A - (RMULT * BCR))}{8000}$

NOTE- The SUBMULT field only has meaning in Mode 3. This is because Modes 0, 1 and 2 only support rates that are multiples of the bearer rate (in Mode 2, multiples of 63/64 of the bearer rate.). Mode 3 supports application data rates of any multiple of 8 kbit/s; therefore, the SUBMULT field is required to specify the application data rate with an 8 kbit/s granularity. In Modes 0, 1 and 2 the SUBMULT field **shall** be set to all ones on transmission and ignored on receipt.

Let EBR (Effective Bearer Rate) represent the effective bandwidth per network channel available for the transfer of application data. EBR is defined as the Bearer Channel Rate (BCR) minus the bandwidth required to transfer the Channel Aggregation overhead octets.

Mode 0, 1:

E8. $EBR = BCR$, since the Channel Aggregation overhead octets are removed before application data is transferred over the channel.

Modes 2 and 3:

E9. $EBR = \left(\frac{63}{64}\right) * BCR$, since every 64th octet of a network channel is required for the transfer of overhead octets.

The number of network channels (N) required to complete the call is then determined directly based on application data rate and effective bearer rate:

Modes 0, 1, 2 and 3:

E10. $N = \left\lceil \frac{A}{EBR} \right\rceil$ rounded up to the nearest integer.

From the above equations it follows that for Modes 0, 1 and 2 the number of channels N required for a given call is equal to the Rate Multiplier RMULT. Only in Mode 3 is it necessary to explicitly calculate N. In the case of Mode 3, N can be calculated from RMULT and SUBMULT by substituting equation E5 for A to get

E11. $N = \frac{RMULT * BCR + SUBMULT * 8000}{EBR}$

Equation E11 must be used on the receiving end of a parameter negotiation frame to determine the number of channels.

7.1.1.1 Distribution of Application Data Across the Channel Aggregation Frame

In Modes 0, 1 and 2 the application data rate is only allowed to be integer multiples of the effective bearer channel rate (EBR). In Mode 0 and 1, EBR is the same as the bearer channel rate BCR (i.e., EBR = BCR = 56 or 64 kbit/s). In Mode 2 EBR is BCR scaled by 63/64 (i.e., EBR = BCR * (63/64) = 55.125 kbit/s (BCR = 56 kbit/s) or 63 kbit/s (BCR = 64 kbit/s)). In Modes 1 and 2 all available bandwidth is used by data (Mode 1) or data and overhead (Mode 2).

In Mode 3 the application data rate (A) is allowed to be an integer multiple of 8 kbit/s. To support this granularity in the application data rate, only a portion of the available bandwidth in the Nth network channel is used. In Mode 3, two parameters (L and M) are calculated based on application rate and bearer channel rate. These parameters define the number of bits within the Nth channel that carry application data.

The parameter L specifies the number of octets in each 64 octet segment (bounded by overhead octets) in the Nth channel that contain application data. Transmission of the L octets in the Nth channel begins (N-1) octets before the overhead octet. The parameter M specifies the number of bits in the last octet of each L octet cluster that contains application data. For example, if M is calculated to be 3, then the most significant three bits are filled with application data and the last 5 bits are forced to ones as shown below.

b1	b2	b3	b4	b5	b6	b7	b8
D1	D2	D3	1	1	1	1	1

NOTE - All unused bit locations in the Nth channel **shall** be set to all ones.

For example, for BCR = 56 kbit/s, N = 4, L = 22 and M = 2 (Application Data Rate of 184 kbit/s, RMULT = 3, SUBMULT = 2), application data is disbursed in the Nth channel such that N - 1 application data octets are transmitted before an overhead octet (e.g., FAW) and the remaining application data octets are transmitted after the overhead octet (e.g., FAW). In this case the output is shown in Figure 6. The first column represents time (e.g., DS1 frames). Columns 2 - 5 contain the four channels used in the call. Column 6 contains the number of octets (in that DS1 frame) that contain data. For example, in the first DS1 frame, one octet is used for FAW and (N-1) octets are used for user data. This example shows data starting at the FAW overhead octet of the first channel instead of the first octet of the frame. In the Nth channel, three octets before the overhead octet are used for data and L - 3 (= 19) octets after the overhead octet are used for user data. An X in the Nth channel indicates that these octets are set to 1 and do not contain User Data. A D indicates a data octet and D2 indicates a data octet containing two data bits.

This pattern repeats itself around every overhead octet.

t	CID=1	CID=2	CID=3	CID=4	# of Octets with Data
1	FAW	D	D	D	N-1
2	D	FAW	D	D	N-1
3	D	D	FAW	D	N-1
4	D	D	D	FAW	N-1
5	D	D	D	D	N
6	D	D	D	D	N
7	D	D	D	D	N
8	D	D	D	D	N
•	•	•	•	•	•
•	•	•	•	•	•
22	D	D	D	D	N
23	D	D	D	D2	N
24	D	D	D	X	N-1
•	•	•	•	•	•
•	•	•	•	•	•
64	D	D	D	X	N-1
65	IC	D	D	D	N-1
66	D	IC	D	D	N-1
67	D	D	IC	D	N-1
68	D	D	D	IC	N-1
69	D	D	D	D	N
70	D	D	D	D	N
•	•	•	•	•	•
•	•	•	•	•	•

Figure 6 - Distribution of Application Data Across the Channel Aggregation Frame

7.1.1.2 Calculation of Parameters L and M

This subclause is used only for Mode 3.

In order to calculate L and M, the total number of application data bits that need to be transferred in the Nth channel (#BITS(N)) must first be determined. Let REM(N) be defined as the remainder of the application data rate A divided by the effective bearer channel rate EBR (EBR = (63/64) * BCR), except when the remainder = 0.

E12. $REM(N) = remainder\left[\frac{A}{EBR}\right]$ except that if remainder = 0, then REM(N) = 1

Since there are 252 octets per Channel Aggregation frame that can contain data,

E13. #BITS(N) = REM(N) * 252 * (7 + BASE),

where BASE is defined to be 1 if the bearer channel rate (BCR) is 64 kbit/s and 0 if the BCR is 56 k/bits. The quantity (7 + BASE) is the number of allowed application data bits per octet.

L is then calculated as follows:

E14. $L = \frac{\#BITS(N)}{(4 * (7 + BASE))}$ rounded up to the nearest integer.

NOTE - The factor of 4 in the divisor is required since the data in the Nth channel is placed in the Channel Aggregation frame in four separate blocks of L octets.

M is then calculated based on L and #BITS(N). If $(L * (7 + BASE)) = \frac{\#BITS(N)}{4}$ exactly (i.e., no rounding is required to determine L) then,

M = (7+BASE), i.e., the Lth octet contains no unused bit fields;

otherwise,

$$M = \frac{\#BITS(N)}{4} - ((L-1)*(7+BASE))$$

The table in Annex B shows the values of M and L for application rates of all multiples of 8000 bit/s that can be supported in Mode 3 on up to 24 bearer channels. The values are shown for bearer channel rates of 56 and 64 kbit/s. In addition, for each rate the appropriate values for the Rate Multiplier RMULT and the Subrate Multiplier SUBMULT are given as well as the number of required network channels N necessary to place the call.

7.2 Information Channel (IC) Synchronization

Except for the initial channel parameter negotiation, an endpoint **shall** establish overall frame alignment prior to establishment of Information Channel synchronization. For initial channel parameter negotiation, the Information channel uses the entire bandwidth of the master channel.

The ALIGN octet is used to determine IC synchronization.

IC Synchronization Detection

An endpoint **shall** declare IC synchronization when it detects the following sequence:

Detection of the ALIGN pattern.

Detection of Bit 1 = 1 of the next octet.

Detection of a second ALIGN pattern in the next correct position (16 octets after the initial ALIGN pattern).

Loss of IC Synchronization

An endpoint **shall** declare the loss of IC synchronization when it receives three consecutive incorrect ALIGN octets.

In Modes 2 and 3, the endpoint continuously monitors the ALIGN pattern to determine if Information Channel synchronization has been lost. Modes 0 and 1 only use the ALIGN pattern and the Information Channel during call setup.

The endpoint **shall** start the IC synchronization detection procedure immediately once it detects loss of IC frame synchronization or achieves multiframe synchronization.

7.3 Transmission and Recognition of an Information Channel Frame

Once a connection has been established, an endpoint transmits the Information Message continuously in the Information Channel (see Clause 8). When this International Standard refers to "sending" an Information Message, it means the endpoint starts transmitting the new values in the Information Channel Frame. Please note that for initial parameter negotiation in the master channel, the Information Message is transmitted and received using the entire bandwidth of the master channel and does not use the Information Channel of the multiframe structure until after parameter negotiation is complete.

Once Information Channel Synchronization has been achieved, the endpoint can transmit and receive Information Messages in the Information Channel. Since these frames are transmitted continuously, the receiving endpoint has to monitor the received frames for a change in value. Also, since there is no CRC or checksum on the IC Frame, the receiving endpoint should not be sensitive to instantaneous changes in state. When this International Standard refers to "receiving" an Information Message, it means that the endpoint has detected a valid change in values of the Information Channel Frame. The method for determining a valid change in values is implementation dependent, but it is recommended that the receiver check several Information Channel Frames in succession before declaring a change in values (i.e., 2 out of 3 majority voting).

When transmitting Information Channel messages, the following constraints **shall** be followed:

- When transmitting messages with CID=0, the same Information Channel message **shall** be transmitted in all channels (including Channel ID = 0).
- The same values of the GID, MODE, RMULT, SUBMULT, BCR, RI, RL REQ, RL IND fields **shall** be transmitted in all channels when CID is not equal to 0. This means that when a value changes in one channel, it changes in all channels.

The rest of this International Standard refers to a valid (debounced) Information Channel Frame as an Information Message.

8 Procedures

8.1 Call Setup

8.1.1 Initial Channel Setup

This subclause describes the interactions between Calling and Answering Endpoints in order to establish a call. It is structured in sequence order.

The initial call setup procedure begins with the connection of the first channel in the n x 56/64 kbit/s call. This channel is referred to as the "master" channel. The parameter negotiation process is performed over this channel. Neither Endpoint **shall** make any additional call requests until the entire negotiation process has been completed.

Calling Endpoint

Once the Calling Endpoint receives an indication that the master channel is connected, and following selection of ISO/IEC aggregation at point (3) of Figure 2 if applicable, it **shall** begin the negotiation process by repeatedly transmitting the Information Message (using the entire bandwidth of the master channel) with the Channel Identifier (CID) set to 0 and **shall** start TCinit and TCnull (if implemented - See 9.2). (A CID of 0 is used as the negotiation flag). The Calling Endpoint also **shall** include the parameter values desired for the call (i.e., Mode, RMULT, SUBMULT, MFG, BCR, and Revision). The Group Identifier (GID) **shall** be set to 0. The Remote Indicator bit and the Remote Loopback bits **shall** be set to 0. In the first message, the Calling Endpoint can send its Manufacturer's ID by setting the MFG Flag to 1 and including the ID in the Digits field. In the initial message, XFLAG **shall** be set to all binary 1's. If the Calling Endpoint does not send its Manufacturer's ID then the MFG Flag **shall** be set to 0 and the Digits field **shall** be set to all 1's.

If the Calling Endpoint supports subaddressing, it includes the subaddress in the Subaddress field. If the Calling Endpoint does not support subaddressing or if the Calling Endpoint does not request a subaddress, it sets the Subaddress field to all binary 0's.

Answering Endpoint

Upon receipt of an incoming call and indication that the channel is connected, and following selection of ISO/IEC aggregation at point (3) in Figure 2 if applicable, the Answering Endpoint **shall** transmit all 1's on the channel, start TAnull and start searching for multiframe or Information Channel alignment (i.e., Frame Alignment Word or ALIGNment octet).

NOTE - If the Answering Endpoint detects Frame Alignment, then the incoming call is an additional channel of an existing call. If the Answering Endpoint detects Information Channel Alignment without detecting overall Frame Alignment, then the incoming call is the master channel of a new call. See 8.1.2 for a description of additional channel setup. If TAnull expires, the Answering Endpoint defaults to Transparent Mode (see 9.1). Additional transparent channels are treated separately using initial channel setup procedures.

Once the Answering Endpoint detects a valid Information message, at point (2) in Figure 2 if applicable, it **shall** consider the channel to be the master channel of a new call and **shall** stop TAnull. If multiframe is detected, the Answering Endpoint **shall** stop TAnull, **shall** treat it as an additional channel for an existing call and **shall** use the Group Identifier to identify the existing call (see 8.1.2).

For the master channel of a new call, if the Answering Endpoint accepts the requested parameters, it **shall** return the same values as were received and start TAinit. If the Answering Endpoint does not accept the requested parameters, it **shall** either return RMULT and SUBMULT = 0 (disconnect request) with a valid Mode value and start TAdisc or return a different set of parameters (parameter negotiation) and start TAinit. The Answering Endpoint **shall** assign a Group Identifier to the call and return it in the GID field. This identifier identifies this call uniquely among calls received by this Answering Endpoint. The XFLAG field **shall** be set to all 1's.

If the Answering Endpoint chooses to return a Manufacturer's ID, it **shall** set the MFG Flag to 1, include the ID in the Digits field and send the message back with the other parameters as the first response to the Calling Endpoint. If the Answering Endpoint does not return a Manufacturer's ID, it **shall** transmit MFG set to 0 and the Digits fields set to all 1's.

If the Answering Endpoint does not support Subaddressing, it returns all binary 0's in the Subaddress field and ignores the incoming Subaddress field. If the Answering Endpoint does support subaddressing, it looks at the incoming Subaddress field to get the subaddress. If this subaddress is acceptable, it returns the same subaddress. If it is unacceptable, it **shall** either disconnect the call (using negotiation) or return a different subaddress. If the Answering Endpoint receives a subaddress of decimal 0, then it **shall** either return a subaddress of decimal 0 or return the subaddress assigned to the call.

Calling Endpoint

The Calling Endpoint **shall** detect the response from the Answering Endpoint, at point (2) in Figure 1bis if applicable, when it receives an Information Message with CID=0 and the MODE field containing a valid value (i.e., "000", "001", "010", "011"). At this point, the Calling Endpoint **shall** stop timer TCnull (if implemented - see 8.2). The Calling Endpoint **shall** either accept the parameter values requested by the Answering Endpoint or initiate disconnect procedures. If the MFG field is set to 1, the Manufacturer's ID is contained in the Digits field. If the MFG field is set to 0, the Calling Endpoint **shall** ignore the Digits field. In either case, the Calling Endpoint **shall** ignore the XFLAG field. If it decides to proceed, it **shall** initiate the transfer of phone numbers .

If the Calling Endpoint does not use the Subaddress field, it **shall** ignore the contents of that field. Otherwise, it can use the contents of the Subaddress field to determine the subaddress assigned to the call by the Answering Endpoint.

The Calling Endpoint **shall** request the initial phone number by sending the Answering Endpoint an Information Message with CID=0 and the transfer flag (XFLAG) set to Decimal 1 and the Digits field set to all 1's. The Calling Endpoint **shall** set the rest of the parameters to the negotiated values.

Answering Endpoint

After transmitting the initialization response, when the Answering Endpoint receives a message with XFLAG set to decimal 1, the Answering Endpoint **shall** return the first number by setting the XFLAG field to decimal 1 and placing the phone number in the Digits field. If the phone number to be used is greater than 7 digits in length (e.g., international calls), the Answering Endpoint **shall** return the last (rightmost) 7 digits. If the phone number to be used is less than 7 digits in length (e.g., 5 digit private numbers), the Answering Endpoint **shall** place the digits in the first M Digit fields where M is the number of digits. The Answering Endpoint **shall** fill the last 7 - M Digit fields with the EON character (bit pattern "1 1 0 0").

Calling Endpoint

When the Answering Endpoint returns M digits, the Calling Endpoint replaces the last M digits of the original phone number with the M digits received from the Answering Endpoint. If M = 0 (all Digits fields contain the EON character), the Calling Endpoint uses the original phone number.

The Calling Endpoint **shall** request the additional phone number by sending the Answering Endpoint an Information Message with CID=0 and the transfer flag (XFLAG) incremented by 1 indicating acceptance of the first number and readiness to receive the next phone number. The Digits field **shall** be set to all 1's. The Calling Endpoint **shall** set the rest of the parameters to the negotiated values. If the number of channels is N, the Calling Endpoint **shall** request NÊ-Ê1 phone numbers.

Answering Endpoint

When the Answering Endpoint receives an Information Message with CID=0 and the XFLAG field incremented by 1, it **shall** transmit the next phone number in the Digits field, and the received XFLAG in the XFLAG field.

If the Answering Endpoint only supports one phone number, it **shall** return the EON character in all the Digits fields for each phone number request received from the Calling Endpoint.

Calling Endpoint

The Calling Endpoint **shall** use the phone numbers received from the Answering Endpoint in the same proportion as were received from the Answering Endpoint. For example, if the Answering Endpoint returns EON for all phone numbers, the Calling Endpoint uses the original phone number for all channels. If the Answering Endpoint returns a different phone number for each channel, the Calling Endpoint uses a different phone number for each additional channel. If the Answering Endpoint has two phone numbers A and B and returns phone number A twice and phone number B three times, the Calling Endpoint **shall** use phone number A for two channels and phone number B for three channels.

The complete list of phone numbers is passed in this manner until the Calling Endpoint receives the last number. The Calling Endpoint **shall** indicate the conclusion of the negotiation process by transmitting an Information message with the channel identifier (CID) set to 1 and parameters set to negotiated values.

Answering Endpoint

When the Answering Endpoint receives an Information Message with CID set to 1, it **shall** transmit its Information Message with the channel identifier (CID) set to 1 and the other parameters set to the negotiated values. This indicates to the Calling Endpoint that the Answering Endpoint is ready to accept establishment of additional channels. The Answering Endpoint also **shall** stop TAinit and start TAadd01.

Calling Endpoint

When the Calling Endpoint receives an Information Message with CID=1, it **shall** stop TCinit, start TCadd01 and initiate the connection of the remaining channels in the call. Parallel dialing is optional but recommended to reduce call setup times. However, it should be noted that there may be a requirement in some networks to insert a programmable delay between consecutive additional channel setup attempts. For Modes 1, 2 and 3, the Calling Endpoint also **shall** start transmitting the multiframe structure in the master channel (immediately after receiving CID=1 from the Answering Endpoint) with the Alignment (A) bit in the CRC octet set to zero, the E bit set to 0 and the CRC4 bits set to 1. It also **shall** send all ones in the data octets. The Calling Endpoint also **shall** start searching for overall frame alignment in the master channel and start TCfa. The Calling Endpoint **shall** transmit the following field values in the Information Message:

CID: Set to 1.

RI, RL REQ and RL IND: Set to 0. These value can change subsequently due to achievement of Delay Equalization or Remote Loopback operation.

MFG ID: Set to 0.

Group Identifier: The GID assigned to this call by the Answering Endpoint.

XFLAG: Information subchannel.

Digit Fields: Information subchannel.

Other parameters: Set to negotiated values.

For Mode 0, the Calling Endpoint **shall** cut the channel through to the Application.

Answering Endpoint

After the Answering Endpoint has transmitted an Information Message with CID=1, it **shall** wait until loss of Information Message Alignment. At this point, for Modes 1, 2 and 3, the Answering Endpoint starts to transmit the multiframe structure. When the Answering Endpoint detects loss of Information Message Alignment, for Modes 1, 2 and 3, it **shall** start transmitting the multiframe structure in the master channel with the Alignment (A) bit in the CRC octet set to zero, the E bit set to 0 and the CRC4 bits set to 1. It also **shall** send all ones in the data octets. Once the Answering Endpoint starts transmitting the multiframe structure, it **shall** start searching for overall frame alignment in the master channel and start TAfa. The Answering Endpoint **shall** transmit the following field values in the Information Message in the Information Channel:

CID: Set to 1.

RI, RL REQ and RL IND: Set to 0. These value can change subsequently due to achievement of Delay Equalization or Remote Loopback operation.

Group Identifier: The GID assigned to this call by the Answering Endpoint.

MFG Flag: Set to 0.

XFLAG: Information subchannel.

Digit Fields: Information subchannel.

Other parameters: Set to negotiated values.

For Mode 0, after detecting loss of Information Channel Alignment, the Answering Unit stops sending Information Frames and cuts through the Application.

For Modes 1, 2 and 3, when each end detects Frame Alignment, it **shall** set the A bit to 1 in the transmit direction and start the CRC4 procedures (if implemented). It also starts searching for IC Frame Synchronization.

For Modes 1, 2 and 3, when each end detects frame alignment and detects the A bit set to 1 in the reverse direction it also **shall** stop Timers TAfa and TCfa.

REV: REVision level (Octet 7, Bits 5-7)

This field can be used to check version compatibility between the local and far end units. If this field is not used it **shall** be set to all binary ones on the transmitting end and ignored on the receiving end. All endpoints developed to the main body of Issue 1 of this International Standard **shall** use a Revision Level of 0. Annex D defines the additions necessary to make this into a Revision 1 product which caters for extended digit fields.

This is a binary encoded number in the range $0 \leq REV \leq 7$.

If one CAU transmits a higher value of REV than the other CAU then only those procedures defined for the lower value of REV shall be used by both CAUs.

8.1.2 Additional Channel Setup

For Mode 0, no framing is done on the additional channels; therefore, DEQ Control is not used and no procedures are defined other than setting up the additional channels (Call Control responsibility) and cutting the channels through to the Application as they are connected. When all channels are connected, Timer Txadd01 is stopped.

For Modes 1, 2 and 3, for each additional channel, when each connection is established the Calling Endpoint **shall** begin transmitting the multiframe structure with the Alignment (A) bit in the CRC octet set to zero, the E bit set to 0 and the CRC4 bits set to 1. It also **shall** send all ones in the data octets. The Calling Endpoint starts the TCfa timer.

Upon connection, the Calling Endpoint **shall** send the Information message repeatedly in the Information Channel. The Information message **shall** contain the following field values:

Channel Identifier: Channel Identifier assigned to this channel by the Calling Endpoint.

RI, RL REQ and RL IND: Set to 0.

Group Identifier: The GID assigned to this call by the Answering Endpoint.

XFLAG: Information subchannel.

Digit Fields: Information subchannel.

Other parameters: Set to negotiated values.

The Calling Endpoint **shall** assign the Channel Identifier sequentially beginning with 1 (master channel).

Upon connection, the Answering Endpoint **shall** start looking for multiframe alignment and Information Message Alignment (full bandwidth for the initial channel of a new call) and either begin transmitting all ones on the channel or transmitting multiframe structure with the correct parameters for this channel as follows:

Alignment (A) bit in the CRC octet set to zero.

E bit set to 0.

CRC4 bits set to 1.

All ones in the data octets.

Information Channel messages with the correct Channel ID, Group ID and negotiated parameters.

The Answering Endpoint also starts TAnull.

If the Answering Endpoint detects an Information Message without detecting Frame Synchronization it stops TAnull and recognizes this is the Master Channel of a new call. Procedures in 8.1.1 apply. When the Answering Endpoint detects Frame Alignment, it **shall** stop TAnull and start searching for IC Frame Synchronization. (If TAnull expires, the Answering Endpoint defaults to the Transparent Mode as described in 9.1). When the Answering Endpoint receives an Information Message, it **shall** store the Channel Identifier to be used in sequencing the channels. It also **shall** check the Group Identifier (GID) to identify the call to which the channel belongs. At this point it **shall** start transmitting the multiframe structure (unless it started transmitting on connection in which case it shall continue transmitting) with the A bit set to 1 in the transmit direction and start the CRC4 procedures (if implemented). In addition, it **shall** transmit the following field values in the Information message in the Information Channel:

Channel Identifier: Channel Identifier assigned to this channel by the Answering Endpoint.

RI: Set to 0.

RL REQ, RL IND: Set to 0.

Group Identifier: The GID received from the Calling Endpoint.

XFLAG: Information subchannel.

Digit Fields: Information subchannel.

Other fields: Set to negotiated values.

At this point the Answering Endpoint marks this channel as ready for transmission.

When the Calling Endpoint detects Frame Alignment, it **shall** set the A bit to 1 in the transmit direction and start the CRC4 procedures (if implemented). It also **shall** start searching for IC Frame Synchronization.

When the Calling Endpoint receives a valid Information Message, it **shall** store the Channel Identifier for use in sequencing the channels and marks the channel as ready for transmission.

When transmitting and receiving Information Messages with CID not set to 0, the XFLAG and DIGITS octets (Octets 9-16, Bits 2-7) are available for use as an Information subchannel. The use of this information subchannel is outside the scope of this International Standard and is implementation dependent.

8.1.2.1 Additional Channel Setup - Failure Recovery Procedures

In the event of a failure of one or more channels during additional channel setup, the endpoint can decide either to disconnect the entire call or downspeed the call to a reduced data rate. In this case, failure is defined as Txadd01 expiring before all channels are synchronized or connected. For channels that do not connect or are connected but fail before Txadd01 expires, the Calling Endpoint can choose to disconnect the channel and attempt to reconnect a new channel or can wait for Txadd01 expires and take the recovery action below.

For Modes 1, 2 and 3, if the Endpoint decides to disconnect the call, the Endpoint proceeds with the normal disconnect procedures (8.5). For Mode 0, the endpoint disconnects the channels using call control.

For Modes 1, 2 and 3, if the Endpoint decides to downspeed the call to a reduced rate, it disconnects the channels that have not reached synchronization (through call control signaling), changes the RMULT and SUBMULT fields in the

Information Channel to reflect the new speed, resequences the channels to sequentialize the channel numbers, starts Txdeq and begins to equalize the delays.

For Mode 0, there is no frame synchronization on any of the channels; therefore, the only failure that is detectable is if all the requested channels are not connected when Txadd01 expires. If the Endpoint decides to downspeed the call, the endpoint cuts through the available data channels to the application. For Mode 0, the Endpoint can cut through the connected channels to the application as they are connected.

8.1.3 Delay Equalization

Once each channel has been connected, each end stops Txadd01. Once each channel has been marked ready (i.e., attains local synchronization for each channel), the endpoint starts Txdeq and uses the FC to measure the relative delay variance between the individual channels of the N x 56/64 call. Also, since the incoming call arrival sequence cannot be assumed to be equal to the call setup sequence, each end **shall** use the received Channel Identifiers (CID) to order the channels properly . When each end has resequenced and equalized the delay between channels for the call, it **shall** transmit a remote indication (RI = 1) in the Information message in all channels to the other end. All other parameters **shall** remain the same. For Modes 2 and 3, when each end receives RI = 1 in the Information Message in all channels, it **shall** consider call setup to be complete, stops Txdeq and commences user data transfer.

In Mode 1, when each end receives RI=1 in each channel, it **shall** begin transmitting A=0 (RI remains set to 1) on all bearer channels indicating readiness to move to Mode 1 data transfer. Having transmitted A=0, each end waits to receive A=0 in each bearer channel. When each end receives A=0 in each bearer channel, it **shall** stop transmitting framing pattern and consider the call setup to be complete. At this time each end **shall** remove the multiframe structure from all channels. Also, after transmitting RI=1 and A=0, if an endpoint recognizes loss of frame synchronization on all bearer channels before it receives A=0, it **shall** consider call setup to be complete and **shall** remove the multiframe structure from the bearer channels. In both cases, the channels **shall** be available to transmit and receive user data over the full available bandwidth. In Mode 1, error conditions on one or more channels that disturb overall frame alignment and connection quality cannot be recognized automatically via the framing structure and procedures defined in this International Standard. Error recovery for Mode 1 is beyond the scope of these procedures (Recovery can consist of manual or external intervention).

NOTE - Due to delay and line errors, each end should transmit more than one frame with RI=1 and A=0 before removing the multiframe structure independent of whether or not the end has received RI=1. The endpoints can use a timer or can transmit a fixed number of Information messages before removing framing.

In Modes 2 and 3, each endpoint **shall** continuously monitor each channel of the call for delay equalization and frame alignment. Each endpoint also can continuously monitor each channel or a subset of the channels for end-to-end bit error rate. Error recovery procedures are described in 8.7 below.

When each endpoint is transmitting RI=1 and detects RI=1 (i.e., delay equalization complete) it stops Txdeq.

8.1.4 Timeout Conditions

The endpoint **shall** take the following actions based on the expiration of the associated timer:

TAnull: The Answering Endpoint **shall** default to the Transparent Mode as described in 9.1.

TCnull: The Calling Endpoint **shall** default to the Transparent Mode as described in 9.2.

Txfa: If this is the master channel, the endpoint **shall** either disconnect the call or initiate the channel recovery procedures (See 8.7.1). If this is not the master channel the endpoint **shall** disconnect the channel. Channel disconnection consists of disconnecting the channel through call control signaling during a pre-Active phase or through channel deletion procedures during the Active phase. Other actions for recovery are implementation dependent and can include:

Disconnecting the entire call

Running at the lower speed

Retrying the failed channel.

Txinit: If this timer expires, it means the complete call has not been setup. Actions taken on expiration of this timer are implementation dependent and can include the disconnection of the call.

8.2 Adding Bandwidth to an Existing Call

Bandwidth Addition is only possible in Modes 2 and 3. In Mode 2, Bandwidth Addition is only possible by adding additional channels. In Mode 3, it is possible to add bandwidth without adding additional channels by increasing the number of data bits allowed in the Nth channel.

Bandwidth can be added to an existing call in increments of one or more channels to increase the user data rate without tearing down the original call.

NOTE - Depending on implementation, user data might be lost during the addition of a channel.

8.2.1 Negotiation (Calling Endpoint Initiated)

When the Calling Endpoint wants to set up additional bandwidth, it **shall** send an Information Message on all channels with the CID set to 0 (negotiation flag) and the RMULT and SUBMULT fields set to the new value. The Calling Endpoint **shall** request a phone number (if a new channel is needed) by setting the XFLAG field to Decimal 1 and the Digits fields to all 1's. The rest of the parameters **shall** be set to the negotiated values. The Calling Endpoint **shall** start TCadd. If a new channel is not requested (i.e., in Mode 3 increasing the bandwidth used in the Nth Channel), then the XFLAG and Digits fields is set to all 1's.

When the Answering Endpoint receives an Information Message on all channels with CID=0, it recognizes the message as a negotiation message and checks the RMULT and SUBMULT fields. Bandwidth addition is indicated by an increased RMULT and/or SUBMULT (Mode 3 only) field value. The Answering Endpoint **shall** either accept the requested value by returning the same value, or deny the request by returning the current value (including a relevant Cause Code). In either case, it **shall** return a channel identifier of 0 (negotiation flag) to indicate its response. If the Answering Endpoint accepts the request for a additional bandwidth, the Answering Endpoint **shall** return the phone number in the manner described in 8.1.1. If the RMULT and SUBMULT values require more than one channel to be added, the Calling Endpoint **shall** request one phone number for each additional channel.

When the Calling Endpoint receives the Answering Endpoint's response (i.e., CID=0, parameters set to negotiated values), the Calling Endpoint checks the parameters. If the Answering Endpoint rejected the request, the Calling Endpoint **shall** either disconnect the call or maintain the call at the current rate. In either case, the Calling Endpoint **shall** begin transmitting a channel identifier of 1 with the rest of the parameters set to their negotiated values. (The Calling Endpoint transmits the correct non-zero Channel Identifier in each channel).

When the Answering Endpoint receives an Information Message with CID=1 (non-zero Channel Identifier in each channel), it **shall** respond with the CID=1 (non-zero Channel Identifier in each channel) and with the parameters set to their negotiated values. The Answering Endpoint **shall** start TAadd02 and wait for the additional channels. If no additional channels are required, the Answering Endpoint does not start TAadd02, but starts transmitting/receiving data with the new values of L and M (see 7.1.1.1).

After receipt of an Information Message with CID=1 and an acknowledgment of acceptance of the additional channel by the Answering Endpoint, the Calling Endpoint **shall** initiate the connection of the new channel using the phone number received from the Answering Endpoint and starts TCadd02. If no additional channels are required, the Calling Endpoint does not start TCadd02, but starts transmitting/receiving data with the new values of L and M (see 7.1.1.1).

If the Answering Endpoint rejected the request, the Calling Unit **shall** stop its timers.

8.2.2 Negotiation (Answering Endpoint Initiated)

The Answering Endpoint can also initiate negotiation for additional bandwidth. The Answering Endpoint **shall** request additional bandwidth by transmitting an Information Message on all channels with the CID set to 0 and the RMULT and SUBMULT fields set to the desired rate (as described previously). The Answering Endpoint **shall** start TAadd.

When the Calling Endpoint receives an Information Message with the CID = 0 and the RMULT and SUBMULT fields indicating a request for additional bandwidth on all channels, it **shall** either accept or reject the request. The Calling Endpoint **shall** indicate a rejection of the request by sending an Information Message with CID = 0, the RMULT and SUBMULT fields set to the current value and a Cause Code indicating the reason for rejection. The Calling Endpoint **shall** indicate an acceptance of the request by transmitting an Information Message with CID = 0 and the new value of the RMULT and SUBMULT. In addition, if additional channels are required, the Calling Endpoint **shall** request new phone numbers as described for the initial call.

At this point, additional channel setup is as defined when initiated by the Calling Endpoint.

If the Calling Endpoint rejects the request for additional channel setup, the Answering Endpoint **shall** stop its timers.

8.2.3 Additional Channel Setup

When the Calling Endpoint connects the new channel, it **shall** begin transmitting the multiframe structure with the Alignment (A) bit in the CRC octet set to zero, the E bit set to 0, the CRC4 bits set to all ones and all ones in the data octets. It also **shall** begin transmitting an Information Message with the following field values:

Channel Identifier: Channel Identifier assigned to this channel by the Calling Endpoint.

RI, RL REQ and RL IND: Set to 0.

MFG: Set to 0.

Group Identifier: The GID assigned to this call by the Answering Endpoint.

XFLAG: Information subchannel.

Digit Fields: Information subchannel.

Other parameters: Set to negotiated values.

The Calling Endpoint also **shall** begin the search for frame alignment and start Timer TCfa.

When the Answering Endpoint connects the new channel, it **shall** take the following actions:

- transmit one of the following on the channel:
  - All ones
  - Multiframe structure with the Alignment (A) bit in the CRC octet set to zero, the E bit set to 0, the CRC4 bits set to all ones and all ones in the data octets. It also **shall** begin transmitting an Information Message with the following field values:

    Channel Identifier: Channel Identifier assigned to this channel by the Answering Endpoint.

    RI, RL REQ and RL IND: Set to 0.

    Group Identifier: The GID assigned to this call by the Answering Endpoint.

    XFLAG: Information subchannel.

    Digit Fields: Information subchannel

    Other parameters: Set to negotiated values.
- start TAnull and
- start searching for frame alignment.

When it detects multiframe alignment from the Calling Endpoint, it **shall** stop TAnull and start searching for the Information Message. When the Answering Endpoint receives an Information Message it **shall** check the GID field for the Group Identifier that identifies the call to which this channel belongs. If the GID does not match an existing call, the Answering Endpoint **shall** disconnect the channel. If the GID does match an existing call, the Answering Endpoint **shall** add this channel to the correct call and start transmitting the multiframe structure in the channel with the A bit set to 1 and start the CRC procedure (if implemented). The Answering Endpoint also **shall** start transmitting (or continue to transmit) an Information Message with the fields set to the following values:

Channel Identifier: Channel Identifier assigned to this channel by the Answering Endpoint.

RI, RL REQ and RL IND: Set to 0. These values can change based on subsequent actions.

Group Identifier: The GID assigned to this call by the Answering Endpoint.

MFG: Set to 0.

XFLAG: Information subchannel.

Digit Fields: Information subchannel.

Other parameters: Set to negotiated values.

If the Answering Endpoint loses delay equalization , it **shall** start transmitting RI = 0 in all channels. The Answering Endpoint **shall** use the received CID to sequence the channels in the call and **shall** use the FC to equalize the delay between the new channel and the existing channels.

Once the Calling Endpoint establishes frame alignment, it **shall** set the Alignment bit (A) in the CRC octet to one in the transmit direction and begin the CRC4 procedure (if implemented) and **shall** stop Txfa.

If the Calling Endpoint loses delay equalization, it **shall** start transmitting RI = 0 in all channels. The Calling Endpoint **shall** use the received Channel Identifier to sequence the channels and **shall** use the FC to equalize the delay between the new channel and the existing channels.

Once each end has successfully resequenced and delay equalized, each end **shall** transmit a remote indication (RI = 1) over the IC of all channels.

Once each end receives an Information Message with RI = 1 in all channels, it **shall** consider call setup to be complete and **shall** stop its timers.

8.2.4 Timeout Conditions

The endpoint **shall** take the following actions based on the expiration of the associated timer:

TAnull: The Answering Endpoint **shall** default to the Transparent Mode as described in 9.1.

Txfa: If this is the master channel, the endpoint **shall** either disconnect the call or initiate the master channel recovery procedures (See 8.6.1). If this is not the master channel the endpoint **shall** disconnect the channel. Other actions for recovery are implementation dependent and can include:

Disconnecting the entire call

Running at the lower speed

Retrying the failed channel.

Txadd: If this timer expires, it means the endpoint has not received an acknowledgement for the add channel request. Actions taken on expiration of this timer is implementation dependent and can include the disconnection of the call.

TAadd02: If this timer expires, it means that the additional channel setup has not been completed. The endpoint **shall** initiate disconnect procedures for that channel and **shall** begin transmitting the correct values in the RMULT and SUBMULT fields. Other recovery actions are implementation dependent.

8.3 Deleting Bandwidth from an Existing Call

Bandwidth can be deleted from an existing call to decrease the user data rate without tearing down the entire call. Bandwidth Deletion is only available for Modes 2 and 3. Either side can initiate a request to delete one or more channels from the call. For Mode 2, Bandwidth Deletion always implies deletion of at least one channel. For Mode 3, an endpoint can request a reduction of bandwidth without deleting a channel by reducing the number of bits used in the Nth channel.

8.3.1 Calling Endpoint Initiated

The Calling Endpoint **shall** initiate a request to delete bandwidth by transmitting an Information Message over the IC of all channels with the CID set to 0 and the RMULT and SUBMULT fields set to the new value. If only one channel is to be deleted and the Channel Identifier is available, the XFLAG field **shall** contain the Channel Identifier (as assigned by the Calling Endpoint) of the channel to be deleted. If more than one channel is to be deleted, they shall be channels with the highest CID values as assigned by the Calling Endpoint. The Digits fields **shall** contain either a Cause code or PAD characters. The rest of the parameters **shall** contain the negotiated values. The Calling Endpoint **shall** start the TCdel timer .

When the Answering Endpoint receives an Information Message on all channels with CID=0 and with the RMULT and SUBMULT fields indicating deletion of bandwidth, it **shall** either accept the request by returning a channel identifier of 0 (negotiation flag) and the same values for RMULT, SUBMULT and XFLAG as were received or reject the request by returning the original parameter values and a Cause Code with the reason for rejection.

When the Calling Endpoint receives an Information Message with CID = 0 and the channel delete acknowledgment, it **shall** start transmitting Information Messages on all channel with CID set to the correct value and the new parameters.

When the Answering Endpoint receives an Information Message on all channels with CID set to the assigned value and the new RMULT and SUBMULT, it **shall** transmit Information Messages with CID set to the assigned valueand the new parameters. At this point, the Answering Endpoint **shall** perform the following actions:

1. If a channel is to be deleted, the Answering Endpoint **shall** remove the negotiated channel from the call (i.e., stop transmitting and receiving on it).
2. If delay equalization is lost, start Transmitting RI = 0 on all channels.
3. Resequence the Channel Identifiers (if needed). When resequencing Channel Identifiers, the Channel Identifiers **shall** maintain their relative order. Thus for example, if the channel with the highest CID is lost, the CID's for the rest of the channel remain the same. If a middle channel is deleted, the channels with CID's higher than the deleted channel are moved down by 1.
4. Start Equalizing the delays on all channels (if it was lost).

From this point forward, the endpoint acts as described in 8.1.3. Data transfer commences with the new values of L and M (as defined in 7.1).

When the Calling Endpoint receives an Information Message with CID = 1 and the new RMULT and SUBMULT, it **shall** perform the following actions:

1. If a channel is to be deleted, remove the negotiated channel from the call (i.e., stop transmitting and receiving on it).
2. If Delay Equalization is lost, start Transmitting RI = 0 on all channels.
3. Resequence the Channel Identifiers(if needed). When resequencing Channel Identifiers, the Channel Identifiers **shall** maintain their relative order. Thus for example, if the channel with the highest CID is lost, the CID's for the rest of the channel remain the same. If a middle channel is deleted, the channels with CID's higher than the deleted channel are moved down by 1.
4. Start Equalizing the delays on all channels (if needed).
5. Initiate the disconnection of the negotiated channel if a channel is to be deleted.

From this point forward, the endpoint acts as described in 8.1.3. Data transfer commences with the new values of L and M (as defined in 7.1).

NOTE - Depending on implementation, user data can be lost during the removal of a channel.

8.3.2 Answering Endpoint Initiated

The Answering Endpoint **shall** initiate bandwidth deletion by transmitting an Information Message on all channels to the Calling Endpoint with CID = 0, the new RMULT and SUBMULT values. If only one channel is to be deleted the XFLAG field shall contain the CID (as assigned by the Calling Endpoint) of the channel to be deleted. If more than one channel is to be deleted, they shall be the channels with the highest values of CID as assigned by the Calling Endpoint. The digits field shall contain either a Cause code or PAD characters and the Answering Endpoint shall start TAdel.

When the Calling Endpoint receives an Information Message on all channels with CID = 0, and the new RMULT and SUBMULT, it **shall** check the XFLAG field for the CID of the channel to be deleted (if requested). The Calling Endpoint **shall** either: Reject the request by returning CID = 0 and the original parameters and a Cause Code with the reason for the reject, or accept the request by returning the requested parameters.

If the bandwidth delete is accepted, the call proceeds as defined in 8.3.1 except that the Answering unit stops TAdel when it receives non-zero CID in all channels.

Actions taken if the Answering Unit receives a channel delete reject are implementation or user dependent.

8.3.3 Timeout Conditions

The endpoint **shall** take the following actions based on the expiration of the associated timer:

Txdel: If this timer expires, it means the endpoint has not received an acknowledgement for the channel delete request. Implementation dependent actions can include disconnecting the channel and downspeed anyway, maintaining the current speed and notifying management and disconnecting the call.

8.4 Collision Resolution

In the case of a collision (one end requesting bandwidth addition and the other end requesting bandwidth deletion) the bandwidth addition request shall normally take precedence. However, in the case of an attempt to remove (an) errored channel(s), a request for bandwidth deletion shall take precedence.

8.5 Call Disconnection

These procedures describe how to negotiate the disconnection of a call . Either the Calling or the Answering Endpoint can initiate a disconnection. Negotiated call disconnection is only available for Modes 2 and 3.

8.5.1 Calling Endpoint Initiated

The Calling Endpoint **shall** indicate a call disconnect by transmitting an Information Message on all channels with the CID set to 0 (negotiation flag) and the RMULT and SUBMULT set to 0 and **shall** start timer TCdisc.

When the Answering Endpoint receives an Information Message on all channels with the CID set to 0 (negotiation flag) and the RMULT and SUBMULT fields set to 0, it **shall** either accept the call disconnection by returning a CID of 0 (negotiation flag) and a RMULT and SUBMULT of 0 or reject the call disconnection by returning a CID of 0 (negotiation flag) and the current parameter values. The Answering Endpoint may optionally include a Cause code indicating the reason for rejection of the disconnect. If the Answering Endpoint accepts the call disconnection, it **shall** stop transmitting user data over the connection , **shall** wait for disconnect and **shall** start timer TAdisc.

If the Calling Endpoint receives a disconnect acknowledgment, it **shall** begin channel disconnection procedures for all channels associated with the call and **shall** stop TCdisc. If the Calling Endpoint receives a reject of the disconnect request, it **shall** either begin channel disconnection procedures for all channels associated with the call or maintain the current call (this is implementation dependent).

When all channels are disconnected, the Answering Unit stops timer TAdisc.

8.5.2 Answering Endpoint Initiated

The Answering Endpoint **shall** indicate a call disconnect by transmitting an Information Message on the all channels with the CID set to 0 (negotiation flag) and the RMULT and SUBMULT fields set to 0 and **shall** start Timer TAdisc.

When the Calling Endpoint receives an Information Message on all channels with the CID set to 0 (negotiation flag) and the RMULT and SUBMULT fields set to 0, it **shall** either accept the call disconnection by returning a CID of 0 (negotiation flag) and the RMULT and SUBMULT fields set to 0 or reject the call disconnection by returning a CID of 0 (negotiation flag) and the current parameter values. The Calling Endpoint may optionally include a Cause Code indicating the reason for the rejection of the disconnect. If the Calling Endpoint accepts the call disconnection, it **shall** stop transmitting user data over the connection and **shall** start TCdisc.

If the Answering Endpoint receives a disconnect acknowledgment, it **shall** begin channel disconnection procedures for all channels associated with the call and **shall** stop TAdisc. If the Answering Endpoint receives a reject of the disconnect request, it **shall** either begin channel disconnection procedures for all channels associated with the call or maintain the current call (this is implementation dependent).

When all channels are disconnected, the Calling Endpoint stops timer TCdisc.

8.5.3 Timeout Conditions

The endpoints **shall** take the following actions based on expiration of the relevant timer:
Txdisc: The endpoint **shall** initiate disconnection of all channels associated with the call.

8.6 Remote Loopback

Either end can request a remote loopback. Remote Loopback indication is only available for Modes 2 and 3. In the following procedure the side initiating the loopback is called the initiating endpoint and the side receiving the loopback request is called the remote endpoint. The transitions of the RL IND and RL REQ between 1 and 0 occur in all channels and are independent of the CID value.

The initiating endpoint **shall** request that the remote endpoint go into loopback by transmitting an Information Message with the RL REQ bit set to 1 in all channels and starting Timer Tloop.

When the remote endpoint receives an Information Message with the RL REQ bit set to 1, it **shall** either acknowledge it by transmitting an Information Message with the RL IND bit set to 1 in all channels or reject it by returning RL IND = 0.

When the remote endpoint accepts the loopback request (i.e., starts transmitting RL IND =1), it **shall** loop the user data back toward the initiating endpoint at a point after delay equalization (i.e., payload loopback).

If the remote Endpoint does not accept the loopback request, it **shall** continue with user data transmission on all channels.

When the initiating endpoint receives a loopback acknowledgment (i.e., Information Message with RL IND set to 1), it **shall** continue to transfer user data over all channels and stop Timer Tloop. Actual tests using the loopback capability (e.g., Bit Error Rate Tests) are outside the scope of this International Standard..

If the initiating endpoint does not receive a loopback accept (i.e., Information Message with RL IND set to 0) before Tloop expires, it **shall** continue transmitting user data on all channels. Other actions are implementation dependent.

When in a loopback state, if the initiating endpoint receives an Information Message with RL IND set to 0, it **shall** take the call out of loopback mode. It also **shall** start transmitting RL REQ = 0 in the Information message. Other actions are implementation dependent.

When in a loopback state, if the remote Endpoint receives an Information Message with RL REQ set to 0, it **shall** take the call out of loopback mode and start transmitting RL IND = 0 in the Information message. The remote endpoint also starts transmitting user data at this time. Other actions are implementation dependent.

NOTE - Depending on implementation, user data can be lost during remote loopback negotiation.

When an endpoint receives an Information Message with RL IND = 1 when not requesting a loopback, it means that the remote endpoint has gone into loopback mode.

If an endpoint goes into local loopback mode (i.e., looping user data back to the remote endpoint) it **shall** transmit RL IND =1 in all channels. When an endpoint goes out of local loopback mode, it **shall** transmit RL IND = 0 in all channels.

All actions (adding and deleting a channel, and disconnecting the call) are possible in loopback mode.

8.6.1 Timeout Conditions

If Tloop expires, the initiating endpoint **shall** start transmitting normal user data (if available). Other actions are implementation dependent.

8.7 Error Conditions

Call failure procedures depend on the nature of the failure. The following are examples of call failure causes.

High Error Rate (CRC errors or E bit exceeding threshold):

An endpoint can detect a high error rate by checking the CRC field and the E bit. The definition of high error rate is application and implementation dependent.

Extended Loss of Frame Alignment/Inconsistency in Frame Count

An endpoint can detect a loss of frame alignment or an error in the Frame Count. As defined in 6.3, when an endpoint detects loss of frame alignment, it begins searching for frame synchronization and transmitting A = 0. The error condition described here is if frame synchronization is not achieved after an extended period of time (i.e., expiration of Txfa).

Receipt of a non-negotiated Disconnect for one or more channels:

An endpoint can receive a non-negotiated disconnect for one or more channels in a call.

Loss of the Master Channel

An endpoint can lose the use of the Master Channel by any of the above failure modes.

Loss of Delay Equalization

An endpoint detects the loss of delay equalization.

Error recovery mechanisms not specified in these procedures can include the following:

Replace the lost channel using the procedures defined in 8.2 for adding a channel to an existing call, except during call setup.

Continue operating with reduced bandwidth and delete the channel if necessary (8.3), except during call setup.

Disconnect the entire call using the methods described in 8.6, except during call setup.

8.7.1 Loss of Channel (Other Than During Call Setup)

An endpoint can detect loss of a channel due to the following conditions:

Receipt of disconnect: The endpoint can receive a network disconnect for the master channel without any prior disconnect negotiation. The endpoint completes the disconnect procedure and starts recovery procedures.

Extended Loss of Frame Synchronization: The endpoint can detect extended loss of frame synchronization. This is detected by expiration of Txfa. The endpoint disconnects the channel and starts recovery procedures.

High Error Count: When the endpoint detects excessive errors on the master channel (CRC or E bit) it disconnects the channel and starts recovery procedures.

If an endpoint loses a channel (as described above) and wants to recover , it **shall** recover by taking the following action:

1. Start transmitting RI = 0 in all remaining channels (if delay equalization is lost).

2. Resequence the Channel Identifiers. As mentioned earlier, the Channel Identifiers **shall** maintain their relative order.

3. Re-equalize the delays (as described in 8.1.3)

The parameters passed in the Information message **shall** reflect the new status of the call (i.e., New RMULT and SUBMULT values).

When recovering from a failed channel, the endpoint **shall** drop back to the next lowest value of RMULT and a SUBMULT value of 0 when transmitting/receiving data over the interface. The endpoint can immediately begin negotiations for recovery of bandwidth using the normal means.

If the endpoint chooses not to recover, it **shall** disconnect the call.

Procedures for Loss of Channel during call setup are described in 8.1.

8.7.2 Loss of Delay Equalization

When an endpoint detects loss of delay equalization during the active phase of the call, it **shall** send RI = 0 in all channels and attempt to resynchronize the call as described in 8.1.3.

When an endpoint receives RI = 0 in all channels during the active phase of the call, it means that the remote endpoint has lost delay equalization. Actions taken by the endpoint on receipt of RI=0 are implementation dependent.

9 Transparent Mode Operation

This mode of operation is necessary when one endpoint is a Channel Aggregation Unit and the other endpoint is not. In this mode, network channels are "cut through" to the applications served by the Channel Aggregation Unit. In this context, "cut through" means that the channel aggregation functionality is bypassed. Channels are connected to the served application without the Channel Aggregation Unit ever recognizing Channel Aggregation Framing or establishing Information Channel communication. This mode makes the endpoint which is a Channel Aggregation Unit act as if it is not a Channel Aggregation Unit, thus providing compatibility between the two endpoints. In Transparent Mode, the Channel Aggregation Unit does not alter the bit stream received from the network.

There are two cases where the Transparent Mode is invoked, depending on whether the Calling or Answering Endpoint is the Channel Aggregation Unit. 9.1 describes the case where the Answering Endpoint is a Channel Aggregation Unit while the Calling Endpoint is not. 9.2 describes the case where the Calling Endpoint is a Channel Aggregation Unit while the Answering Endpoint is not.

9.1 Non Aggregating Unit Calling Aggregating Unit

The Transparent Mode **shall** be the default mode that is invoked when neither the Channel Aggregation nor the Information Channel framing is detected by the Answering Endpoint within the period specified by timer TAnull.

In this case the Answering Endpoint is a Channel Aggregation Unit, while the Calling Endpoint is not. When the Answering Endpoint answers a channel, it starts Timer TAnull. Timer TAnull runs while the endpoint is determining the type of call. Timer TAnull is stopped when the Answering Endpoint detects either Channel Aggregation Framing or Information Channel Framing on the channel.

If TAnull expires, the channel shall be "cut through" to the application, thereby providing a clear channel path between the attached application and the Calling Endpoint. What happens at this stage is outside the scope of this International Standard and is controlled by the attached application (and Calling Endpoint).

For Channel Aggregation Units intended for use in video applications, the Transparent Mode is mandatory. A capability can optionally be provided to allow the user to manually disable the mandatory Transparent Mode for non-video applications and for video applications using more than 6 channels. In this case, the channel is disconnected after expiration of TAnull. The Transparent mode shall always be enabled for video applications which operate using 6 or fewer bearer channels.

For Channel Aggregation Units not intended for use in video applications, the transparent mode is optional.

9.2 Aggregating Unit Calling Non Aggregating Unit

For an aggregating unit calling a non aggregating unit, the Calling Endpoint **shall** support the Transparent Mode using at least one of the following two methods.

Method 1:

The Calling Endpoint chooses to use the Transparent Mode directly. The Calling Endpoint never attempts to establish Information Channel Framing, but instead cuts though the channels to the application as soon as the channels are established.

Method 2:

The Calling Endpoint starts Timer TCnull when it connects a channel. TCnull is stopped when Information Channel Framing is detected. If TCnull expires and the Calling Endpoint has not been able to recognize Information Channel Framing coming from the Answering Endpoint, the channel shall be "cut through" to the application.

Annex A
(normative)

RMULT and SUBMULT Values

The following table contains values for RMULT and SUBMULT for BCRs of 56 kbit/s and 64 kbit/s and for number of channels up to 24.

Table A.1 - Bearer Channel Rate (BCR) = 56 kbit/s

A	RMULT	SUB MULT	N	L	M
8000	0	1	1	10	1
16000	0	2	1	19	2
24000	0	3	1	28	3
32000	0	4	1	37	4
40000	0	5	1	46	5
48000	0	6	1	55	6
56000	1	0	2	1	7
64000	1	1	2	11	1
72000	1	2	2	20	2
80000	1	3	2	29	3
88000	1	4	2	38	4
96000	1	5	2	47	5
104000	1	6	2	56	6
112000	2	0	3	2	7
120000	2	1	3	12	1
128000	2	2	3	21	2
136000	2	3	3	30	3
144000	2	4	3	39	4
152000	2	5	3	48	5
160000	2	6	3	57	6
168000	3	0	4	3	7
176000	3	1	4	13	1
184000	3	2	4	22	2
192000	3	3	4	31	3
200000	3	4	4	40	4
208000	3	5	4	49	5
216000	3	6	4	58	6
224000	4	0	5	4	7
232000	4	1	5	14	1
240000	4	2	5	23	2
248000	4	3	5	32	3
256000	4	4	5	41	4
264000	4	5	5	50	5
272000	4	6	5	59	6
280000	5	0	6	5	7
288000	5	1	6	15	1
296000	5	2	6	24	2
304000	5	3	6	33	3
312000	5	4	6	42	4
320000	5	5	6	51	5
328000	5	6	6	60	6
336000	6	0	7	6	7
344000	6	1	7	16	1
352000	6	2	7	25	2

A	RMULT	SUB MULT	N	L	M
360000	6	3	7	34	3
368000	6	4	7	43	4
376000	6	5	7	52	5
384000	6	6	7	61	6
392000	7	0	8	7	7
400000	7	1	8	17	1
408000	7	2	8	26	2
416000	7	3	8	35	3
424000	7	4	8	44	4
432000	7	5	8	53	5
440000	7	6	8	62	6
448000	8	0	9	8	7
456000	8	1	9	18	1
464000	8	2	9	27	2
472000	8	3	9	36	3
480000	8	4	9	45	4
488000	8	5	9	54	5
496000	8	6	9	63	6
504000	9	0	10	9	7
512000	9	1	10	19	1
520000	9	2	10	28	2
528000	9	3	10	37	3
536000	9	4	10	46	4
544000	9	5	10	55	5
552000	9	6	11	1	6
560000	10	0	11	10	7
568000	10	1	11	20	1
576000	10	2	11	29	2
584000	10	3	11	38	3
592000	10	4	11	47	4
600000	10	5	11	56	5
608000	10	6	12	2	6
616000	11	0	12	11	7
624000	11	1	12	21	1
632000	11	2	12	30	2
640000	11	3	12	39	3
648000	11	4	12	48	4
656000	11	5	12	57	5
664000	11	6	13	3	6
672000	12	0	13	12	7
680000	12	1	13	22	1
688000	12	2	13	31	2
696000	12	3	13	40	3
704000	12	4	13	49	4

Table A.1 (cont)

A	RMULT	SUB MULT	N	L	M
712000	12	5	13	58	5
720000	12	6	14	4	6
728000	13	0	14	13	7
736000	13	1	14	23	1
744000	13	2	14	32	2
752000	13	3	14	41	3
760000	13	4	14	50	4
768000	13	5	14	59	5
776000	13	6	15	5	6
784000	14	0	15	14	7
792000	14	1	15	24	1
800000	14	2	15	33	2
808000	14	3	15	42	3
816000	14	4	15	51	4
824000	14	5	15	60	5
832000	14	6	16	6	6
840000	15	0	16	15	7
848000	15	1	16	25	1
856000	15	2	16	34	2
864000	15	3	16	43	3
872000	15	4	16	52	4
880000	15	5	16	61	5
888000	15	6	17	7	6
896000	16	0	17	16	7
904000	16	1	17	26	1
912000	16	2	17	35	2
920000	16	3	17	44	3
928000	16	4	17	53	4
936000	16	5	17	62	5
944000	16	6	18	8	6
952000	17	0	18	17	7
960000	17	1	18	27	1
968000	17	2	18	36	2
976000	17	3	18	45	3
984000	17	4	18	54	4
992000	17	5	18	63	5
1000000	17	6	19	9	6
1008000	18	0	19	18	7

A	RMULT	SUB MULT	N	L	M
1016000	18	1	19	28	1
1024000	18	2	19	37	2
1032000	18	3	19	46	3
1040000	18	4	19	55	4
1048000	18	5	20	1	5
1056000	18	6	20	10	6
1064000	19	0	20	19	7
1072000	19	1	20	29	1
1080000	19	2	20	38	2
1088000	19	3	20	47	3
1096000	19	4	20	56	4
1104000	19	5	21	2	5
1112000	19	6	21	11	6
1120000	20	0	21	20	7
1128000	20	1	21	30	1
1136000	20	2	21	39	2
1144000	20	3	21	48	3
1152000	20	4	21	57	4
1160000	20	5	22	3	5
1168000	20	6	22	12	6
1176000	21	0	22	21	7
1184000	21	1	22	31	1
1192000	21	2	22	40	2
1200000	21	3	22	49	3
1208000	21	4	22	58	4
1216000	21	5	23	4	5
1224000	21	6	23	13	6
1232000	22	0	23	22	7
1240000	22	1	23	32	1
1248000	22	2	23	41	2
1256000	22	3	23	50	3
1264000	22	4	23	59	4
1272000	22	5	24	5	5
1280000	22	6	24	14	6
1288000	23	0	24	23	7
1296000	23	1	24	33	1
1304000	23	2	24	42	2
1312000	23	3	24	51	3
1320000	23	4	24	60	4

Table A.2 Bearer Channel Rate (BCR)= 64 kbit/s

A	RMULT	SUB MULT	N	L	M
8000	0	1	1	8	8
16000	0	2	1	16	8
24000	0	3	1	24	8
32000	0	4	1	32	8
40000	0	5	1	40	8
48000	0	6	1	48	8
56000	0	7	1	56	8
64000	1	0	2	1	8
72000	1	1	2	9	8
80000	1	2	2	17	8
88000	1	3	2	25	8
96000	1	4	2	33	8
104000	1	5	2	41	8
112000	1	6	2	49	8
120000	1	7	2	57	8
128000	2	0	3	2	8
136000	2	1	3	10	8
144000	2	2	3	18	8
152000	2	3	3	26	8
160000	2	4	3	34	8
168000	2	5	3	42	8
176000	2	6	3	50	8
184000	2	7	3	58	8
192000	3	0	4	3	8
200000	3	1	4	11	8
208000	3	2	4	19	8
216000	3	3	4	27	8
224000	3	4	4	35	8
232000	3	5	4	43	8
240000	3	6	4	51	8
248000	3	7	4	59	8
256000	4	0	5	4	8
264000	4	1	5	12	8
272000	4	2	5	20	8
280000	4	3	5	28	8
288000	4	4	5	36	8
296000	4	5	5	44	8
304000	4	6	5	52	8
312000	4	7	5	60	8
320000	5	0	6	5	8
328000	5	1	6	13	8
336000	5	2	6	21	8
344000	5	3	6	29	8
352000	5	4	6	37	8
360000	5	5	6	45	8
368000	5	6	6	53	8
376000	5	7	6	61	8
384000	6	0	7	6	8
392000	6	1	7	14	8
400000	6	2	7	22	8
408000	6	3	7	30	8
416000	6	4	7	38	8
424000	6	5	7	46	8
432000	6	6	7	54	8
440000	6	7	7	62	8
448000	7	0	8	7	8
456000	7	1	8	15	8
464000	7	2	8	23	8
472000	7	3	8	31	8

A	RMULT	SUB MULT	N	L	M
480000	7	4	8	39	8
488000	7	5	8	47	8
496000	7	6	8	55	8
504000	7	7	8	63	8
512000	8	0	9	8	8
520000	8	1	9	16	8
528000	8	2	9	24	8
536000	8	3	9	32	8
544000	8	4	9	40	8
552000	8	5	9	48	8
560000	8	6	9	56	8
568000	8	7	10	1	8
576000	9	0	10	9	8
584000	9	1	10	17	8
592000	9	2	10	25	8
600000	9	3	10	33	8
608000	9	4	10	41	8
616000	9	5	10	49	8
624000	9	6	10	57	8
632000	9	7	11	2	8
640000	10	0	11	10	8
648000	10	1	11	18	8
656000	10	2	11	26	8
664000	10	3	11	34	8
672000	10	4	11	42	8
680000	10	5	11	50	8
688000	10	6	11	58	8
696000	10	7	12	3	8
704000	11	0	12	11	8
712000	11	1	12	19	8
720000	11	2	12	27	8
728000	11	3	12	35	8
736000	11	4	12	43	8
744000	11	5	12	51	8
752000	11	6	12	59	8
760000	11	7	13	4	8
768000	12	0	13	12	8
776000	12	1	13	20	8
784000	12	2	13	28	8
792000	12	3	13	36	8
800000	12	4	13	44	8
808000	12	5	13	52	8
816000	12	6	13	60	8
824000	12	7	14	5	8
832000	13	0	14	13	8
840000	13	1	14	21	8
848000	13	2	14	29	8
856000	13	3	14	37	8
864000	13	4	14	45	8
872000	13	5	14	53	8
880000	13	6	14	61	8
888000	13	7	15	6	8
896000	14	0	15	14	8
904000	14	1	15	22	8
912000	14	2	15	30	8
920000	14	3	15	38	8
928000	14	4	15	46	8
936000	14	5	15	54	8
944000	14	6	15	62	8

Table A.2 (cont)

A	RMULT	SUB MULT	N	L	M
952000	14	7	16	7	8
960000	15	0	16	15	8
968000	15	1	16	23	8
976000	15	2	16	31	8
984000	15	3	16	39	8
992000	15	4	16	47	8
1000000	15	5	16	55	8
1008000	15	6	16	63	8
1016000	15	7	17	8	8
1024000	16	0	17	16	8
1032000	16	1	17	24	8
1040000	16	2	17	32	8
1048000	16	3	17	40	8
1056000	16	4	17	48	8
1064000	16	5	17	56	8
1072000	16	6	18	1	8
1080000	16	7	18	9	8
1088000	17	0	18	17	8
1096000	17	1	18	25	8
1104000	17	2	18	33	8
1112000	17	3	18	41	8
1120000	17	4	18	49	8
1128000	17	5	18	57	8
1136000	17	6	19	2	8
1144000	17	7	19	10	8
1152000	18	0	19	18	8
1160000	18	1	19	26	8
1168000	18	2	19	34	8
1176000	18	3	19	42	8
1184000	18	4	19	50	8
1192000	18	5	19	58	8
1200000	18	6	20	3	8
1208000	18	7	20	11	8
1216000	19	0	20	19	8
1224000	19	1	20	27	8
1232000	19	2	20	35	8
1240000	19	3	20	43	8
1248000	19	4	20	51	8
1256000	19	5	20	59	8
1264000	19	6	21	4	8
1272000	19	7	21	12	8
1280000	20	0	21	20	8
1288000	20	1	21	28	8
1296000	20	2	21	36	8
1304000	20	3	21	44	8
1312000	20	4	21	52	8
1320000	20	5	21	60	8
1328000	20	6	22	5	8
1336000	20	7	22	13	8
1344000	21	0	22	21	8
1352000	21	1	22	29	8
1360000	21	2	22	37	8
1368000	21	3	22	45	8
1376000	21	4	22	53	8
1384000	21	5	22	61	8
1392000	21	6	23	6	8
1400000	21	7	23	14	8
1408000	22	0	23	22	8
1416000	22	1	23	30	8
1424000	22	2	23	38	8
1432000	22	3	23	46	8
1440000	22	4	23	54	8
1448000	22	5	23	62	8
1456000	22	6	24	7	8
1464000	22	7	24	15	8
1472000	23	0	24	23	8
1480000	23	1	24	31	8
1488000	23	2	24	39	8
1496000	23	3	24	47	8
1504000	23	4	24	55	8
1512000	23	5	24	63	8

Annex B
(normative)

Manufacturer's Identifiers

Assignment of manufacturer's IDs shall use the following process. The first two digits (8 bits) shall be used for a country code. The country codes shall be the 8 bit fields defined by CCITT Recommendation T.35. The next two digits (8 bits) shall be reserved and set to zero. The last three digits (12 bits) shall be used to code individual manufacturers.

The allocation of individual manufacturer's IDs shall be administered by the appropriate National Body.

Assignment of an appropriate manufacturer's ID is only required if proprietary extensions to this International Standard are to be implemented.

The coding process is as shown below.

	b1	b2	b3	b4	b5	b6	b7	b8
				Country Code				
Octet 10	1	1	1	b8	b7	b6	b5	1
Octet 11	1	1	1	b4	b3	b2	b1	1
Octet 12	1	1	1	0	0	0	0	1
Octet 13	1	1	1	0	0	0	0	1
				Manufacturer Code				
Octet 14	1	1	1	b12	b11	b10	b9	1
Octet 15	1	1	1	b8	b7	b6	b5	1
Octet 16	1	1	1	b4	b3	b2	b1	1

Annex C
(normative)

Phone Number Length Extension

Introduction

An additional capability is required to allow for more than seven digits in the phone number in the Information Channel. This capability is optional, but shall be used whenever phone numbers longer than 7 digits are to be exchanged in the Information Channel.

Equipment providing this facility will be defined as Revision level 1.

Equipment supporting revision level 1 must be capable of interworking with Revision level 0 implementations.

Realisation

Bits b2 and b3 of Octet 10 (Digit - 1) in the DIGITS field of Information Channel Frames are defined as follows:

b2	b3	Meaning
0	1	Continuity of DIGITS field (odd order)
1	0	Continuity of DIGITS field (even order)
1	1	Last of DIGITS field

Example

The following illustrates an example of transmitting/receiving the phone numbers "12345678901234" and "987654321" contiguously.

			DIGITS							
Sender	Message	XFLAG	(b2,b3)	1	2	3	4	5	6	7
CU-->AU	DN REQ	1	(1,1)	PAD	PAD	PAD	PAD	PAD	PAD	PAD →
CU<--AU	DN ACK	1	(0,1)	1	2	3	4	5	6	7
CU-->AU	DN REQ CONT	1	(0,1)	1	2	3	4	5	6	7
CU<--AU	DN ACK	1	(1,1)	8	9	0	1	2	3	4
CU-->AU	DN REQ	2	(1,1)	PAD	PAD	PAD	PAD	PAD	PAD	PAD
CU<--AU	DN ACK	2	(0,1)	9	8	7	6	5	4	3
CU-->AU	DN REQ CONT	2	(0,1)	9	8	7	6	5	4	3
CU<--AU	DN ACK	2	(1,1)	2	1	EON	EON	EON	EON	EON

NOTE - the use of the technique described in 8.1.1 may be used to reduce the number of digits to be transmitted between Answering and Calling Endpoints.

Annex D
(informative)

State Machine Description of Channel Aggregation Control

This annex describes the Channel Aggregation control procedures using a State Machine approach. It includes Specification Design Language diagrams as well as definitions of each state and messages and events allowed in each state. The State Machine described in this annex is a logical model meant to describe the actions of the endpoint at the interface to the network. The endpoint is not required to implement this model exactly as written as long as the interactions at the network interface are the same as the interactions on the network interface as defined in this State Machine.

In the following description, all references to transmitting or receiving CID=1 mean that the proper Channel Identifier (i.e., not Channel ID 0) is transmitted or received in all channels on the call (not just channel 1). All references to transmitting or receiving CID=0 mean that CID=0 is transmitted or received in all channels.

D.1 Reference Architecture

This subclause describes the reference architecture used to describe the state machine. This architecture is a logical model only and is not meant to restrict implementations .

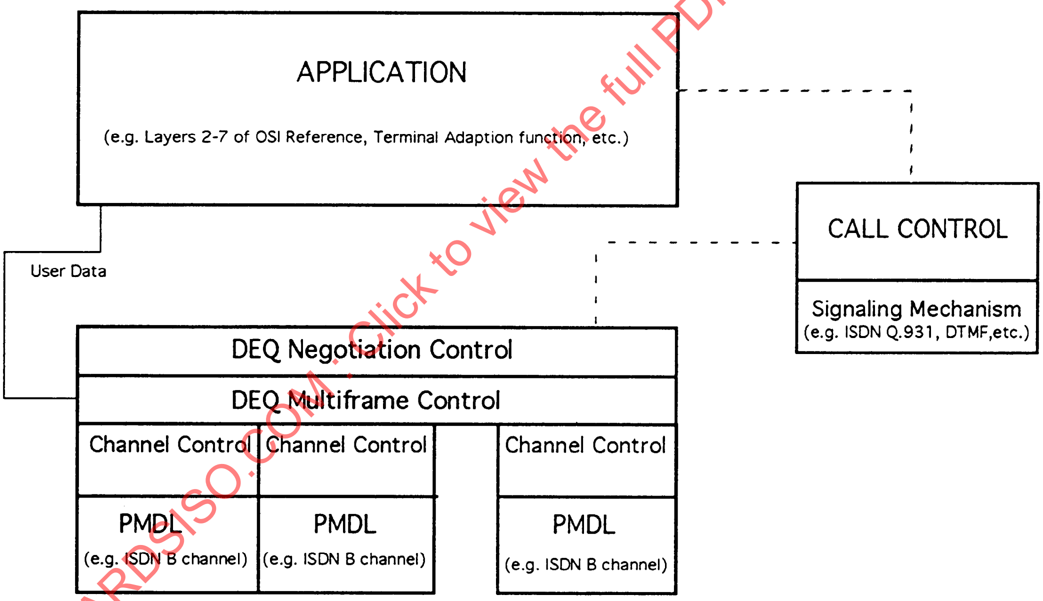


Figure D.1 - Reference Architecture

In this reference architecture, the following entities are defined:

Application:

This is the application using the Delay Equalization algorithms defined in this International Standard. For example, it can consist of Layers 2 - 7 of the OSI Protocol Reference Model (in which case, the Delay Equalization algorithm is acting at Layer 1) or it can consist of a Terminal Adaptation function for inverse multiplexing for an attached video codec or LAN router. The actions of the application is outside the scope of this International Standard.

DEQ NEGOTIATION CONTROL:

This function contains most of the negotiation control functions. This function controls the parameter negotiation in the Information Channel for initial call negotiation and adding and deleting bandwidth during a call.

DEQ MULTIFRAME CONTROL:

This function contains most of the Delay Equalization control functions. It is where the Delay Equalization State Machine resides. This function controls the Multiframe Delay Equalization across all channels. The DEQ Multiframe Control function provides the physical layer interface to the application.

Channel Control:

This function controls the synchronization of the actual individual physical channels. It maintains channel framing and synchronization and provides the Information Channel and data channels for the DEQ Control function in an individual channel. It also interfaces to the actual physical media control (i.e., device drivers).

Physical Media Dependent Layer (PMDL):

This function controls the actual physical media. The definition of this function is outside the scope of this International Standard. The algorithms defined in this International Standard are independent of the electrical characteristics and formatting of the physical media as long as the physical media provides 56 kbit/s or 64 kbit/s channels. Examples of the physical connection are Switched 56 DDS lines, T1 lines with Robbed Bit Signaling, ISDN Primary Rate Access, ISDN Basic Rate Access, etc.

Call Control:

This function provides the overall call control for a call. It takes care of signaling for individual bearer channels and decisions such as when to add or delete a channel from a call. The definition of this function is outside the scope of this International Standard; however, messages are defined between this function and the DEQ Control function for the purposes of defining the state machine.

Signaling Mechanism:

This function provides the actual signaling for connecting and disconnecting channels. Examples of signaling are ISDN D-channel Q.931 and A/B Robbed bit signaling using Dial Pulse or DTMF.

D.2 State Definitions

This subclause contains the definitions of the states used in the procedures. These states are defined for the DEQ Negotiation Control and DEQ Multiframe Control functions.

D.2.1. DEQ Negotiation Control

D.2.1.1. Calling Endpoint States

NULL (0): No calls active

CALL INIT (1): The Calling Endpoint has transmitted an INIT REQ message on the initial channel for initial parameter exchange and is waiting for an INIT ACK message containing the response from the Answering Endpoint.

CALL INIT - WAIT CID=1 (1a): The Calling Endpoint has completed parameter negotiation and DN transfer, has transmitted CID=1 and is waiting for CID=1 from Answering Endpoint

AWAIT DN (2): The Calling Endpoint has requested additional DNs and is waiting for the DNs from the Answering Endpoint.

ADDITIONAL CHANNELS (3): The Calling Endpoint has completed all parameter negotiation and DN transfer successfully and is setting up rest of the channels for the call.

ACTIVE (8): All the channels have been set up for the call (or downspeeding has occurred) and further negotiation is now possible. The DEQ Multiframe Control state machine controls the actual status of data transfer (i.e., local and remote synchronization) on the call.

ACTIVE - DELETE INIT (8a): The Calling Endpoint has requested a Delete Channel and is waiting for a response.

ACTIVE - ADD INIT (8b-1): The Calling Endpoint has requested an Add Channel and is waiting for a response.

ACTIVE - ADDITIONAL CHANNELS (8b-2): The Calling Endpoint has received a positive response to an Add Channel request and is setting up the additional channels.
ACTIVE - WAIT CID = 1 (8c): The Calling Endpoint has transmitted a CID = 1 to the Answering Endpoint signaling completion of the operation (i.e., Add or Delete Channel) and is waiting for CID=1.
ACTIVE - MODE 1 (8d): This is the ACTIVE state for Mode 1. In this state, no information messages are exchanged. The DEQ Negotiation Control passes messages (e.g., CC_RSYNCH_IND) between Call Control and DEQ Multiframe Control and accepts disconnect messages from Call Control.

ACTIVE - TRANSPARENT (8e): This is the ACTIVE state for the Transparent Mode.

DISCONNECT REQ (9): The Calling Endpoint has requested a DISCONNECT and is waiting for response.

D.2.1.2 Answering Endpoint States

NULL (0): No calls active.

CALL RECEIVED (4) (GLOBAL): The Answering Endpoint has received an incoming channel connect request and is determining what type of call it is (i.e., Waiting for INIT REQ or multi-frame) and to which call the channel belongs. It does this either by waiting for the inband information from the Calling Endpoint or by out-of-band mechanisms (e.g., ISDN D-channel information).

INIT RECEIVED (5): INIT REQ has been received and the Answering Endpoint has responded with an INIT ACK. The Answering Endpoint is waiting for CID = 1 or a request for a DN from the Calling Endpoint.

AWAIT ADDITIONAL CHANNELS (6): The Answering Endpoint has successfully completed the parameter negotiation and Directory Number (DN) exchange and is waiting to receive rest of channels for the call.

ACTIVE (8): All the channels have been set up for the call (or downspeeding has occurred) and further negotiation is now possible. The DEQ Multiframe Control state machine controls the actual status of data transfer (i.e., local and remote synchronization) on the call.

ACTIVE - DELETE INIT (8a): The Answering Endpoint has requested a Delete Channel and is waiting for a response.

ACTIVE - ADD INIT (8b-1): The Answering Endpoint has requested an Add Channel and is waiting for a response.

ACTIVE - WAIT ADDITIONAL CHANNELS (8b-2): The Answering Endpoint has received a positive response to an Add Channel request and is waiting to receive the added channels.

ACTIVE - WAIT CID = 1 (8c): The Answering Endpoint has completed the operation (i.e., Add Channel or Delete Channel) and is waiting to receive CID=1. After receipt of CID=1, further negotiation is possible.

ACTIVE - MODE 1 (8d): This is the ACTIVE state for Mode 1. In this state, no information messages are exchanged. The DEQ Negotiation Control passes messages (e.g., CC_RSYNCH_IND) between Call Control and DEQ Multiframe Control and accepts disconnect messages from Call Control.
ACTIVE - TRANSPARENT (8e): This is the ACTIVE state for the Transparent Mode.

DISCONNECT REQ (9): The Answering Endpoint has requested a DISCONNECT and is waiting for response.

D.2.1.3 ACTIVE - Loopback Substates

The following states are substates of the ACTIVE state. All actions and events defined for the ACTIVE state apply to these states.

ACTIVE (LOOP NULL): In this state, no loopbacks are active (i.e., transmitting and receiving RL REQ = 0 and RL IND = 0)..

REMOTE LOOP REQUEST: An endpoint has requested that the remote endpoint go into loopback state (i.e., transmitted RL REQ = 1) and is waiting for a response (i.e., waiting for RL IND = 1).

LOCAL LOOP ACTIVE: An endpoint is in a loopback state and is looping data back to the remote endpoint. The entire call is being looped back. The endpoint is transmitting RL IND = 1.

REMOTE LOOP ACTIVE: An endpoint has received an indication that the remote endpoint is in loopback mode (i.e., received RL IND = 1).

REMOTE LOOP OFF REQ: An endpoint has requested that the remote endpoint leave loopback mode and go back to ACTIVE mode (i.e., transmitted RL REQ = 0).

D.2.2. DEQ MULTIFRAME CONTROL States

NULL (0): The endpoint is waiting for a channel to be connected to the multiframe function. In this state, the DEQ Negotiation Control can be transmitting and receiving full bandwidth Information Channel messages in the initial channel (i.e., for parameter negotiation and DN exchange).

UNKNOWN SYNCH SEARCH (0a): In this state, the Answering Endpoint has connected the channel and is waiting to receive either multiframe synchronization or a full bandwidth information message signifying that this is an initial channel. DEQ Negotiation is running TAnull during this state.

NOTE - If the Answering Endpoint can determine the identity of the connected channel through other means (e.g., through ISDN D-channel signaling), it does not enter this state, but either stays in the NULL state (for the initial channel) or goes directly to the WAIT-LOCAL SYNCH state.

WAIT - LOCAL SYNCH (1): The endpoint is transmitting multiframe pattern. It is waiting for multiframe synchronization on all channels.

WAIT - REMOTE SYNCH (2): The endpoint has achieved multiframe synchronization on all channels. It is waiting for the remote endpoint to indicate multiframe synchronization on all channels (i.e., by transmitting A= 1 or RI = 1 in all channels). In this state, all channels are synchronized and the delays have been equalized locally. Receipt of user data is possible in this state.

MODE 1 HANDSHAKE (2a): This state is defined only for Mode 1. In this state, the endpoint is transmitting RI=1 and A = 0 in all channels and is waiting for either A=0 or loss of synchronization in all channels before removing framing and transmitting data.

ALL CHANNELS SYNCHED (3): Multiframe synchronization and delay equalization has been achieved in both directions in all channels. Receipt and transmission of user data is possible in this state.

MODE 1 ACTIVE (3a): This state is only relevant to Mode 1. In this state, framing has been removed and data is being transmitted. This state is not defined in the SDL Diagram since the DEQ Multiframe Control has no function.

D.3 Messages and Events

This subclause defines the messages that can be transmitted and received over the Information Channel, the internal messages passed between logical units within an endpoint and events that can happen over the interface. Equipment built to this specification **shall** conform to the actions described for events and messages that occur on the network interface. The definition of primitives between layers of the logical model are for description of the state machine and are not meant to constrain implementation to this logical model.

D.3.1 Information Channel Messages

In order to define the state machine more clearly, the Information Channel Messages are defined as separate messages as shown in Table A.1 below. Because the meaning of some messages are context specific (i.e., depending on which endpoint transmitted it and in what state the receiving endpoint is), some of the contents of different messages can be the same. The following terms are used in the messages and state machine:

CU:	Calling Endpoint
AU:	Answering Endpoint
CID:	Non-zero Channel Identifier assigned to the channel
Ma:	Mode requested by CU
Mb:	Mode requested by AU
Mx:	Mode negotiated for the call
Ba:	BCR requested by CU
Bb:	BCR requested by AU
Bx:	BCR agreed for call.
RMULTa:	(RMa) Rate Multiplier requested by CU
RMULTb:	(RMb) Rate Multiplier requested by AU
RMULTx:	(RMx) Rate Multiplier negotiated for call
RMULTy:	(RMy) New Rate Multiplier requested for Add or Delete Channel procedure.
SUBMULTa:	(SMa) Submultiplier requested by CU
SUBMULTb:	(SMb) Submultiplier requested by AU
SUBMULTx:	(SMx) Submultiplier negotiated for call
SUBMULTy:	(SMy) New Submultiplier requested for Add or Delete Channel procedure
GIDx:	Group Identifier assigned to the call.
Xa:	XFLAG indicating request for next Directory Number from the AU. Also sequence number assigned by AU for DNs.
Xb:	XFLAG indicating request for DN, Sequence number, etc. from CU
U:	User Information

For all references to CID=1, the assigned Channel Identifier is transmitted in each channel. For all references to CID=0, Channel Identifier 0 is transmitted in all channels. For Information Channel messages with CID = 0, the same information is transmitted in all channels for all fields.

The same values are always transmitted in all channels regardless of the value of the Channel Identifier for GID, MODE, RMULT, SUBMULT, BCR, RI, RL REQ and RL IND fields .

The Subaddress field and Manufacturer's ID Flag are only relevant to the INIT REQ and INIT ACK messages during parameter negotiation. They are ignored in all other messages.

A value of B 0's means that the field is filled with binary zeros and B 1's means the field is filled with binary ones.

Table D.1 - Information Channel Messages

MESSAGE	Orig	CID	MODE	BCR	R MULT	SUB MULT	GID	RI	RL REQ	RL IND	XFLAG	DIGIT
INIT REQ	CU	0	Ma	Ba	RMa	SMa	0	0	0	0	B 1's	B 1's/ MFG ID
INIT ACK	AU	0	Mb	Bb	RMb	SMb	GIDx	0	0	0	B 1's	B 1's/ MFG ID
DN REQ	CU	0	Mx	Bx	RMx	SMx	GIDx	0	0	0	Xa	B 1's
DN ACK	AU	0	Mx	Bx	RMx	SMx	GIDx	0	0	0	Xa	DN
CID=1	AU/CU	CID	Mx	Bx	RMx	SMx	GIDx	0/1	0/1	0/1	U	U
ADD CH	CU, AU	0	Mx	Bx	RMy	SMy	GIDx	0/1	0/1	0/1	Xa	B 1's
ADD CH ACK	CU, AU	0	Mx	Bx	RMy	SMy	GIDx	0/1	0/1	0/1	Xb	DN
ADD CH REJ	CU, AU	0	Mx	Bx	RMx	SMx	GIDx	0/1	0/1	0/1	Don't Care	B1's/ Cause
DISC	CU, AU	0	Mx	Bx	0	0	GIDx	0/1	0/1	0/1	Don't Care	B1's/ Cause
DISC REJ	CU, AU	0	Mx	Bx	RMx	SMx	GIDx	0/1	0/1	0/1	Don't Care	B1's/ Cause
DEL CH	CU, AU	0	Mx	Bx	RMy	SMy	GIDx	0/1	0/1	0/1	CI/B 0's	B1's/ Cause
DEL CH ACK	CU, AU	0	Mx	Bx	RMy	SMy	GIDx	0/1	0/1	0/1	CI/B 0's	B1's
DEL CH REJ	CU, AU	0	Mx	Bx	RMx	SMx	GIDx	0/1	0/1	0/1	Don't Care	B1's/ Cause
RL REQ	CU, AU	CID	Mx	Bx	RMx	SMx	GIDx	0/1	1	0/1	Don't Care	Don't Care
RL IND	CU, AU	CID	Mx	Bx	RMx	SMx	GIDx	0/1	0/1	1	Don't Care	Don't Care
RL OFF REQ	CU, AU	CID	Mx	Bx	RMx	SMx	GIDx	0/1	0	0/1	Don't Care	Don't Care
RL OFF IND	CU, AU	CID	Mx	Bx	RMx	SMx	GIDx	0/1	0/1	0	Don't Care	Don't Care
MESSAGE	Orig	CI	MODE	BCR	RM	SM	GID	RI	RL REQ	RL IND	XFLAG	DIGIT

D.3.2 DEQ Negotiation Control to Call Control Messages

This subclause defines messages exchanged between the DEQ Negotiation Control and Call Control functions. The definition of the Call Control function is outside the scope of this International Standard; however, the messages exchanged between them are defined here to provide the state machine definition. These messages are defined in the form of primitives. The primitives between the DEQ Negotiation Control and Call Control functions are of the form DQ_XXXX_yyy where DQ identifies the interface, XXXX identifies the message and yyy is of the form REQ, RESP, IND or CONF identifying the direction. The split in functionality defined in this subclause is not meant to restrict implementation as long as the implementation conforms to the state machine at the external interface.

DQ_CONN_REQ:

Call Control sends this message to DEQ Negotiation Control to notify DEQ Negotiation Control that a connection has been established. This primitive has the following parameters:

Direction: Indicates whether this is the Calling Endpoint or Answering Endpoint.

Initial: Indicates whether or not this is the initial connection of a new call or a subsequent connection of an existing call. If this is not known, then this information is not given (only for Answering Endpoint).

Call Identifier: Identifies the call. This is a combination of an internal identifier and the Group Identifier. If this is not known, then this information is not given (e.g., initial channel for Calling Endpoint and for Answering Endpoints).

Parameters: If this is the initial connection of a new call, this indicates the parameters to use for the call. This could also include a list of acceptable parameters for parameter negotiation. Examples of parameters are Mode, RMULT, SUBMULT, etc.

DQ_INIT_IND:

DEQ Negotiation Control sends this message to Call Control to indicate that Parameter Negotiation has been completed and Call Control can commence setting up additional channels if this is the Calling Endpoint or should wait for additional channels if this is the Answering Endpoint. It contains the following parameters:

Call Identifier: Identifies this call. This can include the Group Identifier and an internal identifier.

DNs: Indicates Directory Numbers received.

Parameters: Indicates negotiated parameters.

DQ_DISC_REQ:

Call Control sends this primitive to the DEQ Negotiation Control to indicate that the call is to be disconnected gracefully. It includes the following information:

Call Identifier: Identifies this call.

Cause.

DQ_DISC_CONF:

DEQ Negotiation Control sends this primitive to Call Control to indicate that graceful disconnect has completed. It includes the following information:

Call Identifier: Identifies this call.

DQ_DISC_IND:

DEQ Negotiation Control sends this primitive to Call Control to indicate that it has completed a graceful disconnect initiated from the remote endpoint. It includes the following information:

Call Identifier: Identifies this call.

Cause.

DQ_DEL_CH_REQ:

Call Control sends this primitive to DEQ Negotiation Control to request that DEQ Negotiation Control delete one or more channels from a call (e.g., due to receipt of a disconnect indication from the network). The following parameters are defined:

Call Identifier: Identifies the call.

Channels to be deleted: Identifies the channels to be deleted or the number of channels to be deleted.

Cause.

DQ_DEL_CH_CONF:

DEQ Negotiation Control sends this primitive to Call Control to confirm that the channel(s) has (have) been deleted.

Call Identifier: Identifies the call.

Channels deleted: Identifies the channels deleted .

DQ_DEL_CH_IND:

DEQ Negotiation Control sends this primitive to Call Control to indicate that a Channel Deletion procedure has been initiated from the remote side or that a Channel has failed.

Call Identifier: Identifies the call.

Channels to be deleted: Identifies the channels to be deleted.

Cause

DQ_DEL_CH_FAIL_IND:

DEQ Negotiation Control sends this primitive to Call Control to indicate that the requested Channel Deletion procedure has failed.

Call Identifier: Identifies the call.

Channels to be deleted: Identifies the channels to be deleted.

Cause

DQ_ADD_CH_REQ:

Call Control sends this primitive to DEQ Negotiation Control to request that it initiate procedures to add one or more channels.

Call Identifier: Identifies the call.

Number of Channels to be added/Desired Bandwidth: This identifies the desired bandwidth as a result of the add operation.

DQ_ADD_CH_IND:

DEQ Negotiation Control sends this primitive to Call Control to indicate that a Channel Add procedure has been initiated from the remote side. For the Calling Endpoint, Call Control should set up the additional channel. For the Answering Endpoint, it should wait for an additional channel.

Call Identifier: Identifies the call.

Number of Channels to be added/Desired Bandwidth

DQ_ADD_CH_CONF:

DEQ Negotiation Control sends this primitive to Call Control to confirm that the channel addition negotiation initiated by the Call Control is complete. Call Control should set up the additional channel..

Call Identifier: Identifies the call.

Number of Channels added/Desired Bandwidth

DQ_ADD_CH_FAIL_IND:

DEQ Negotiation Control sends this primitive to Call Control to indicate that the requested Channel Add procedure has failed.

Call Identifier: Identifies the call.

Number of Channels to be added/Desired Bandwidth

DQ_CID_FAIL_IND:

DEQ Negotiation Control sends this primitive to Call Control to indicate that timer TCID has expired meaning that the call attempt has failed.

Call Identifier: Identifies the call.

DQ_LLOS_IND:

DEQ Negotiation Control sends this primitive to Call Control to indicate that a channel has lost local synchronization. This is a pass-through of the CC_LLOS_IND.

Call Identifier: Identifies the call.

Channel Identifier: Identifies the channel.

Cause

NOTE 1 - Actions taken by equipment written to this specification for loss of multiframe synchronization should be consistent with the actions defined in these procedures for receipt of the DQ_LLOS_IND.

DQ_RLOS_IND:

DEQ Negotiation Control sends this primitive to Call Control to indicate that a channel or the call has lost remote synchronization (i.e., received A bit = 0 or RI = 0). This is a pass-through of the CC_RLOS_IND.

Call Identifier: Identifies the call.

Channel Identifier: Identifies the channel.

Cause

NOTE 2 - Actions taken by equipment written to this specification for loss of multiframe synchronization should be consistent with the actions defined in these procedures for receipt of the DQ_RLOS_IND.

DQ_ABORT_REQ:

Call Control sends this primitive to DEQ Negotiation Control to request that it disconnect the call without going through graceful disconnect procedures.

Call Identifier: Identifies the call.

DQ_ABORT_CONF:

DEQ Negotiation Control sends this primitive to Call Control to confirm that it has disconnected the call without going through graceful disconnect procedures (i.e., returned all channels and state machines associated with the call to the null state).

Call Identifier: Identifies the call.

DQ_LL_IND:

DEQ Negotiation Control sends this primitive to Call Control to signal receipt of a request for the local endpoint to loop the data back to remote endpoint.

Call Identifier: Identifies the call.

DQ_LL_OFF_IND:

DEQ Negotiation Control sends this primitive to Call Control to signal receipt of a request for the endpoint to leave loopback mode.

Call Identifier: Identifies the call.

DQ_LL_OFF_REQ:

Call Control sends this primitive to DEQ Negotiation Control to request that it send a RL OFF IND message to the remote endpoint indicating that the local endpoint has left loopback mode.

Call Identifier: Identifies the call.

DQ_LL_RESP:

Call Control sends this primitive to DEQ Negotiation Control to request that it send a RL IND message to the remote endpoint indicating that the local endpoint is in loopback mode.

Call Identifier: Identifies the call.

DQ_RL_REQ:

Call Control sends this primitive to DEQ Negotiation Control to request that it send a RL REQ message to the remote endpoint to place the remote endpoint into loopback mode.

Call Identifier: Identifies the call.

DQ_RL_IND:

DEQ Negotiation Control sends this primitive to Call Control to signal receipt of an indication that the remote endpoint has gone into loopback mode.

Call Identifier: Identifies the call.

DQ_RL_OFF_IND:

DEQ Negotiation Control sends this primitive to Call Control to signal receipt of an indication that the remote endpoint has left loopback mode.

Call Identifier: Identifies the call.

D.3.3 DEQ Negotiation Control to DEQ Multiframe Control Primitives

The following primitives are used between the DEQ Multiframe Control and the DEQ Negotiation Control:

CC_ADD_REQ

DEQ Negotiation Control sends this primitive to DEQ Multiframe Control to request that it add the indicated channel or change the data distribution (i.e., if new RMULT/SUBMULT is required that does not change the number of channels. This is only possible in Mode 3). If this primitive is sent with N, Call Identifier, and Channel ID set to null (only by the Answering Endpoint), then the Multiframe Control must look for both full bandwidth Information Channel messages (i.e., Initial Channel) and Multiframe synchronization to determine the call to which the channel belongs. Otherwise, it contains the values of N (RMULT/SUBMULT), Call Identifier and Channel ID (optional in case of data redistribution) to add.

N: Identifies the number of channels in the call (or RMULT/SUBMULT)

Call Identifier: Identifies the call

Channel ID: Identifies the channel connected.

CC_LSYNCH_IND:

DEQ Multiframe Control sends this primitive to DEQ Negotiation Control to confirm that the call has achieved local multiframe synchronization and delay equalization.

Call Identifier: Identifies the call.

NOTE 1 - Actions taken by equipment written to this specification for achieving channel synchronization **shall** be consistent with the actions defined in these procedures for receipt of the CC_LSYNCH_IND.

CC_RSYNCH_IND:

DEQ Multiframe Control sends this primitive to DEQ Negotiation Control to confirm that the remote endpoint has achieved multiframe synchronization and delay equalization on all channels. This can be determined by receipt of A=1 or RI=1 in all channels.

Call Identifier: Identifies the call.

NOTE 2 - Actions taken by equipment written to this specification for achieving channel synchronization **shall** be consistent with the actions defined in these procedures for receipt of the CC_RSYNCH_IND.

CC_RSYNCH_FAIL_IND:

DEQ Multiframe Control sends this primitive to DEQ Negotiation Control to indicate that the remote endpoint has failed to reach multiframe synchronization and delay equalization in all channels before the expiration of Txdeq.

Call Identifier: Identifies the call.

Cause: Identifies the cause for the failure

NOTE 3 - Actions taken by equipment written to this specification for achieving channel synchronization **shall** be consistent with the actions defined in these procedures for receipt of the CC_RSYNCH_FAIL_IND.

CC_INFO_REQ/IND:

DEQ Control sends this primitive to Channel Control to request that it send an Information Message in the Information channel. This message shows up in the state machine description as Information Channel messages. This message can include an indication of whether the message was a full bandwidth message on the initial channel or an Information Channel message received in the multiframe.

Call ID

Information Message Type: Full Bandwidth (FB) or Multiframe (MF)

Information Message: The Information Channel message received.

Channel ID: This is not needed for standard Information Channel messages since these messages are repeated on all channels.

CC_FAIL_IND:

DEQ Multiframe Control sends this primitive to DEQ Negotiation Control to indicate that a channel has failed.

Call ID

Channel ID

Cause

NOTE 4 - Actions taken by equipment written to this specification for channel failure (e.g., extended loss of synchronization, excessive errors, etc.) **shall** be consistent with the actions defined in these procedures for receipt of the CC_FAIL_IND.

CC_LLOS_IND:

DEQ Multiframe Control sends this primitive to DEQ Negotiation Control to indicate that a channel has lost local synchronization. This also implies that DEQ Multiframe Control has lost delay equalization and data is being lost.

Call ID

Channel ID

Cause

NOTE 5 - Actions taken by equipment written to this specification for loss of multiframe synchronization **shall** be consistent with the actions defined in these procedures for receipt of the CC_LLOS_IND.

CC_RLOS_IND:

DEQ Multiframe Control sends this primitive to DEQ Negotiation Control to indicate that the remote endpoint has lost synchronization on a channel . This could be sent if the endpoint received A = 0 in one or more channels or RI=0 in all channels.

Call ID

Channel ID: This can be one Channel ID, a list of Channel IDs or a value of ALL for all channels in the call.

Cause

NOTE 6 - Actions taken by equipment written to this specification for loss of multiframe synchronization **shall** be consistent with the actions defined in these procedures for receipt of the CC_RLOS_IND.

CC_DEL_REQ

DEQ Negotiation Control sends this primitive to DEQ Multiframe Control to request that it disconnect the channel. This primitive can also be used to notify the DEQ Multiframe Control of a new value of N to use without deleting a channel. For example, this would occur if the DEQ Negotiation Control's Txadd01 timer timed out and downspeed occurred. In addition, for Mode 3, this primitive can be used to tell the DEQ Multiframe Control of a new RMULT/SUBMULT to use if data is to be redistributed without deleting a channel.

Call ID

New RMULT/SUBMULT

Channel ID: This can be one Channel ID, a list of Channel IDs or a value of ALL for all channels in the call. In the case of ALL channels, the call is disconnected.

D.4 Timers

This subclause defines the timers to be used for the call.

Table D.2 - Definition of Timers

TIMER	Default	Range	Start	Stop	Action on Expiry
Txinit NOTES 1, 2	5 sec	500 msec - 10 sec	Endpoint starts negotiation procedures for initial channel	Initial channel Negotiation procedures complete (i.e., parameter negotiation and DN request)	Disconnect call and notify management
Txfa NOTES 1, 2	1 sec	32 msec - 10 sec.	Endpoint starts searching for multiframe structure	Endpoint detects Frame Synchronization and receives indication that remote endpoint has synchronized.	Remove Channel and notify Mgt.
Txadd01 NOTE 2	NOTE 3	2 - 180 sec	Finish DN transfer procedures.	Final channel in initial call is connected	Notify Mgt. Downspeed or Disconnect
Txdeq NOTE 2	7 sec.	.5 - 10 sec.	Start Delay Equalizing call	Delay is equalized. Transmit and Receive RI = 1/ A = 1.	Notify Mgt
Txadd02 NOTE 2	NOTE 2	2 - 180 sec.	Finish negotiating for adding a channel	Channel connected	Notify Mgt. Downspeed or disconnect.
Txadd Txdel NOTE 2	5 sec.	1 - 10 sec.	Send CH ADD or CH DEL message	Receive acknowledgment of CH ADD or CH DEL	Notify Mgt. State Dependent.
Txdisc NOTE 2	5 sec	1 sec - 10 sec.	Endpoint requests disconnection of call	Endpoint completes disconnection of call..	Disconnect call via call control and notify mgt.
TAnull	2 sec (NOTE 4)	50 msec - 10 sec.	Answering Endpoint receives incoming call.	Answering Endpoint receives first Information Message or detects framing on the channel	Default to Transparent Mode (NOTE 5)
TCnull	2 sec	50 msec - 10 sec	Calling Endpoint connects call	Calling Endpoint detects Information Channel Framing	Default to Transparent Mode
TCID NOTE 6	5 sec	3 - 5 sec	Transmit CID = 1 or wait for CID=1	Receive CID = 1	Notify mgt.

NOTES

1 - the Txinit and/or Txfa timer values may need to be increased when utilising circuits with long round-trip delay or when needing to cater for long telephone numbers

2 - in the timer table, x can be A or C depending on the side implementing the timer.

3 - depends on number of channels and on the delay between consecutive additional channel setup attempts.

4 - for interworking with endpoints implementing ITU-T H-series recommendations (i.e., H.221), it is recommended that a maximum value of 2 seconds be used (except as noted below). This is to allow fallback to H.221 before the H.221 initialization timer (10 seconds) expires. For interworking with endpoints that use Echo Cancelling Disabling Tone or for international calling, it is recommended that TAnull be increased to 5 seconds. Note that this value will still allow for fallback to H.221 for video applications.

5 - for non-video applications and for applications using more than 6 channels, the Transparent Mode can be disabled. In this case, the endpoint disconnects the channel at the expiration of TAnull. In addition, in this case, the maximum value of TAnull can be set to 60 seconds.

6 - TCID applies only during the Active State (not during call setup).

D.5 State Transition Diagrams

This subclause contains State Transition Diagrams for the DEQ Negotiation Control and the Multiframe Control function. These diagrams provide an overview of the state transitions.

D.5.1 DEQ Negotiation Control

D.5.1.1 Calling Endpoint

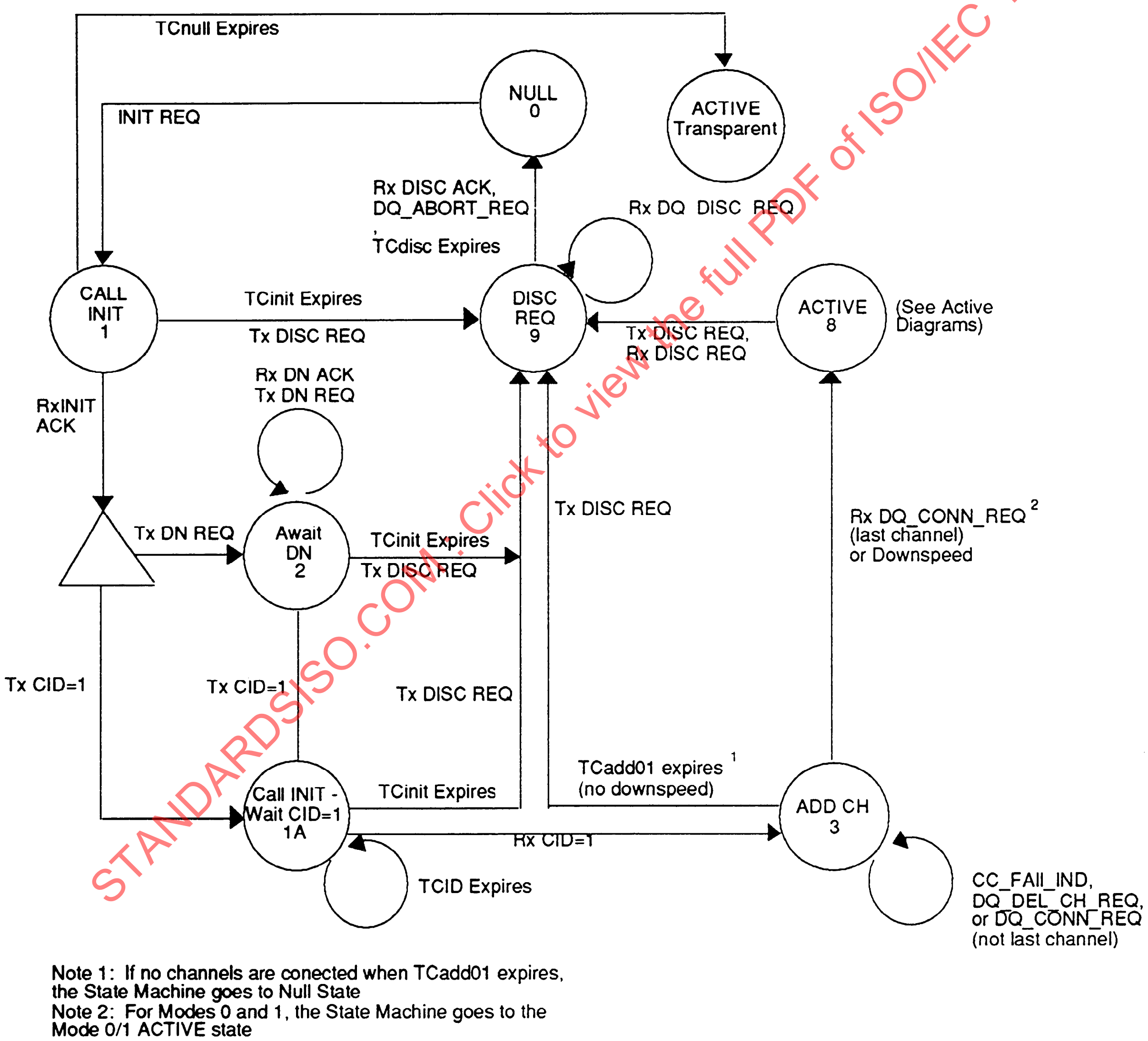


Figure D.2 - Calling Endpoint Setup and Clear

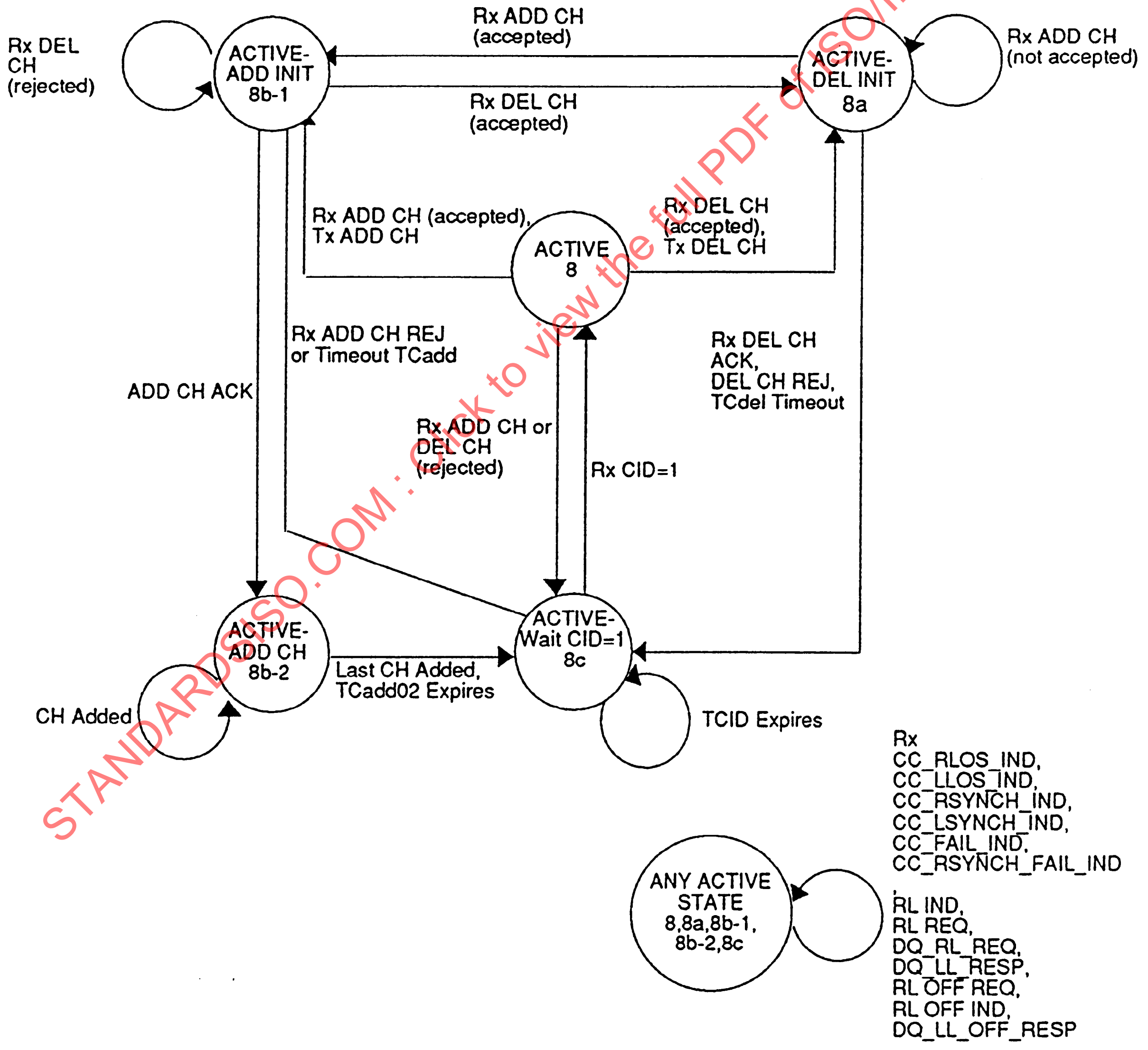


Figure D.3 - Calling Endpoint Channel Add and Delete

D.5.1.2 Answering Endpoint

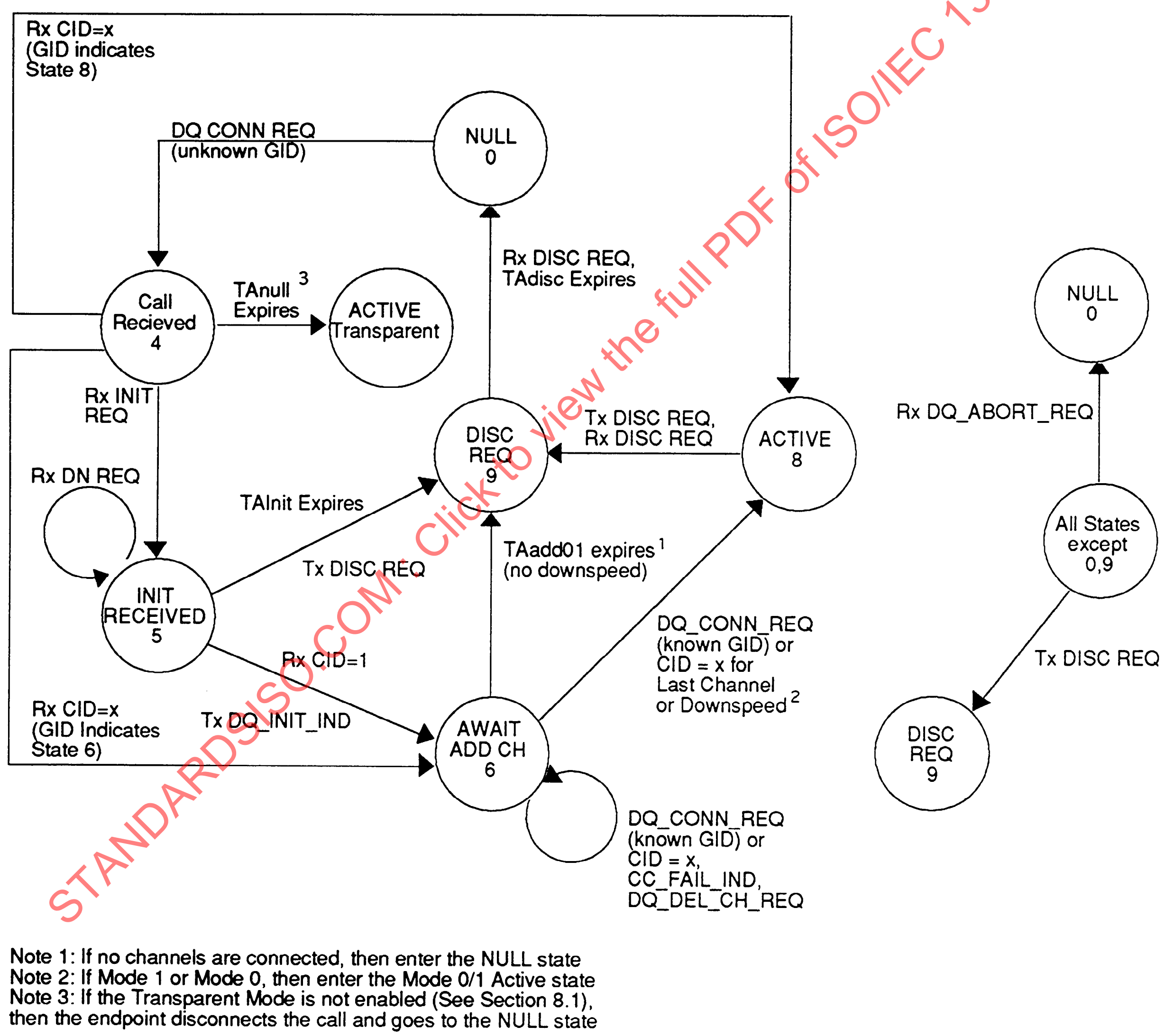


Figure D.4 - Answering Endpoint Setup and Clear

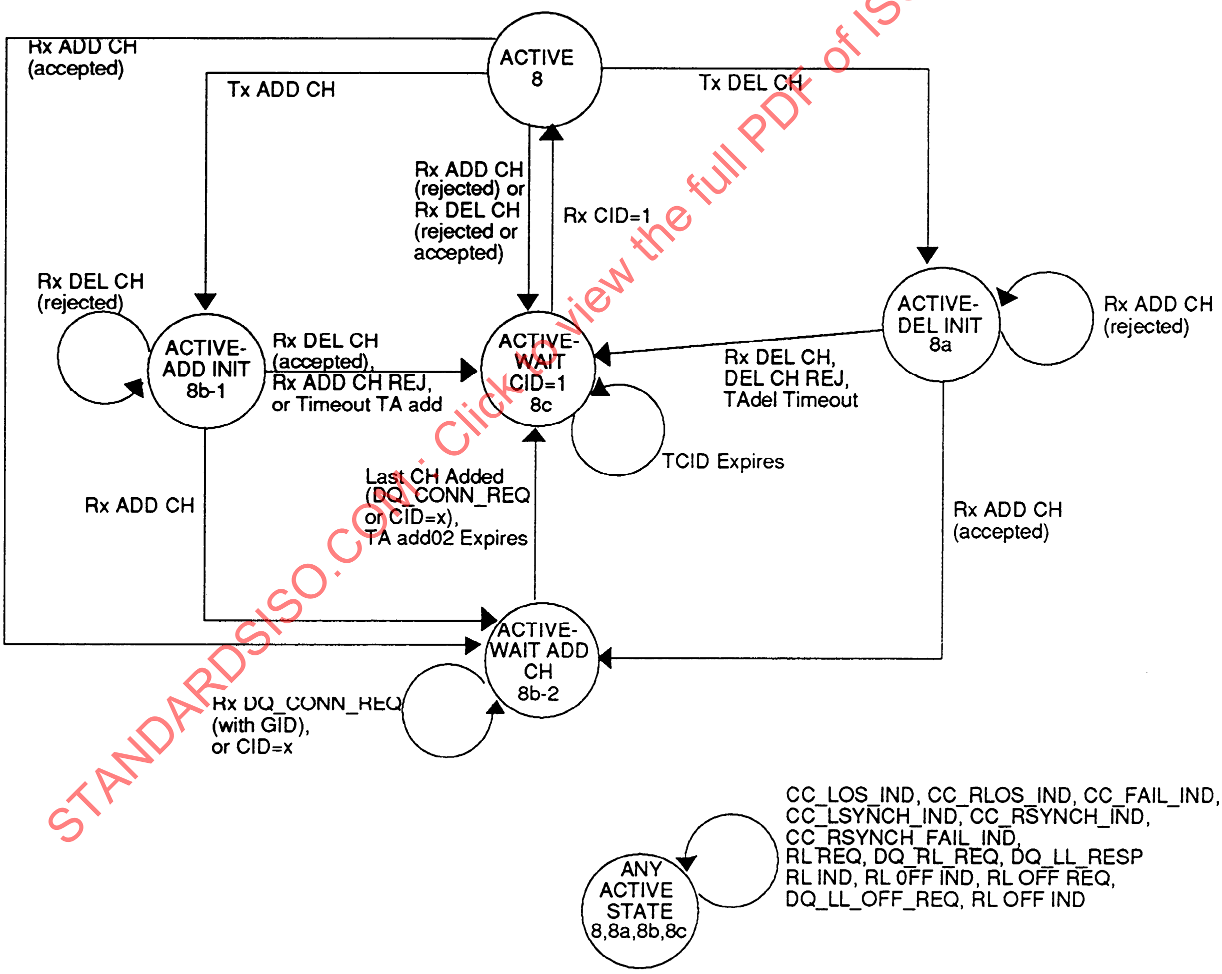


Figure D.5 - Answering Endpoint Channel Add and Delete

D.5.2 DEQ Multiframe Control

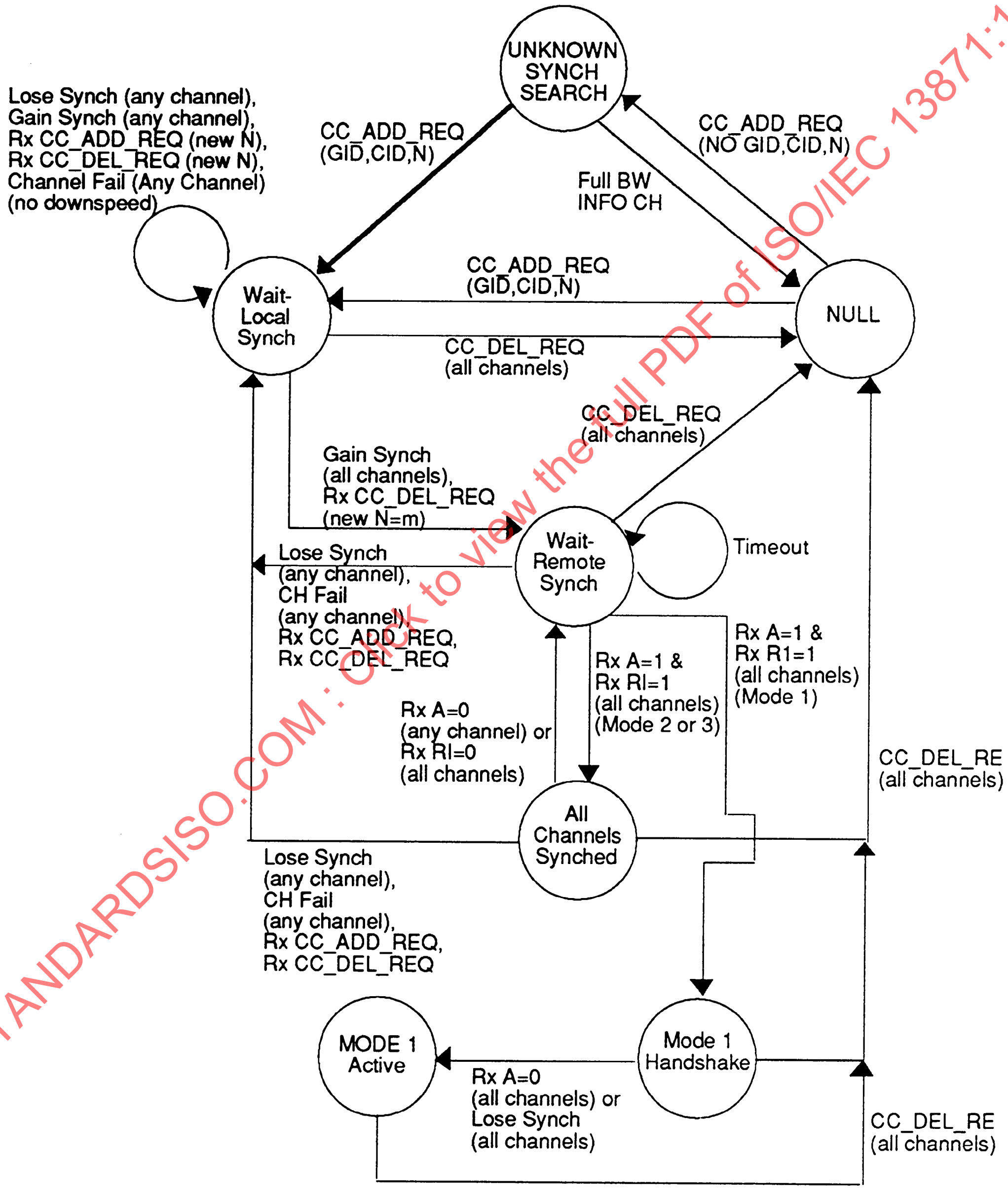
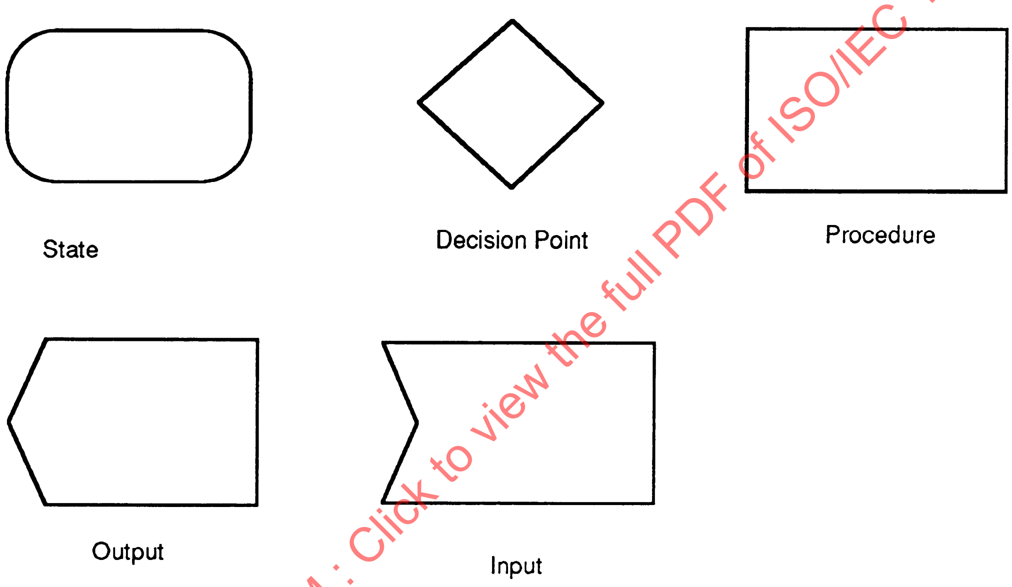


Figure D.6 - DEQ Multiframe Control

D.6 SDL Diagrams

The following figure defines the symbols used in the SDL Diagrams.



- Outputs and Inputs with the prefix DQ represent primitives between DEQ Negotiation Control and Call Control
- Outputs and Inputs with the prefix CC represent primitives between DEQ Negotiation Control and DEQ Multiframe Control
- All other Outputs and Inputs are either peer-to-peer messages or timer expirations

Figure D.7 - SDL Diagrams

The following counters and variables are used in the SDL Diagrams:

N: The number of channels requested. Also, N-1 gives the number of DNs to be requested. This is determined from the RMULT/SUBMULT negotiated during parameter exchange.

m: The number of channels connected and frame aligned

d: Determines the number of DNs received.

C: The number of channels connected.

D.6.1 DEQ Negotiation Control

This subclause contains the Calling Endpoint SDL Diagrams for DEQ Negotiation Control.

D.6.1.1 Calling Endpoint

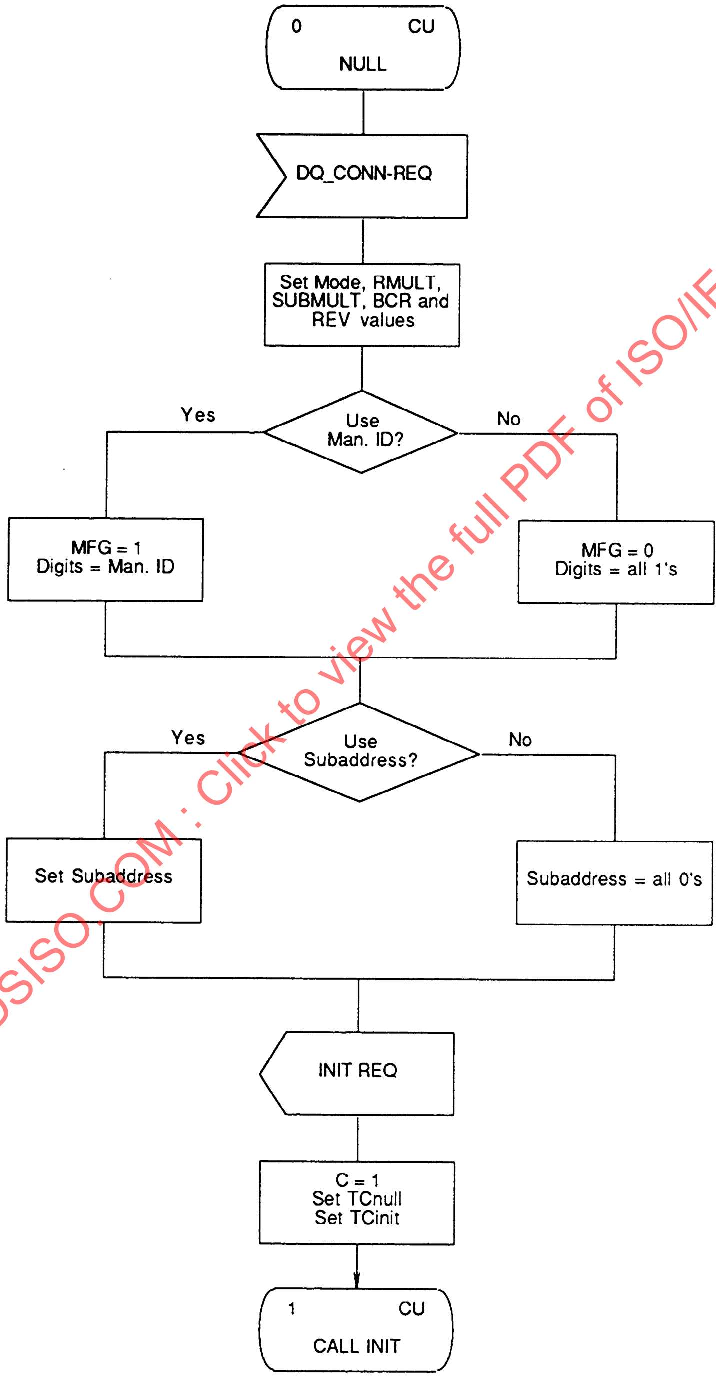


Figure D.8 - Calling Endpoint NULL State

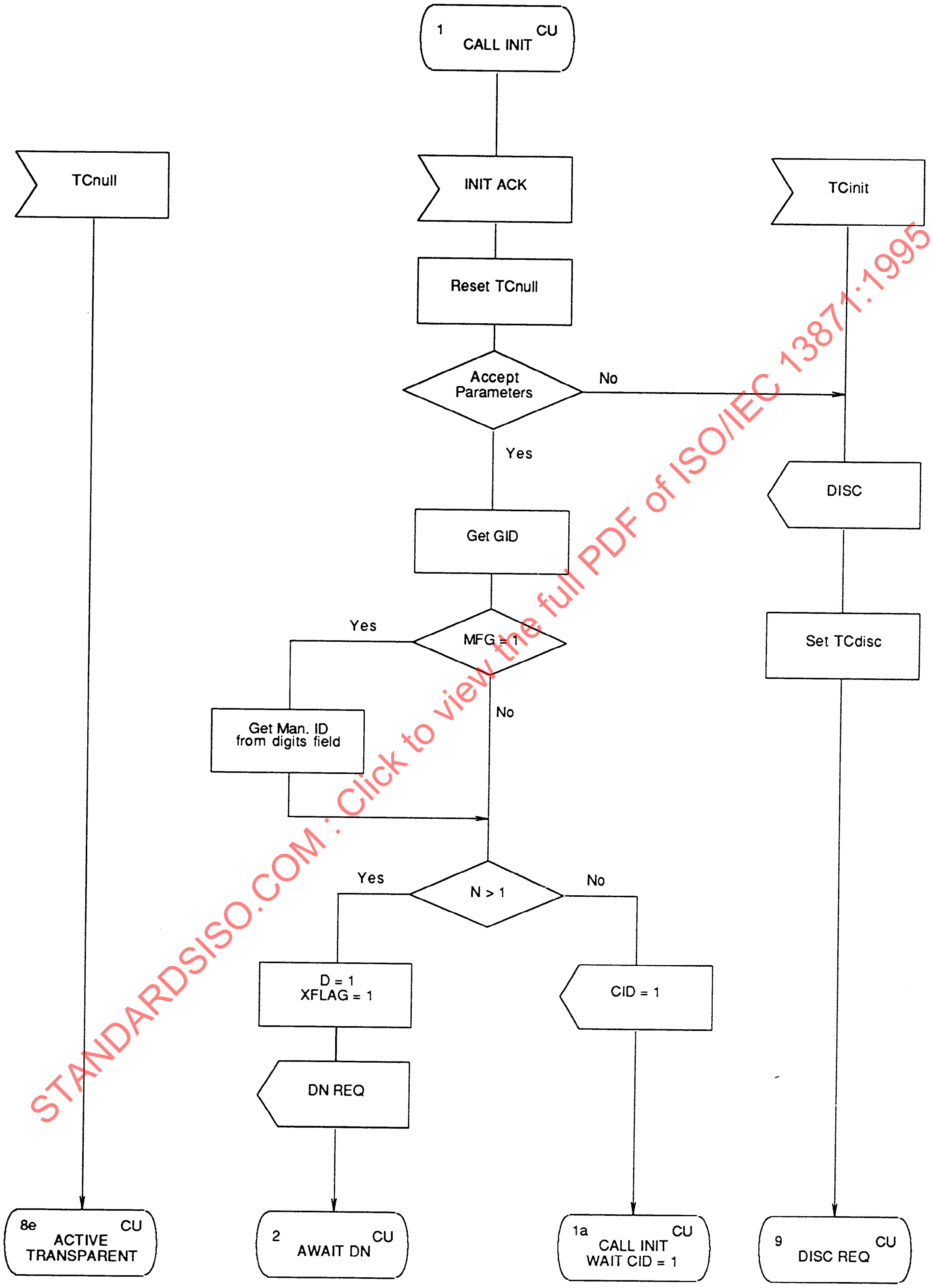


Figure D.9 - Calling Endpoint CALL INIT State

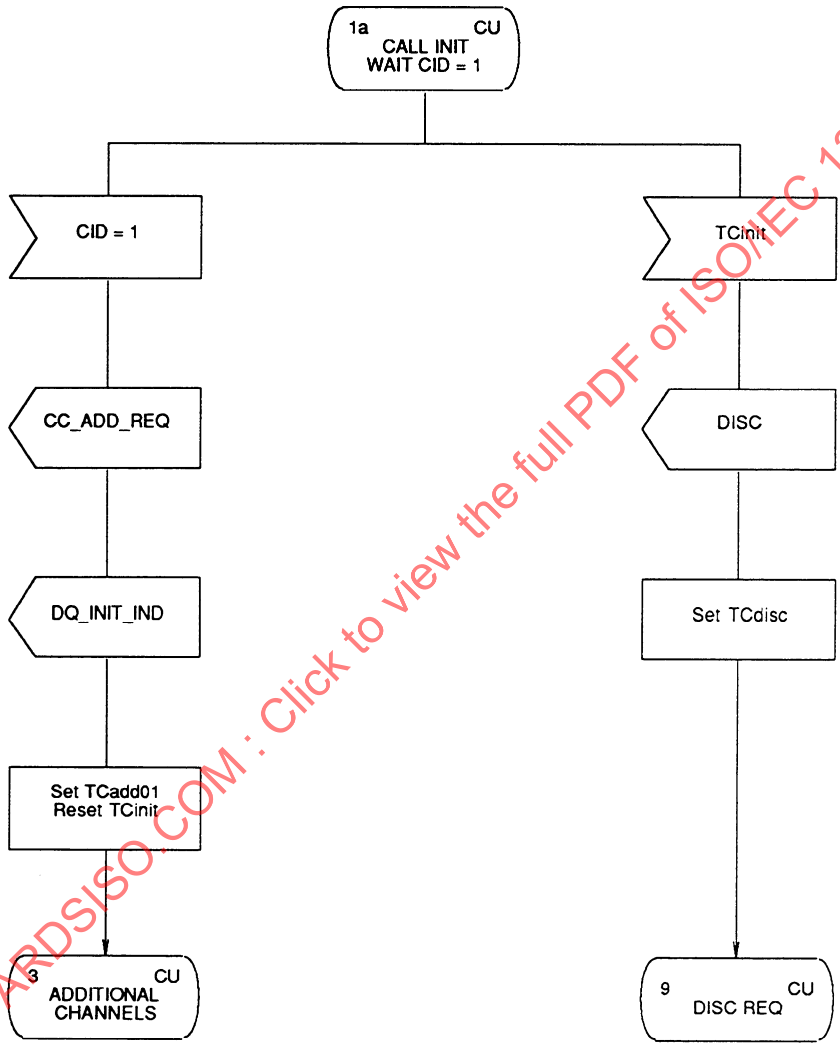


Figure D.10 - Calling Endpoint CALL INIT WAIT CID = 1 State

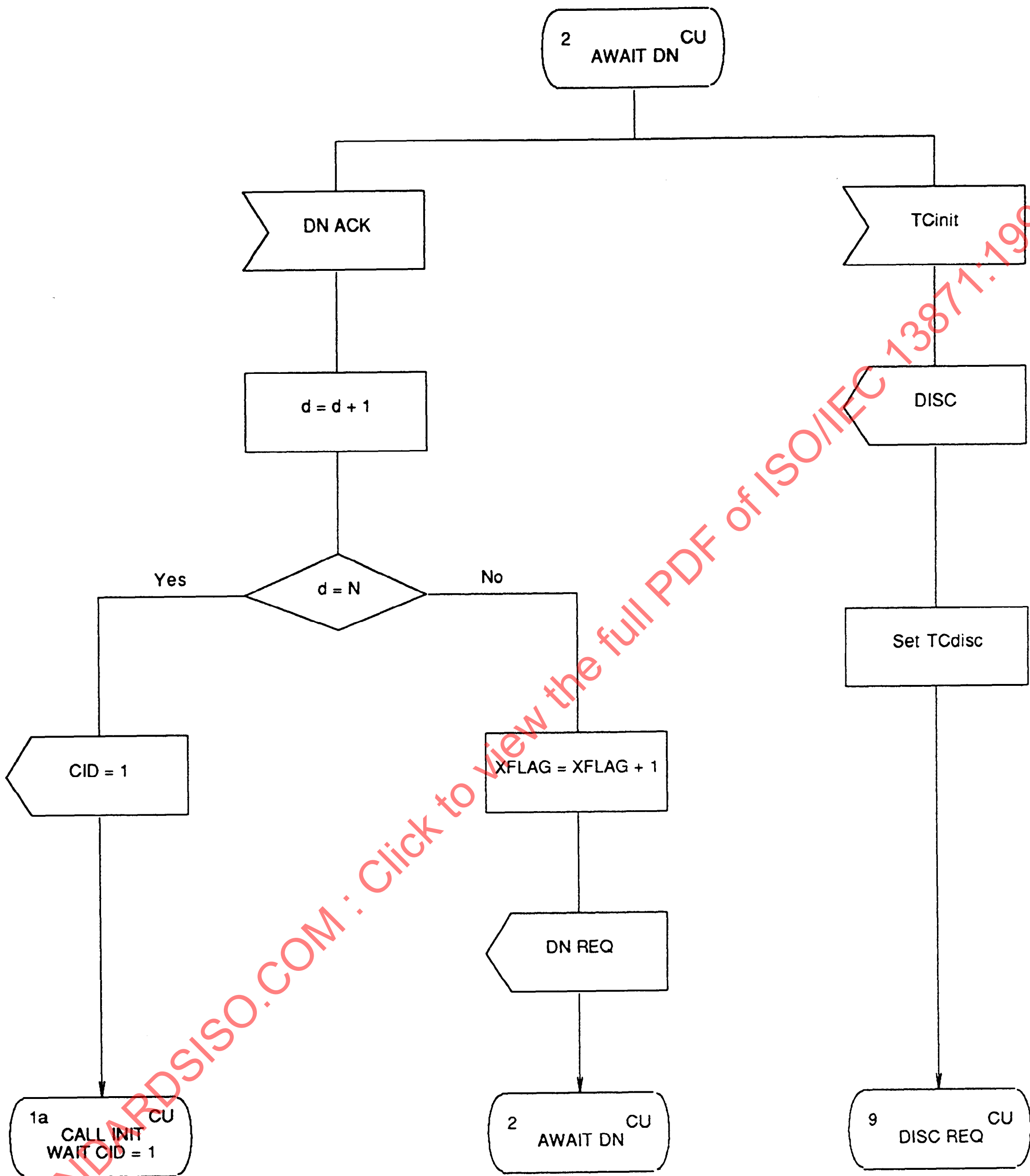


Figure D.11 - Calling Endpoint AWAIT DN State

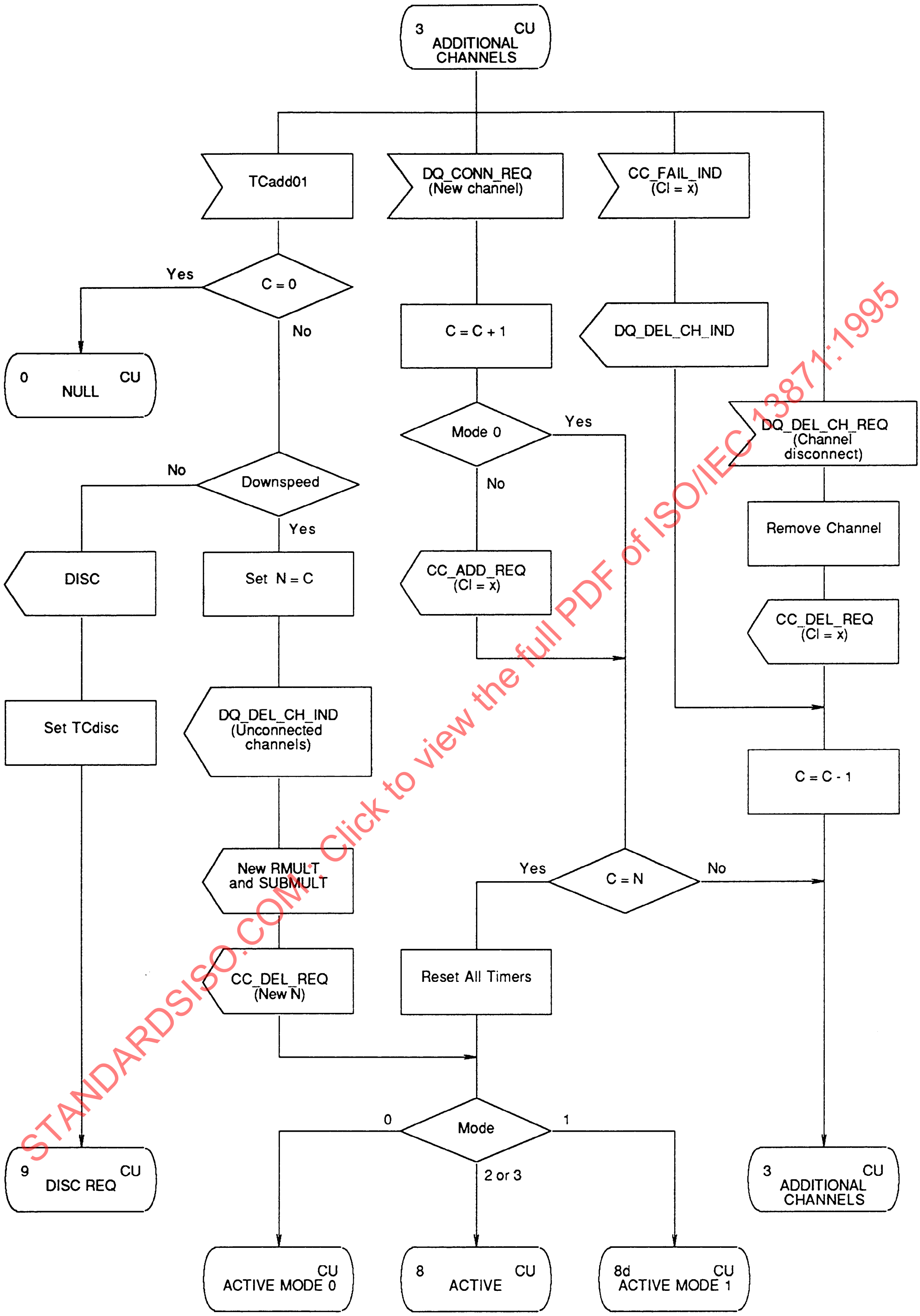


Figure D.12 - Calling Endpoint ADDITIONAL CHANNELS State

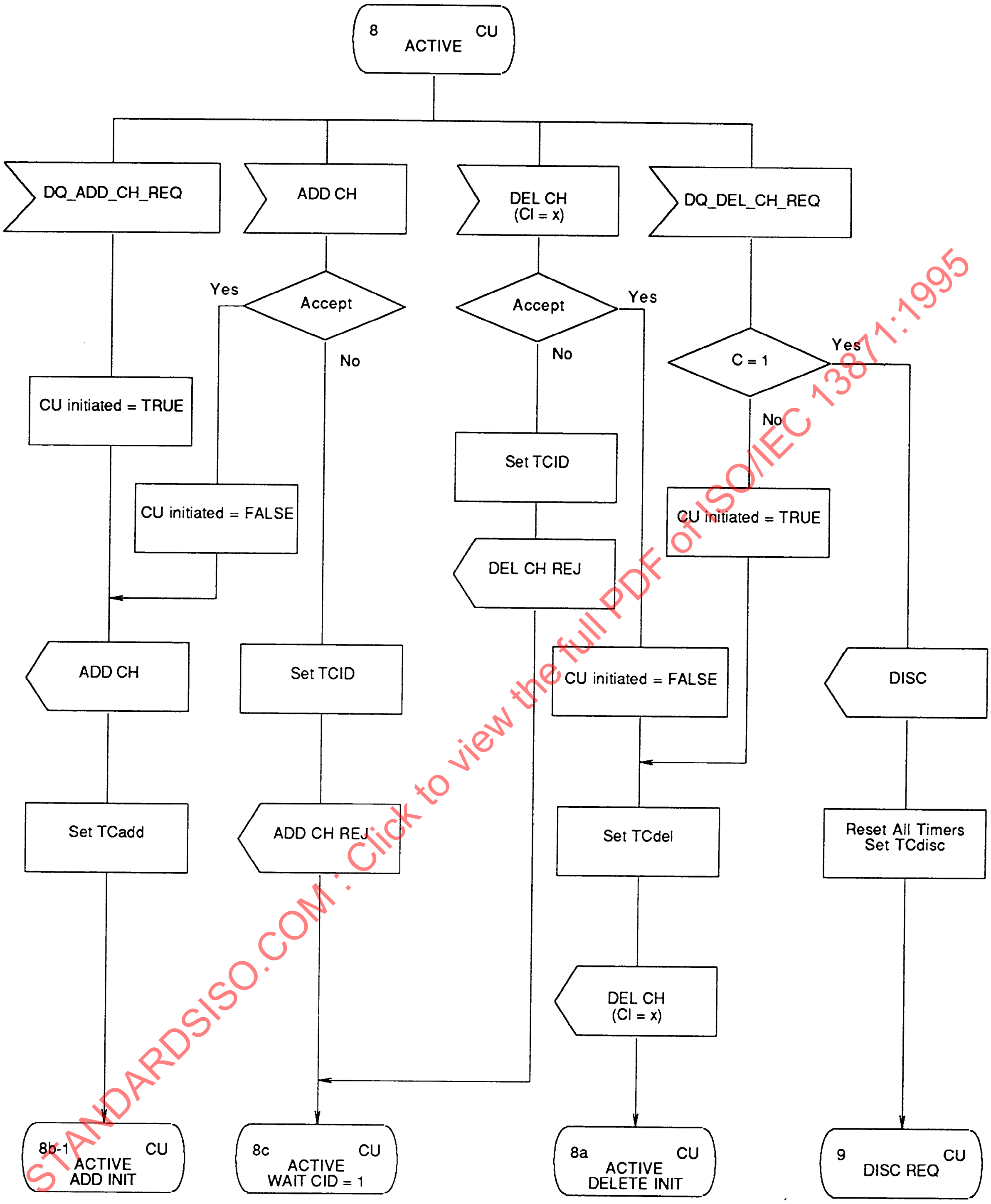


Figure D.13 - Calling Endpoint ACTIVE State

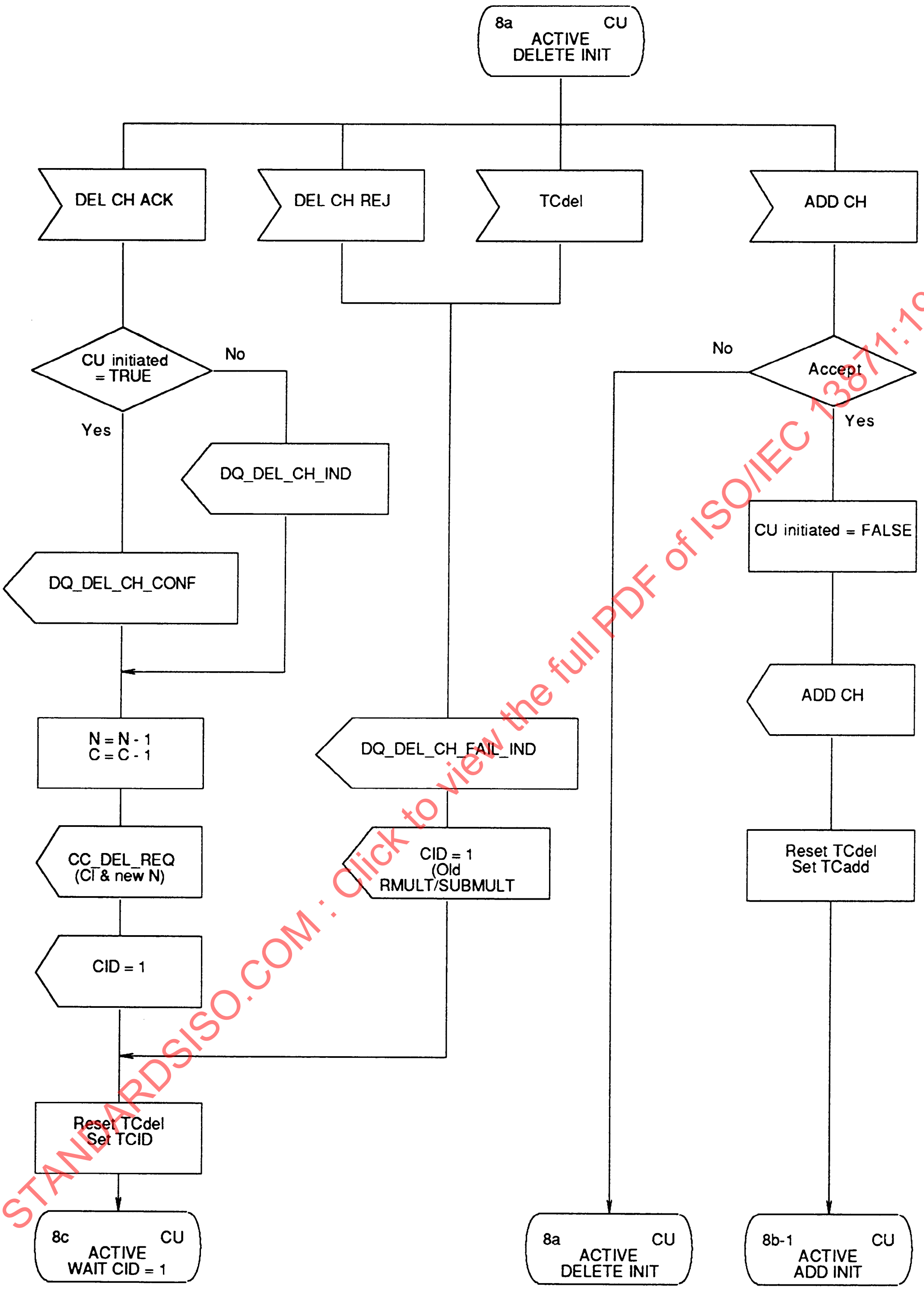


Figure D.14 - Calling Endpoint ACTIVE DELETE INIT State

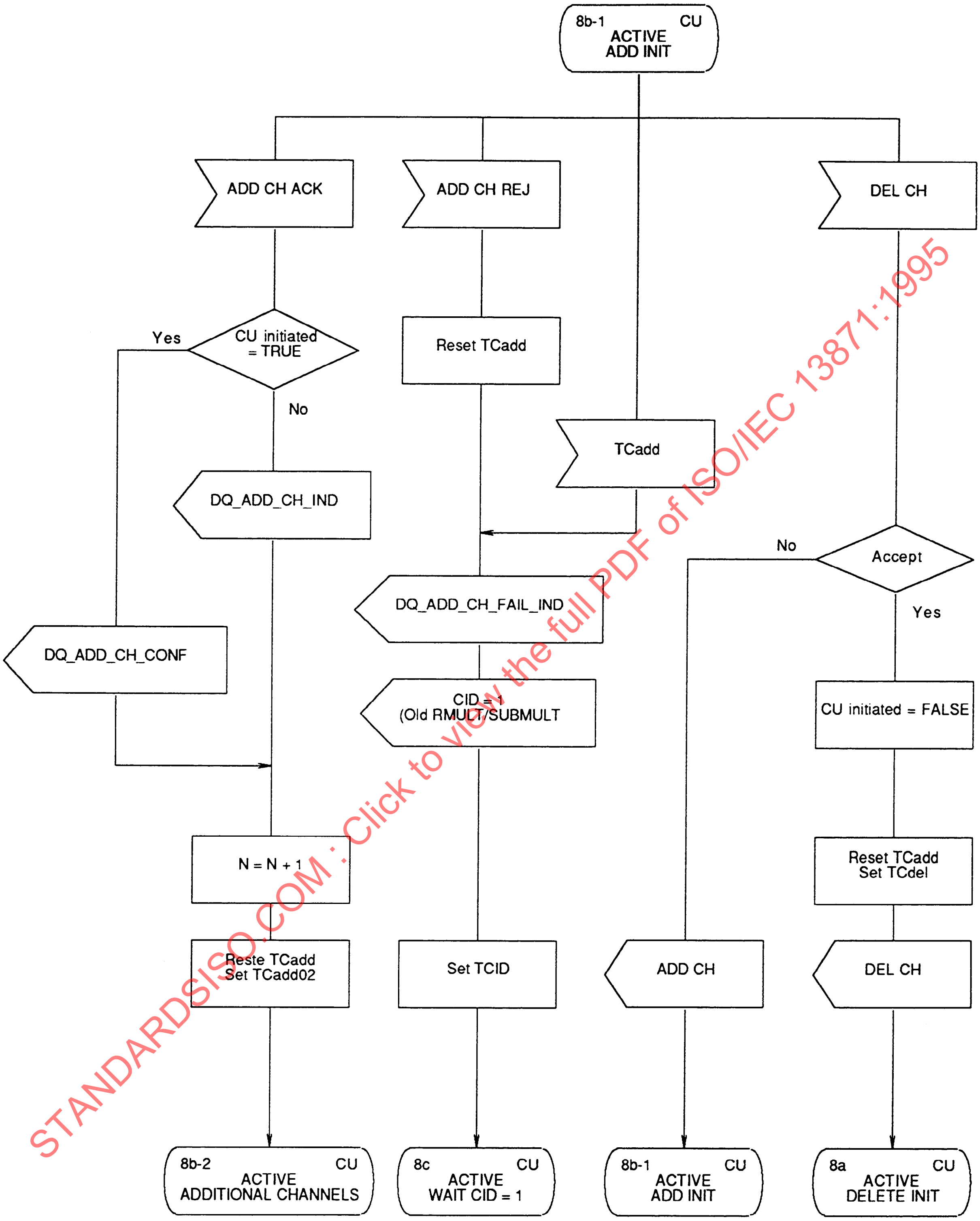


Figure D.15 - Calling Endpoint ACTIVE ADD INIT State

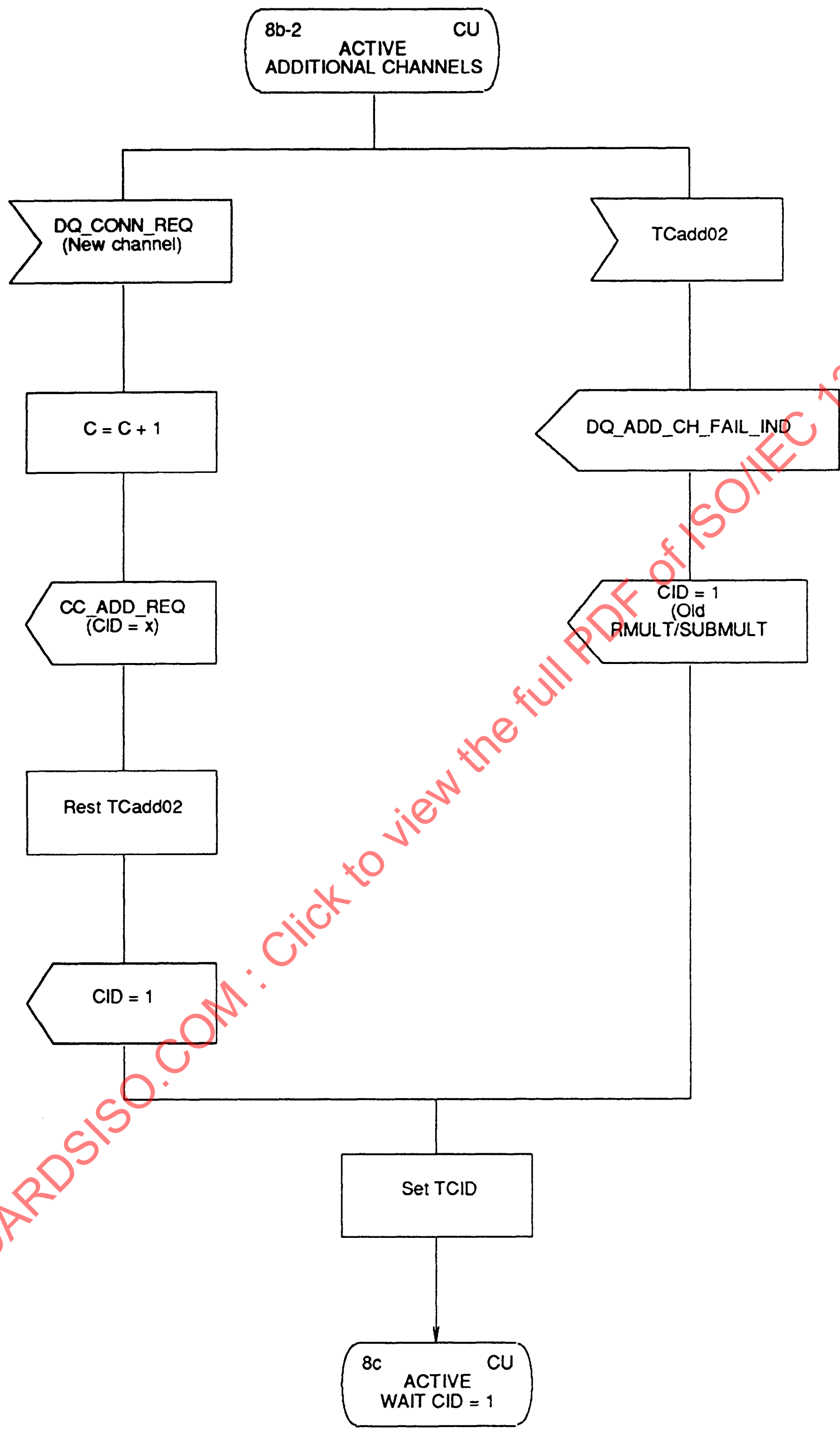


Figure D.16 - Calling Endpoint ACTIVE ADDITIONAL CHANNELS State

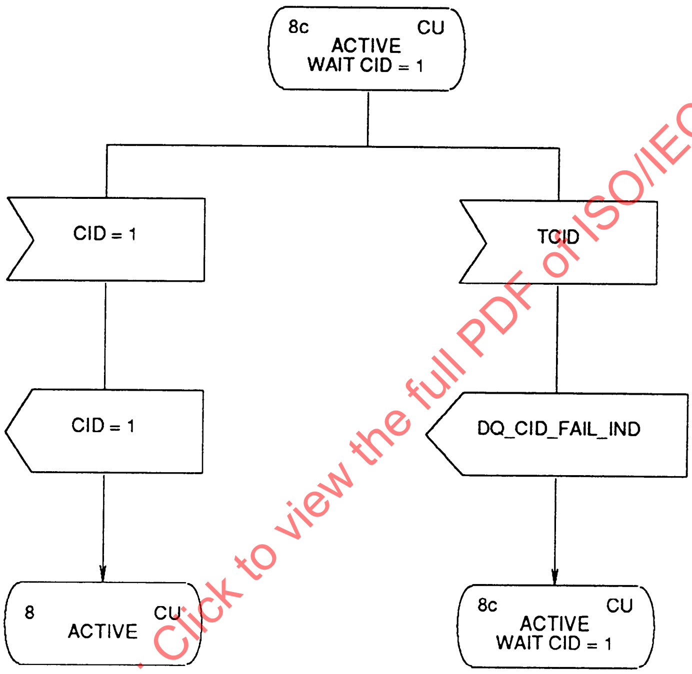


Figure D.17 - Calling Endpoint ACTIVE WAIT CID = 1 State

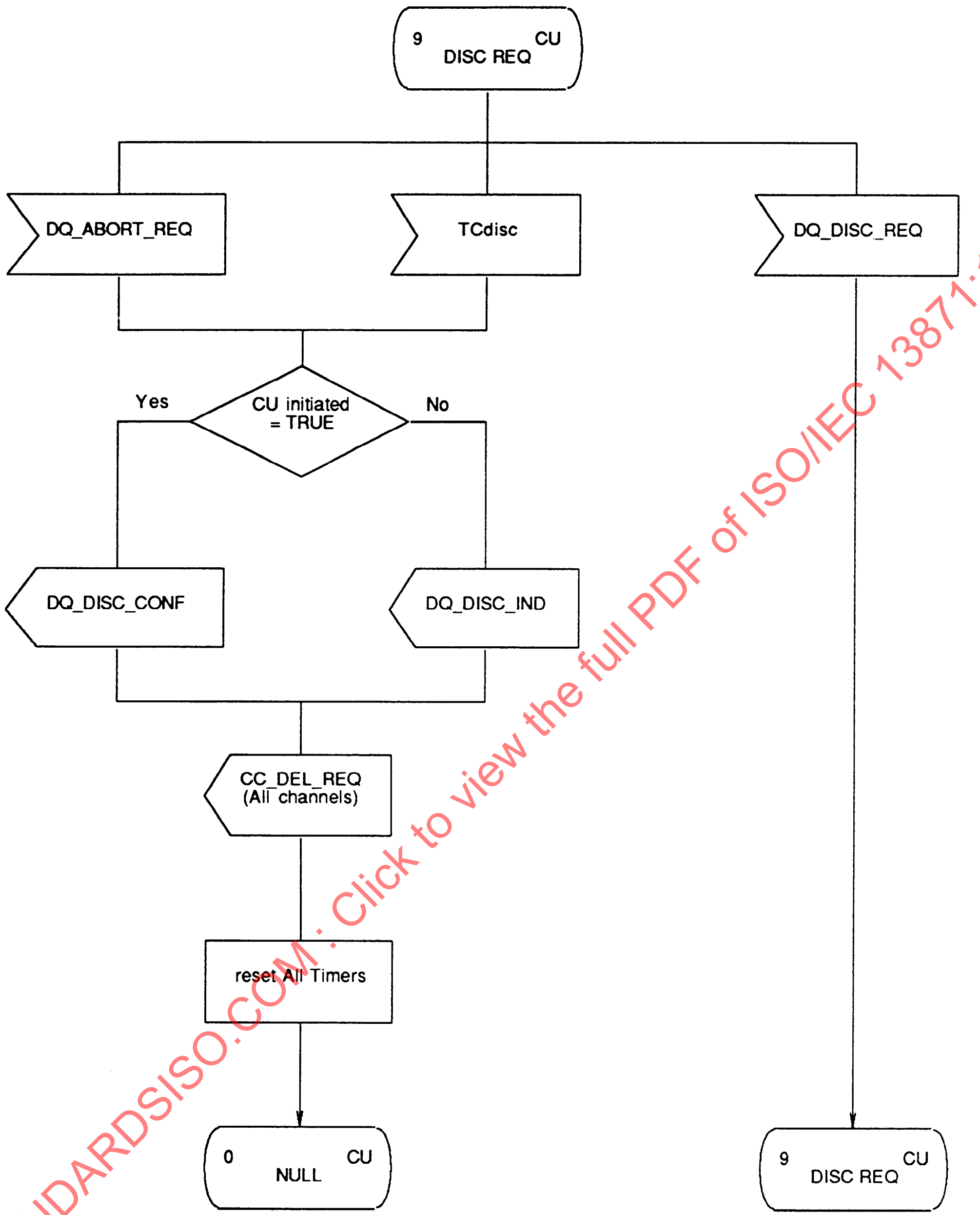


Figure D.18 - Calling Endpoint DISC REQ State

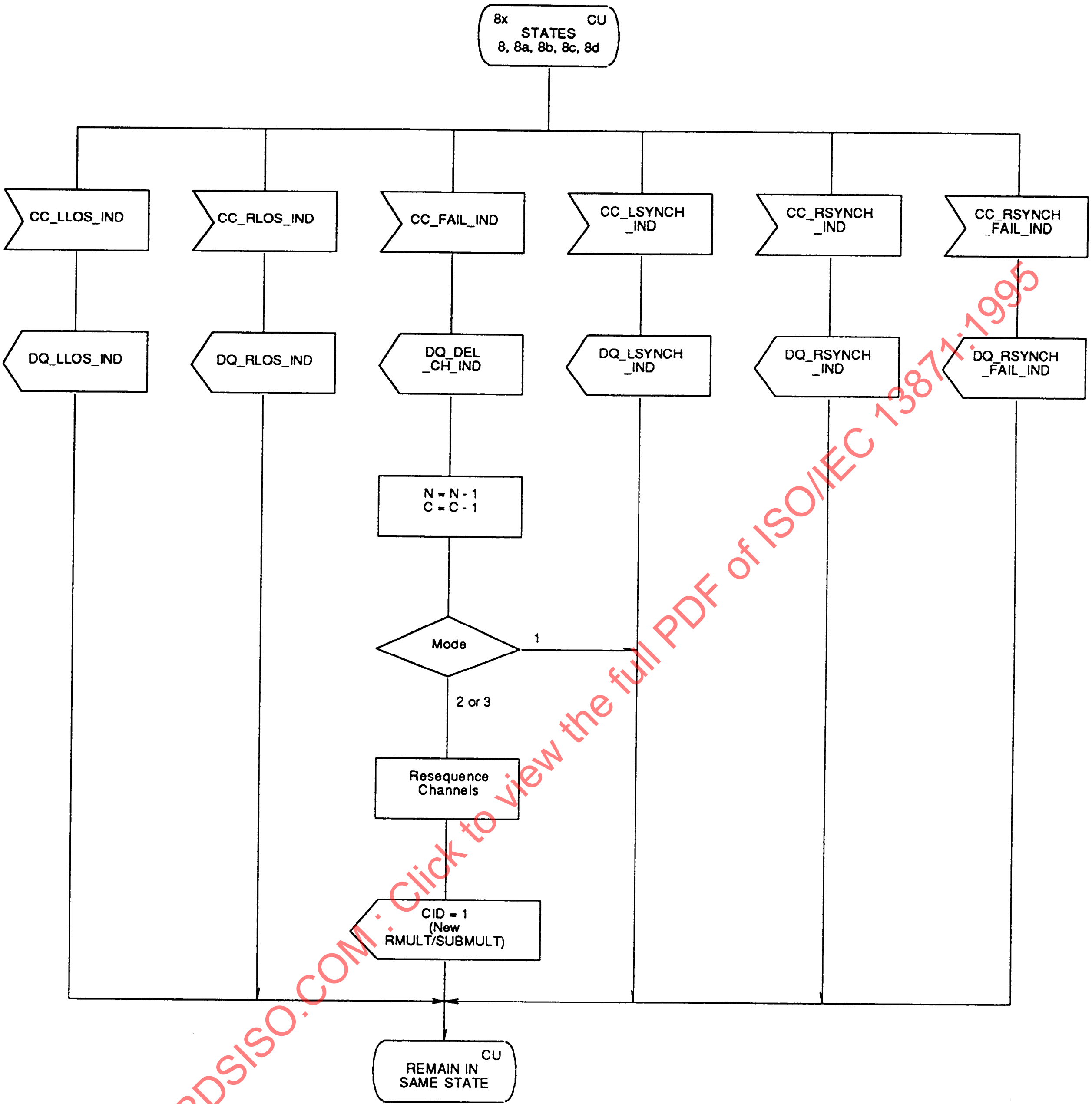


Figure D.19 - Calling Endpoint States 8, 8a, 8b, 8c, 8d

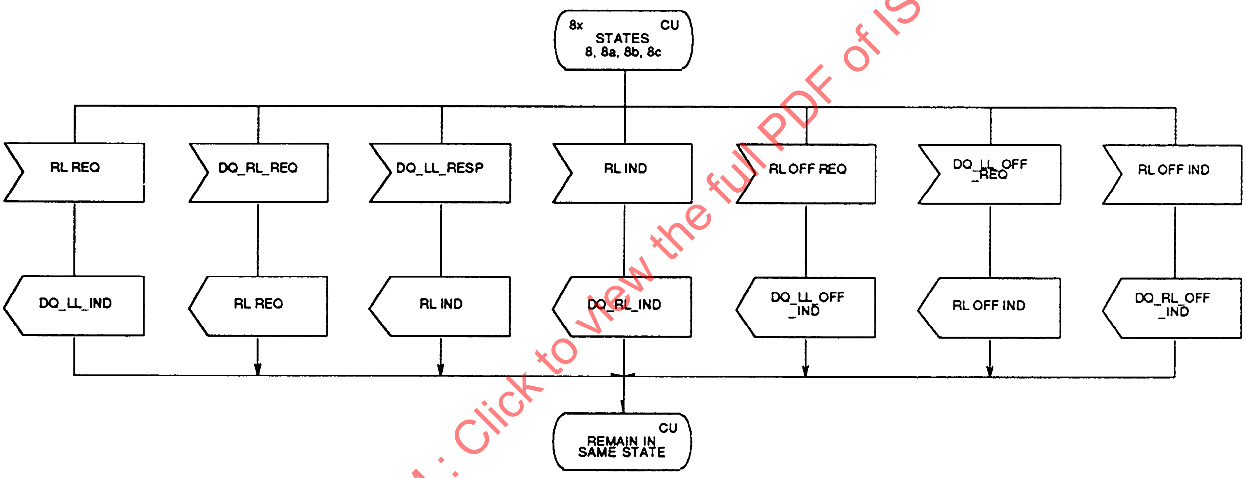


Figure D.20 - Calling Endpoint States 8, 8a, 8b, 8c

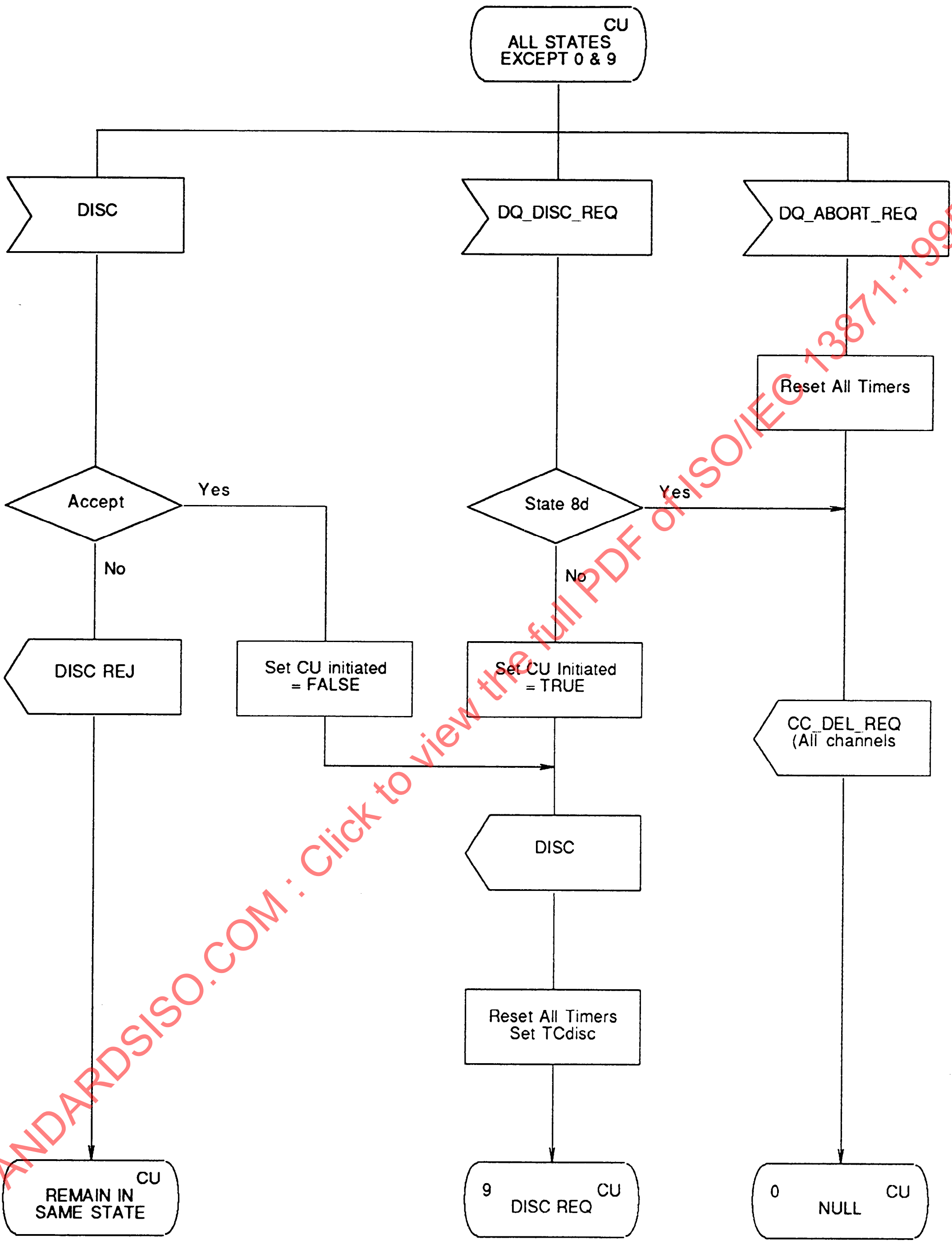


Figure D.21 - Calling Endpoint All States Except 0 and 9

D.6.1.2 Answering Endpoint

This subclause contains the Answering endpoint SDL diagrams for DEQ Negotiation Control.

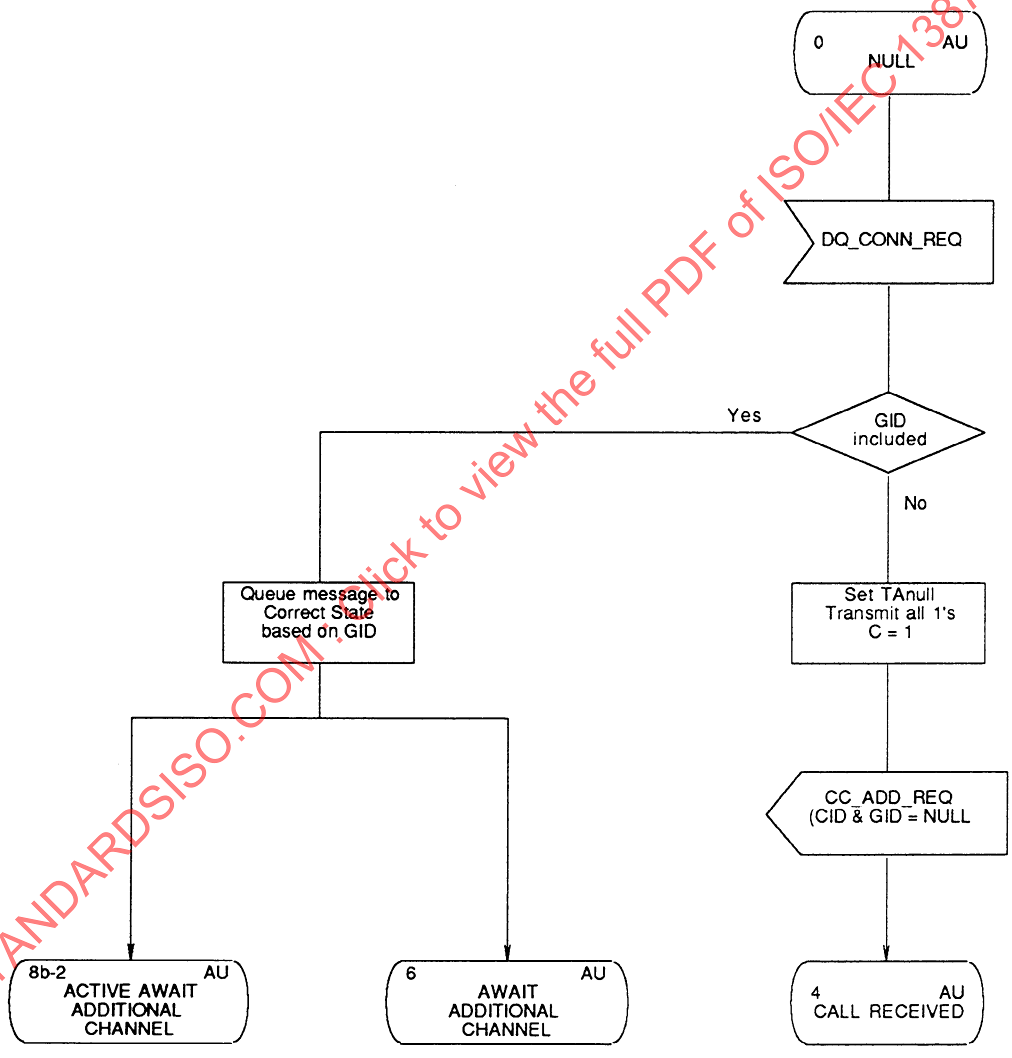


Figure D.22 - Answering Endpoint NULL State

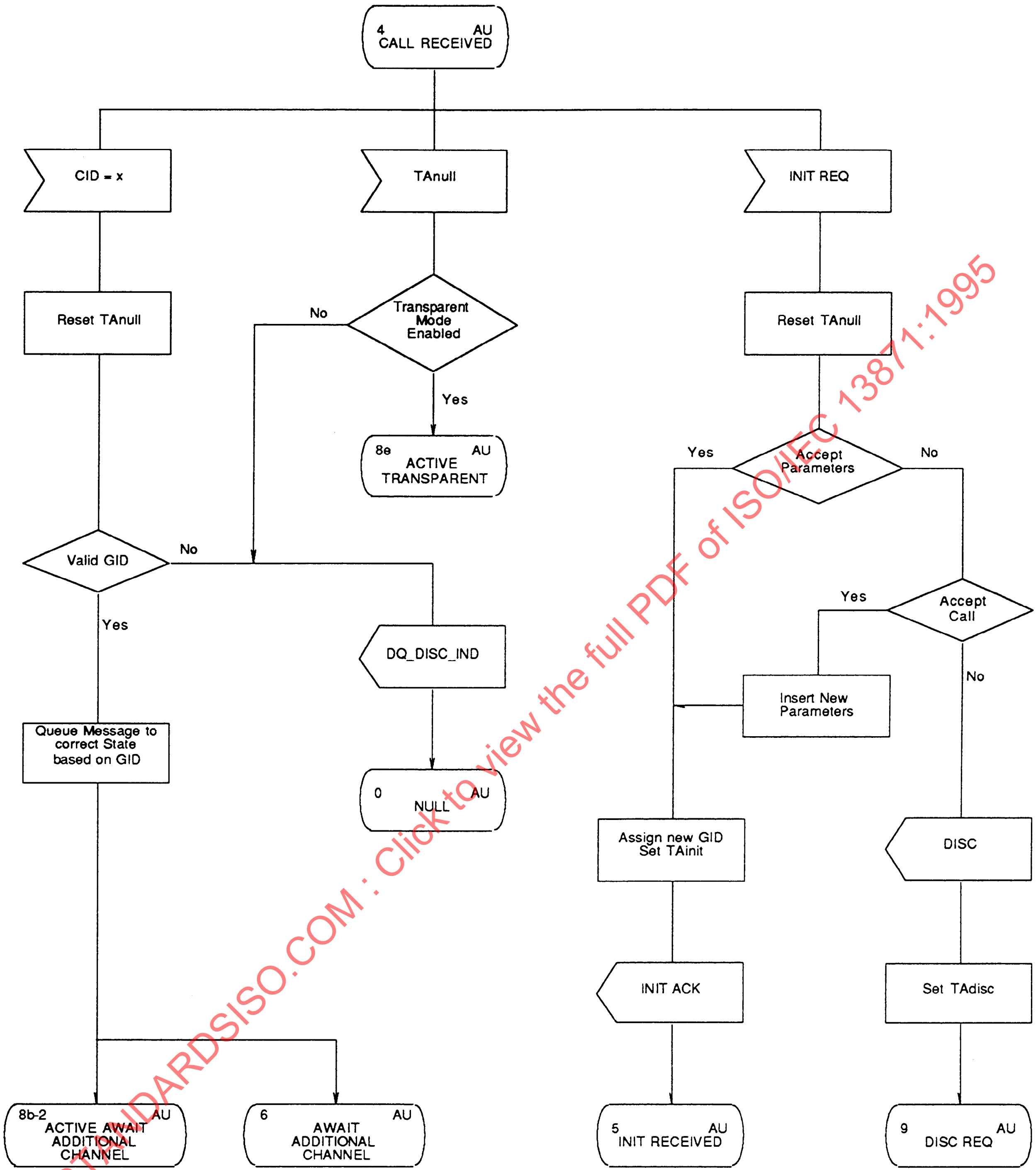


Figure D.23 - Answering Endpoint CALL RECEIVED State