INTERNATIONAL STANDARD

ISO 11519-3

First edition
1994-06-15
AMENDMENT 1
1995-04-01

Road vehicles — Low-speed serial data communication —

Part 3:
Vehicle area network (VAN)

AMENDMENT 1

Véhicules routiers — Communication en série de données à vitesse basse —

Partie 3: Réseau local de véhicule (VAN)

AMENDEMENT 1

Reference number
ISO 11519-3:1994/Amd.1:1995(E)

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

Draft International Standards adopted by the technical committees are circulated to the member bodies for voting. Publication as an International Standard requires approval by at least 75 % of the member bodies casting a vote.

Amendment 1 to International Standard ISO 11519-3:1994 was prepared by Technical Committee ISO/TC 22, *Road vehicles*, Subcommittee SC 3, *Electrical and electronic equipment.*

ISO 11519 consists of the following parts, under the general title *Road vehicles — Low-speed serial data communication* :

— *Part 1: General and definitions*

— *Part 2: Low-speed controller area network (CAN)*

— *Part 3: Vehicle area network (VAN)*

International Organization for Standardization
Case postale 56 • CH-1211 Genève 20 • Switzerland

Printed in Switzerland

Road vehicles — Low-speed serial data communication —

Part 3:
Vehicle area network (VAN)

AMENDMENT 1

Page iv

Insert new page v and the following Introduction.

Introduction

Validation tests on vehicles have been conducted on the basis of ISO 11519-3:1994. The speed of data transmission has been increased up to 250 kTS/s with the same reliability providing that:

— Filter description and system characteristics are more precisely given, and

— parameter specifications of the transceiver are improved.

Amendment 1 to ISO 11519-3 details the necessary changes to the 1994 Standard.

Page 79

Add a new clause before clause 8, to read as follows.

7.6 Alternative up to 250 kTS/s

This clause describes the network interface up to 250 kTS/s.

The definition of TS (Time Slot) is according to clause 7.2.3 Bit encoding of ISO 11519-3 (VAN).

250 kTS/s is corresponding to 125 kbit/s in Manchester coding and 200 kbit/s in Enhanced Manchester coding.

7.6.1 System description

7.6.1.1 Functional block diagram

This block diagram is given in figure 52.

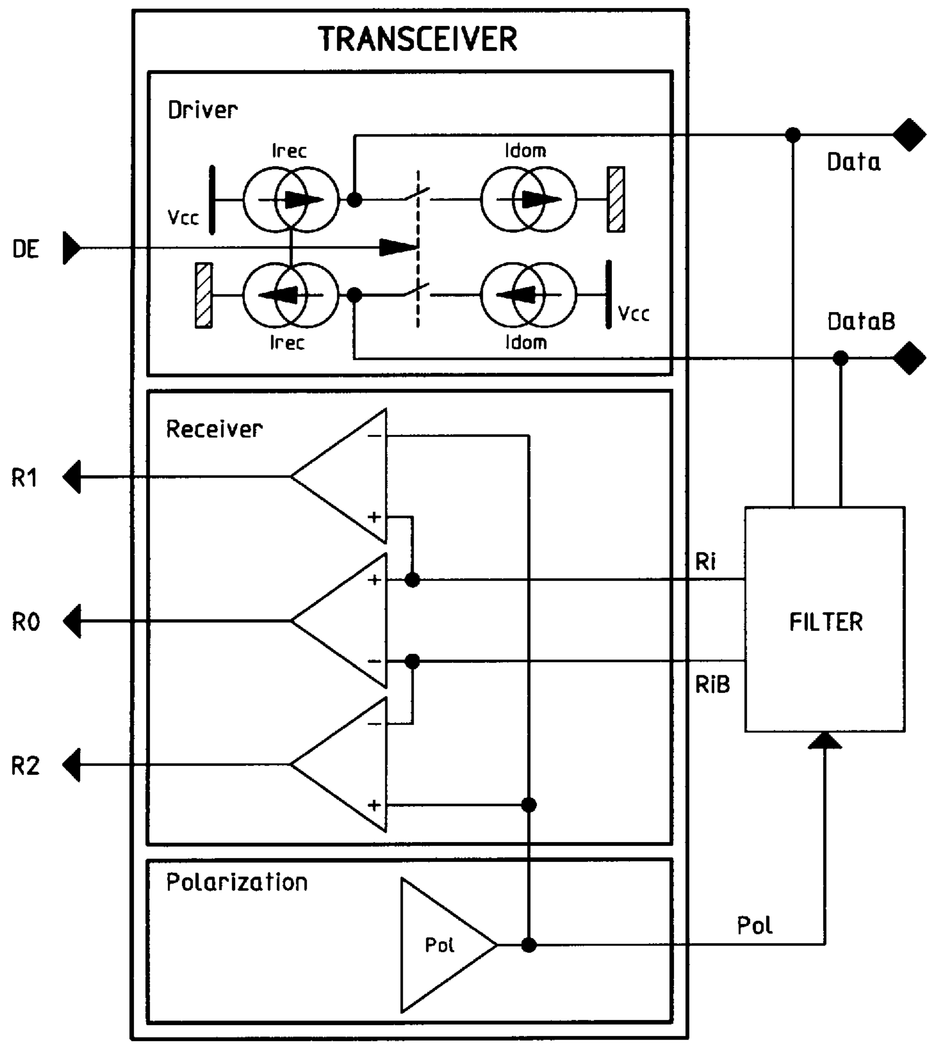
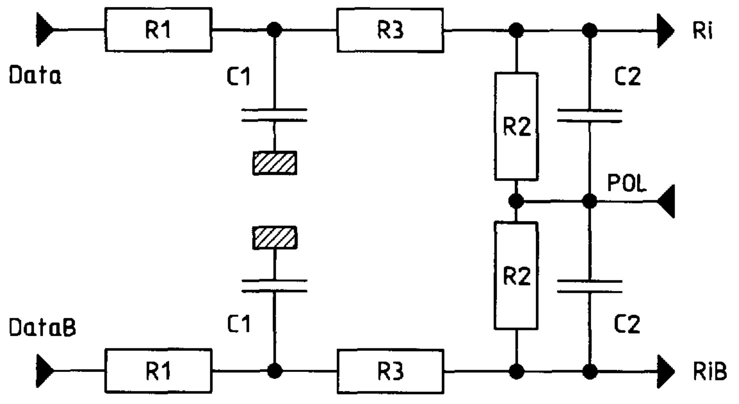


Figure 52 — 250 kTS/s VAN bus interface

7.6.1.2 Filter section

The diagram is given in figure 53, together with the parameters.



Parameter	Recommended value	Tolerance	Comments
C1	100 pF	± 10 %	⩾ 50 V
C2	120 pF		⩾ 25 V
R1	100 Ω		1/4 W continuous
R2	1800 Ω	± 2 %	1/16 W continuous, 0,3 W during 400 ms
R3	2700 Ω		1/4 W continuous 1,25 W during 400 ms

Figure 53 — 250 kTS/s filter

7.6.1.3 Cable characteristics

Parameter	Test condition and description	Value min.	Value typical	Value max.	Unit
Cg-data	Overall capacitance between ground and Data	0		200	pF/module
Cg-dataB	Overall capacitance between ground and DataB	0		200	pF/module
Cdata-dataB	Overall capacitance between Data and DataB	0		100	pF/module
Coverall	= Cg-data + CgdataB + 2 CgData-DataB; by node connected and 2,0 V offset ground in nominal mode or 0,7 V offset in degraded mode.	0		200	pF/module
Lcable	Overall cable length	0		20	m
RISOLg-data	Overall resistor isolation between ground and data	50			kΩ
RISOLg-dataB	Overall resistor isolation between Data and DataB	50			kΩ
RISOLdata-dataB	Overall resistor insulation between Data and DataB	20			kΩ
RCOND	Overall serial resistor of Data or DataB between nodes	0		4	Ω

7.6.1.4 System characteristics

7.6.1.4.1 Number of nodes connected to the network

Parameter	Test condition and description	Value min.	Value typical	Value max.	Unit
Mnode	Number of nodes connected to network	2		12	node

7.6.1.4.2 Timing characteristics

In accordance to VAN protocol at 250 kTS/s, with Enhanced Manchester coding.

Parameter	Test condition and description	Value			
		min.	typical	max.	Unit
TS	Time Slot duration without resynchronization (DE input)	3,96	4	4,04	µs
TDelay	Propagation delay time between DE logical input for one node to R0, R1, and R2 logical outputs from any node	0	0,6	1,25	µs
Tsample	Sample point of protocol controller	12/16	12/16	12/16	TS

7.6.1.4.3 Transient stress capability (automotive application)

The network interface is designed to withstand automobile type transients on Data and DataB lines as defined in ISO 7637 part 1.

Pulse characteristics					
ISO pulse type	Magnitude no load	Duration	Source impedance	Period	Number of pulses
5	V_s = 36,5 V	T_d = 400 ms	R_i = 2 Ω	1 min	5
1	V_s = -50 V	t_d = 2 ms	R_i = 10 Ω	5 s	50

In addition the interface is fully functional during start engine phase.

7.6.1.4.4 Continuous stress capability (automotive application)

The network interface is designed to withstand – 24 V and +24 V voltages on Data and DataB lines.

7.6.1.4.5 Ground offset between nodes

Parameter	Test condition and description	Value			
		min.	typical	max.	Unit
Vnode-nom	Offset between the 2 nodes in nominal mode (for worst case parameters of cable and interface)	-2		+2	V
Vnode-deg	Offset between the 2 nodes in degraded mode (for worst case parameters of cable and interface)	-0,7	0	+0,7	V

Nominal mode: The network uses Data and DataB line for communication.

Degraded mode: In this case, one line is broken-down and the network uses Data or DataB line for communication.

7.6.2 Transceiver description

7.6.2.1 Driver section

Parameter	Test condition and description	Value			
		min.	typical	max.	Unit
Irec	DE = 1; Vdata = -5 V to +3,75 V; VdataB = +1,25 V to +10 V	+1,6	+1,8	+2,0	mA
	DE = 1; Vdata = +3,75 V to +5 V; VdataB = +0 V to +1,25 V	-0,10		+2,0	mA
	DE = 1; Vdata = +5 V to +10 V; VdataB = -5 V to 0 V	-0,10		+0,10	mA
Idom	DE = 0; Vdata = +1,25 V to +10 V; VdataB = -5 V to +3,75 V	+45	+50	+60	mA
	DE = 0; Vdata = 0 V to +1,25 V; VdataB = +3,75 V to +5 V	-0,10		+60	mA
	DE = 0; Vdata = -5 V to 0 V; VdataB = +5 V to +10 V	-0,10		+0,10	mA
	DE = 1; Vdata = -5 V to +10 V; VdataB = -5 V to +10 V	-0,10		+0,10	mA
OverShot	Current overshot during transition recessive --> dominant			10	mA
M-I	Static matching of current output	-5		+5	%
TON	Propagation delay of dominant current from recessive state to dominant state			200	ns
TOFF	Propagation delay of dominant current from dominant state to recessive state			200	ns

7.6.2.2 Receiver section

Parameter	Test condition and description	Value min.	Value typical	Value max.	Unit
VMCR	Common mode	0,5		4,5	V
ZMC	Common mode impedance	100			kΩ
ZMD	Differencial mode impedance	100			kΩ
CMC	Input capacitance between input and ground			10	pF
CMD	Differential input capacitance between inputs			10	pF
OFFr	R1 and R2 comparators offset	-25	0	+25	mV
HYSdif	Differential comparator input hysteresis	150	200	250	mV
HYSSr	R1 and R2 comparator input hysteresis	150	200	250	mV
TDEL	Propagation delay for high to low transition, input overdrive 50 mV			150	ns
TEDH	Propagation delay for low to high transition, input overdrive 50 mV			150	ns

7.6.2.3 Polarization section

Parameter	Test condition and description		Value min.	Value typical	Value max.	Unit
ZGT	Output impedance POL used by filter ± 3 mA output POL current	Frenquency = 1MHz			200	Ω
		Frequency ≤ 1 kHz			100	Ω
VGTint	Internal reference for R1 and R2 used by comparators		2,375	2,5	2,625	V
VGText	Output voltage POL ± 2 mA used by filter		2,375	2,5	2,625	V

ICS 43 040 10

Descriptors: road vehicles, electronic equipment, data communication equipment, data processing, information interchange, local area networks, vehicle area networks, data transmission, data link layer, physical layer.

Price based on 5 pages

INTERNATIONAL STANDARD

ISO 11519-3

First edition
1994-06-15

Road vehicles — Low-speed serial data communication —

Part 3:
Vehicle area network (VAN)

Véhicules routiers — Communication en série de données à basse vitesse —

Partie 3: Réseau local de véhicule (VAN)

Reference number
ISO 11519-3:1994(E)

Contents

International Organization for Standardization
Case Postale 56 • CH-1211 Genève 20 • Switzerland

Printed in Switzerland

Annexes

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

Draft International Standards adopted by the technical committees are circulated to the member bodies for voting. Publication as an International Standard requires approval by at least 75 % of the member bodies casting a vote.

International Standard ISO 11519-3 was prepared by Technical Committee ISO/TC 22, *Road vehicles*, Sub-Committee SC 3, *Electrical and electronic equipment*.

ISO 11519 consists of the following parts, under the general title *Road vehicles — Low-speed serial data communication*:

— *Part 1: General and definitions*

— *Part 2: Low-speed controller area network (CAN)*

— *Part 3: Vehicle area network (VAN)*

— *Part 4: Class B data communication network interface (J1850)*

Annexes A and B form an integral part of this part of ISO 11519.

Road vehicles — Low-speed serial data communication —

Part 3:
Vehicle area network (VAN)

1 Scope

This part of ISO 11519 specifies the data link layer and the physical layer of the Vehicle Area Network (VAN), communications network up to 125 kbit/s, for road vehicle application. The VAN is an access-method-oriented multimaster-multislave which allows optimized request/response management by special method of handling a remote transmission request (retaining access to the medium to allow insertion of a response).

This part of ISO 11519 defines the general architecture of the low-speed communication network up 125 kbits/s and the content of the data link layer, and the physical layer for transmission between different types of electronic modules on board road vehicles.

2 Normative references

The following standards contain provisions which, through reference in this text, constitute provisions of this part of ISO 11519. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this part of ISO 11519 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/TR 8509:1987, *Information processing systems — Open Systems Interconnection — Service conventions.*

ISO 8802-2:1989, *Information processing systems — Local area networks — Part 2: Logical link control.*

3 Definitions and abbreviations

3.1 Definitions

For the purposes of this part of ISO 11519, the following definitions apply.

3.1.1 acknowledgement field (ACK): Field used by a module concerned to indicate correct interpretation of the frame by a receiver.

3.1.2 autonomous module: Module which can initiate data sending over the transmission medium.

3.1.3 bitwise arbitration: Arbitration technique which allows a priority message to take precedence on the bus and dominate other messages of lower priority with which it collides. The collision is thus not destructive for the highest-priority message. This bitwise arbitration technique is based on the use of dominant and recessive states on the bus, with the dominant states taking precedence over the recessive bits.

In the event of a collision in the arbitration field (simultaneous sending of recessive and dominant bits), only those modules sending a dominant bit will keep on transmitting, while the others will cease to transmit. This process is repeated for each bit of the arbitration field.

Bus value — The bus can take 2 electrical states:
dominant (D) corresponds to a logic level "0" ,
recessive (R) corresponds to a logic level "1" .

3.1.4 code violation: Any error that converts a bit or other physical symbol into an out-of-code symbol.

3.1.5 collision; interference: Physical phenomenon that occurs when several signals are superimposed on one another, whether they are of internal origin (modules connected to the bus) or external origin (noise).

3.1.6 collision detection: Collision detected by a sending module when interference occurs on the bus and modifies the signal transmitted (more precisely, the signal received is different from the signal sent).

3.1.7 command field (COM): Field containing command information associated with the frame.

3.1.8 contention: Situation that arises when several modules start transmitting simultaneously on the communication bus.

3.1.9 data field (DAT): Part of the frame containing data. The field consists of a whole number of bytes.

3.1.10 data transmission: Process by which encoded data can be sent over a transmission medium sequentially in binary form.

3.1.11 end of data (EOD): Part of the frame indicating the end of data. The EOD is located just after the Frame Check Sequence (FCS).

3.1.12 end of frame (EOF): Part of the frame indicating the end of a frame.

3.1.13 extensibility: Situation where modules can be added to the network without having to change the software or hardware of any module for an existing application, within the limits of the communication layers specified in this document.

3.1.14 MAC frame: Sequence of fields containing either:

a start of frame field;
an identifier field;
a command field;
a data field;
a frame check sequence field;
an end of data field;
an acknowledgement field;
an end of frame field.

or

a start of frame field;
an identifier field;
a command field;
a frame check sequence field;
an end of data field;
an acknowledgement field;
an end of frame field.

Each frame is separated by an interframe spacing field.

3.1.15 frame check sequence (FCS): Part of the frame which checks its integrity. In the present case, this function is performed by means of a Cyclic Redundancy Check (CRC).

3.1.16 identifier field (IDEN): Part of the frame following the SOF, which identifies and specifies the data conveyed in the frame.

3.1.17 interframe spacing (IFS): Minimum time interval locally required between the sending of two consecutive frames, which is controlled by the MAC sublayer.

3.1.18 module: Physical entity connected to the network, capable of receiving and/or sending data via the medium.

3.1.19 remote transmission request: By sending a data request, a module that wishes a data unit can request another module to send it the corresponding data. The data unit can be sent either immediately in the same frame or later in a separate frame identified by the same identifier.

3.1.20 slave module: Module which can

— receive data

— send data when requested, by means of an in-frame response mechanism.

3.1.21 start of frame (SOF): Part of the frame which indicates the start of the frame and synchronizes the receiving modules' clocks.

3.1.22 synchronous access module: Module which can initiate transmission only after a Start of Frame (SOF) character appears on the bus.

3.2 List of abbreviations

ACK	Acknowledge
ADT	Acknowledged Data Transfer
BR	Bit Rate
BT	Bit Time
D	Dominant State
DL	Data Link
EOD	End Of Data
EOF	End Of Frame
FCS	Frame Check Sequence
IFS	Interframe Spacing
LLC	Logical Link Control
LSB	Least Significant Bit
LSDU	Link Service Data Unit

MAC	Medium Access Control
MDI	Medium Dependent Interface
MSB	Most Significant Bit
NADT	Not Acknowledged Data Transfer
OSI	Open Systems Interconnection
PL	Physical Layer
PLS	Physical Signalling
PMA	Physical Medium Attachment
Q/R	Question/Response Frame
RD	Recessive State
RAK	Request Acknowledge
RT	Remote Transmission
RTR	Remote Transmission Request
SOF	Start Of Frame
TS	Time Slot

4 Presentation of architecture

4.1 General

The objectives of the VAN are to interconnect different types of electronic modules on board a vehicle and to transmit messages having different priority levels.

The VAN is an asynchronous data transmission system which allows the transfer of packets of data.

The messages handled can be typically:

— messages of 1 byte, to write or to read from a slave peripheral module;

— messages from 0 to 28 bytes, to exchange parameters and/or events between the different autonomous modules;

— long messages segmented by the user.

The document allows the possibility of interconnecting heterogeneous modules including, among others, very simple slave modules.

The implications for the frame format and layer design are:

— the use of a special Start Of Frame field which can correct the local clocks of the simple modules and establish a common time base;

— the possibility of chaining the response of a module in the same request frame concerning it;

— the possibility of direct in-frame acknowledgement.

4.2 Reference to OSI model

The VAN architecture complies with the ISO reference model for Open Systems Interconnection (OSI) with respect to the breakdown by layers. This breakdown is shown in figure 1.

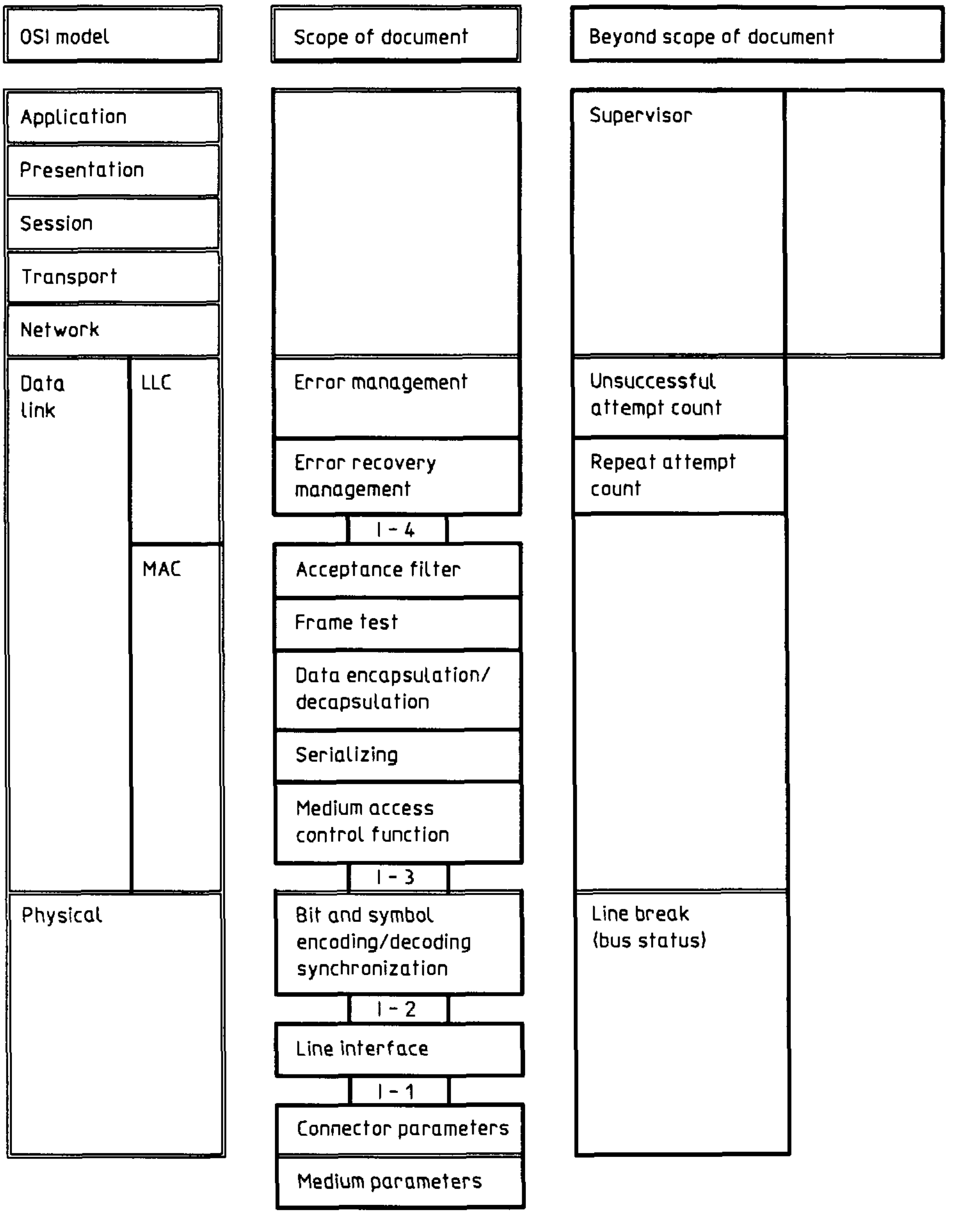


Figure 1 — Breakdown by layers in accordance with OSI model

5 Description of LLC sublayer

The LLC (Logical Link Control) sublayer is the upper part of the data link layer (layer 2 of the ISO reference model — see figure 1).

The objective is to provide a multipoint data transmission service, in connectionless mode.

The architecture of the VAN, described in this part of ISO 11519, allows two types of service which can be provided by the LLC sublayer (in accordance with the International Standard on LAN architecture, ISO 8802-2):

— **The class I LLC service** which offers a data transfer service without acknowledgement: this provides a minimum level of service and requires the implementation of greatly reduced functionalities at the LLC layer protocol level;

— **The class III LLC service** which offers, in addition to class I services (point-to-point, multipoint or broadcast data transfer), check services on the operation of the data link (acknowledgement, flow control, sequencing and error recovery).

Generally speaking, the LLC sublayer provides the functionalities needed for supervision of the data links and should incorporate at its level recovery capabilities in the event of failure of a module.

The description of the services of the LLC sublayer is given in 5.1. The precise definition of LLC procedure elements required for implementation of these services is not yet contained in this part of ISO 11519.

5.1 LLC service specifications

5.1.1 Object

This subclause specifies the services which are provided by the LLC sublayer to the LLC user in the framework of the VAN architectural model defined in this part of ISO 11519 (see figure 1 for the description of the layers).

5.1.2 General presentation of LLC services

The services provided by the LLC sublayer are designed to allow exchange of packets between the local user entity (LLC user) and peer entities (LLC users) which are connected to the communication bus.

In order to provide these services, the LLC sublayer builds its functions on the services provided by the next lower sublayer (MAC services: see 6.1).

There are three types of services provided by the LLC sublayer. All these services are connectionless oriented (i.e. they do not need the establishment and maintenance of a connection, see ISO 8802-2).

5.1.2.1 Unacknowledged data transfer service

The unacknowledged data transfer service provides the means by which a transfer entity can send Link Service Data Units (LSDU) to transfer entities.

This service can be used for point-to-point, multipoint or broadcast exchanges with maximum efficiency.

5.1.2.2 Acknowledged data transfer service

The acknowledged data unit exchange service provides the means by which a transfer entity can exchange Link Service Data Units (LSDU) to another transfer entity with the guarantee that the LSDU has been transmitted correctly.

This service can be based either on an LLC-level acknowledgement mechanism LLC Type 3 when a high level of reliability is required or on the acknowledge capability provided by the MAC sublayer (see 6.3.2.1.1).

This part of ISO 11519 does not specify the functions of the LLC sublayer necessary to provide acknowledged data unit exchanges (LLC type 3).

The acknowledged data transfer service is to be used for point-to-point exchanges only.

5.1.2.3 Remote data transmission service

The remote data transmission service provides the means by which a transfer entity at one module can request (consume) a data unit (LSDU) that is located (produced) at some other remote module.

This service can be used for point-to-point, multipoint or broadcast exchanges, with respect to the transferred data units (LSDU).

This service is divided in two subservices:

— the response preparation service,

— the reply service.

5.1.3 Description of LLC service interactions

5.1.3.1 Service specification method and notation

This subclause describes the service aspects of the LLC sublayer (corresponding to the various functionalities of the LLC sublayer provided for users located in the upper layer).

This is an abstract representation (or model) of an interface between the LLC sublayer and an LLC service user, independent of any specific implementation.

The LLC service definition proposed hereafter complies with the ISO reference model: in particular, it uses the associated service notations in ISO/TR 8509.

The service primitives used are of three types: request, indication, confirm. Their meaning is summarized below:

Request The request primitive is sent by a (N)-user to the (N)-layer or (N)-sublayer to request initiation of a service.

Indication The indication primitive is sent by the (N)-layer or (N)-sublayer to a user of the (N)-service to indicate that an event internal to the (N)-(sub-)layer has occurred and that the event in question is significant for the (N)-user. An indication can be triggered by a service request executed previously or by an internal event in the (N)-(sub-)layer.

Confirm The confirm primitive is sent by (N)-layer to a (N)-user to retrieve the results associated with the previous service request.

The links (logical and temporal) between indication and confirm are diverse, depending basically on the characteristics of each service.

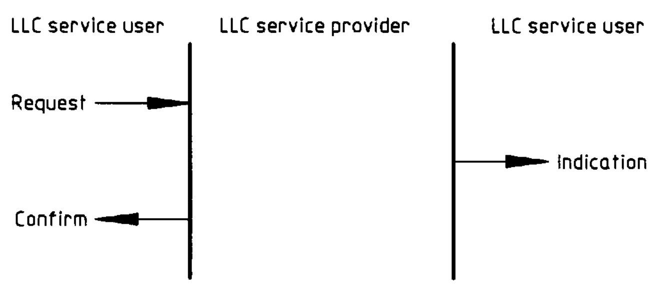


Figure 2 — Schematic diagram of interactions between adjacent layers

5.1.3.2 Description of service interactions

Table 1 gives a list of the service primitives characterizing each of the LLC service elements described in this part of ISO 11519. Subclause 5.1.3.3 gives a detailed description of each of these service primitives.

Table 1 — List of LLC service primitives

Unacknowledged data transfer	
DL-DATA.request	Request for unacknowledged data transfer.
DL-DATA.indication	Indication of unacknowledged data transfer.
DL-DATA.confirm	Confirm of unacknowledged data transfer.
Acknowledged data transfer	
DL-ACK-DATA.request	Request for acknowledged data transfer.
DL-ACK-DATA.indication	Indication of acknowledged data transfer.
DL-ACK-DATA.confirm	Confirm of acknowledged data transfer.
Remote transmission service with acknowledgement	
Reply service	
DL-REPLY.request	Request for remote transmission.
DL-REPLY.indication	Indication of remote transmission.
DL-REPLY.confirm	Confirm of remote transmission.
Response preparation service	
DL-REPLY-UPDATE.request	Request for response preparation.
DL-REPLY-UPDATE.confirm	Confirm of response preparation.
Remote transmission service without acknowledgement	
Reply service	
DL-REPLY.request	Request for remote transmission.
DL-REPLY.indication	Indication of remote transmission.
DL-REPLY.confirm	Confirm of remote transmission.
Response preparation service	
DL-REPLY-UPDATE.request	Request for response preparation.
DL-REPLY-UPDATE.confirm	Confirm of response preparation.

5.1.3.3 Definition of LLC service primitives

This subclause gives the definition of LLC service primitives such as presented in 5.1.3.2 with their associated parameters.

The following parameters are used to formalize the kind of interactions which appear at the transfer/LLC service interface:

— The IDEN parameter is used to identify the data which are exchanged using this interaction (sent, received or requested in a remote transmission);

— The DATA parameter represents the user data transferred (LSDU) using this primitive;

— The SERVICE CLASS specifies whether or not the LLC sublayer uses the acknowledge capability in the medium access control sublayer for the data unit transmission;

— The NREP-DL indicates the maximum number of retransmissions that the data link layer may achieve (using the recovery capabilities of both LLC and MAC sublayers) to provide the complete execution of the corresponding request;

— The NREP-ACK parameter indicates the maximum number of retransmissions that the data link layer may achieve by lack of acknowledgement of the receiving entity;

— The STATUS parameter specifies whether the corresponding request was executed with success or not. In the latter case, it indicates the type of failure.

Table 2 gives the list of parameters associated to the primitives for each LLC service.

Table 2 — Parameters

LLC services	List of parameters	Type of interaction		
		REQ	IND	CONF
Unacknowledged data transfer	IDEN	X	X	X
	DATA	X	X	—
	NREP-DL	X	—	—
	STATUS	—	—	X
Acknowledged data transfer	IDEN	X	X	X
	DATA	X	X	—
	SERVICE-CLASS	X	X	X
	NREP-DL	X	—	—
	NREP-ACK	X	—	—
	STATUS	—	—	X
Reply	IDEN	X	X	X
	DATA	—	—	X
	SERVICE-CLASS	X	X	X
	NREP-DL	X	—	—
	STATUS	—	—	X
Response preparation	IDEN	X	—	X
	DATA	X	—	—
	STATUS	—	—	X

5.2 Error management at LLC level

Transmission errors are indicated by the MAC sublayer to the LLC sublayer.

Error recovery procedures are managed by the LLC sublayer.

Recovery management after loss of arbitration by the LLC sublayer is optional. Repetitions of this type are not accounted for (see 5.3). They can be indicated to the network administration.

5.2.1 Errors in transmit mode

When transmit mode errors are indicated by the MAC sublayer, the LLC sublayer can repeat the transmission request following a time delay.

5.2.2 Errors in receive mode

In receive mode, errors indicated to the LLC sublayer by the MAC sublayer correspond:

— either to an error indicated by the physical layer to the MAC sublayer (code violation, synchronization, etc.);

— or to an invalid frame detected by the MAC sublayer.

If an error is detected in receive mode, the MAC sublayer notifies the LLC sublayer of the type of error (see 6.3.5).

5.3 Error recovery management at LLC level

After a loss of arbitration, an LLC level recovery can occur in the following conditions:

— another frame send attempt is made without further delay, after sending the frame which caused the collision;

— There is no limit to the number of repeat attempts at the LLC sublayer level;

— a specific request by a user of the LLC sublayer can interrupt further send attempts.

A send attempt can only be repeated after sending the EOF character and waiting for the interframe spacing (IFS).

6 Description of the MAC sublayer

This clause describes the functions, main characteristics (and subsequently the protocol) of the MAC (Medium Access Control) sublayer of the VAN.

The MAC sublayer is the lower part of the data link layer of the ISO reference model (see figure 1). It contains the medium access facilities used for sharing a data communication bus between two or more interconnected modules in the vehicle, and the other functionalities required for the implementation of a data link layer (data serializing and deserializing, interface with the physical layer).

This part of ISO 11519 includes an access method specification (general principles) and the corresponding parameter values for implementation on a medium such as that considered in this document (differential pair for bit rates ranging from 10 kbits/s to 125 kbits/s). For the latest technical developments, the limit of the data transfer is 250 kbit/s.

6.1 Specification of MAC service

6.1.1 Object

Subclause 6.1 specifies the services provided by the MAC sublayer to the LLC sublayer within the VAN architecture defined in this document (see figure 1 for the breakdown by layers).

It includes a general description of MAC services (6.1.2) and, for each type of module that is defined, it specifies the list of available services (table 3).

Table 3 — MAC services

Service categories	Role	Types of modules	
		Autonomous	Slave
Unacknowledged data transfer	P(=I)	Yes	No
	C	Yes	Yes
Acknowledged data transfer	P(=I)	Yes	Yes
	C	Yes	Yes
Remote transmission with immediate response	I	Yes	No
	P	Yes	Yes
	C	Yes	Yes
Remote transmission with deferred response	I	Yes	No
	P	Yes	No
	C	Yes	No

Subclause 6.1.3 gives a detailed specification of service primitives: they represent a conceptual description of interactions between the MAC sublayer and a MAC-service user which are necessary for the execution of these services.

6.1.2 General description of MAC service

The services provided by the MAC sublayer are designed to allow data exchange between the local user entity (LLC sublayer) and peer entities (LLC sublayers) connected to the communication bus.

The MAC services proposed in this part of ISO 11519 are of four types:

— **Unacknowledged data transfer service:** This enables the local entity to exchange data over a data link without call connection or acknowledgement of the receiving LLC entities. This simple data transfer is point-to-point, or multipoint, or broadcast-oriented.

— **Acknowledged data transfer service:** This enables the local LLC entity to transfer data to another LLC entity with a request for acknowledgement by the receiving LLC entity. This data transfer mode is point-to-point oriented in connectionless mode.

— **Remote transmission with immediate response service:** This service authorizes the local LLC entity to request the transfer of data produced by another LLC entity at some remote module.

The polled data must be previously prepared and stored at the MAC sublayer level at the producer module: they are sent immediately in the request frame (immediate response: see 6.2.1).

This service can be used for point-to-point, multipoint or broadcast exchanges, with respect to transferred data (MSDU): the transferred data are received by one or more LLC entities, including the entity which initiated the service.

Using the service, the MAC acknowledge capability may be used: in case of success (immediate response); acknowledgement is given by the MAC entity which initiated the remote transfer (the transferred data are considered valid for a receiving MAC entity only in case of a positive acknowledgement: see 6.2.6).

This service is subdivided into two subservices:

- the response preparation service for locally updating the data buffer,
- the reply service for remote access to data.

— **Remote transmission with deferred response service:** This service authorizes the local LLC entity to request the transfer of data produced by another LLC entity at some remote module: the polled data are sent in a deferred mode, using a separate frame identified by the same identifier.

MAC services and module types allow interconnection of heterogeneous modules corresponding to various functional requirements.

This part of ISO 11519 introduces two types of module which are taken into account in specification of VAN layers 1 and 2 (autonomous module and slave module). This subclause specifies the characteristic functional requirements (profiles) for each type of module.

Table 3 specifies the MAC services that may be available depending on the type of the module (autonomous or slave). It states for each category of service which role the module can take depending on its type. The list of roles is the following:

Producer (P) A module is said to be a producer while executing a service if it held the functionality of sending user data (the LLC entity is a data source).

P(=I) means initiator is also producer.

Consumer (C) A module is said to be a consumer when it held the functionality to receive user data (the LLC entity is a data sink).

Initiator (I) A module is said to be an initiator if it has the capability to initiate the execution of the service.

6.1.3 Detailed description of MAC service

6.1.3.1 Service specification method and notation

This subclause describes the service aspects of the MAC sublayer (corresponding to the various functionalities of the MAC sublayer provided for users located in the upper layer).

This is an abstract representation (or model) of an interface between the MAC sublayer and a MAC service user, independent of any specific implementation.

The MAC service definition proposed hereafter complies with the ISO reference model: in particular, it uses the associated service notations in ISO/TR 8509.

The service primitives used are classified into three types: request, indication, confirm. Their meanings are summarized below:

Request The request primitive is sent by an (N)-user to the (N)-layer or (N)-sublayer to request initiation of a service.

Indication The indication primitive is sent by the (N)-layer or (N)-sublayer to a user of the (N)-service to indicate that an event internal to the (N)-(sub-)layer has occurred and that the event in question is significant for the (N)-user. An indication can be triggered by a service request executed previously or by an internal event in the (N)-(sub-)layer.

Confirm The confirm primitive is sent by (N)-layer to an (N)-user to retrieve the results associated with the previous service request.

The links (logical and temporal) between indication and confirm are diverse, depending basically on the characteristics of each service (see 6.1.3.3).

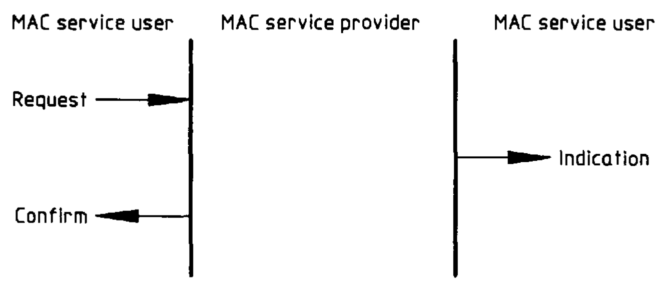


Figure 3 — Schematic diagram of interactions between adjacent layers

6.1.3.2 Description of service interactions

Table 4 gives a list of the service primitives characterizing each of the MAC service elements described in this part of ISO 11519. Subclause 6.1.3.3 gives a detailed description of each of these service primitives.

6.1.3.3 Detailed specification of MAC services

This subclause describes in detail the primitives and their associated parameters corresponding to the various MAC services described in 6.1.3.2.

The IDEN parameter is used to identify the data which are exchanged using this interaction (sent, received or requested in a remote transmission).

The EXT parameter is a 1-bit field reserved for subsequent applications. In this part of ISO 11519 the parameter's value is set at 1 (recessive bit). The DATA parameter specifies the user data (data unit of the MAC service) to be transferred by the MAC entity.

It is assumed that there is sufficient information in this parameter for the MAC entity to be able to determine the data length (possibly nil).

The transmission-status, reception-status and transfer-status parameters specify the level of success in transmitting the corresponding data.

6.1.3.3.1 Primitives associated with unacknowledged data transfer service (NDT)

6.1.3.3.1.1 MA-DATA.request

a) **Function**

This primitive can initiate unacknowledged data transfer between the local LLC entity and one or more receiving LLC entities.

Table 4 — List of MAC service primitives

Unacknowledged data transfer	
MA-DATA.request	Request for unacknowledged data transfer.
MA-DATA.indication	Indication of unacknowledged data transfer.
MA-DATA.confirm	Confirm of unacknowledged data transfer.
Acknowledged data transfer	
MA-ACK-DATA.request	Request for acknowledged data transfer.
MA-ACK-DATA.indication	Indication of acknowledged data transfer.
MA-ACK-DATA.confirm	Confirm of acknowledged data transfer.
Remote transmission service with acknowledgement	
Reply service	
MA-REPLY.request	Request for remote transmission.
MA-Reply.indication	Indication of remote transmission.
MA-REPLY.confirm	Confirm of remote transmission.
Response preparation service	
MA-REPLY-UPDATE.request	Request for response preparation.
MA-REPLY-UPDATE.confirm	Confirm of response preparation.
Remote transmission service without acknowledgement	
Reply service	
MA-REPLY.request	Request for remote transmission.
MA-REPLY.indication	Indication of remote transmission.
MA-REPLY.confirm	Confirm of remote transmission.
Response preparation service	
MA-REPLY-UPDATE.request	Request for response preparation.
MA-REPLY-UPDATE.confirm	Confirm of response preparation.

b) **Parameters associated with MA-DATA request primitive**

This primitive contains the following parameters:

— MA-DATA.request (IDEN, EXT, DATA).

The IDEN field can identify the modules concerned by the transferred data (DATA). The EXT parameter is an extension field which may be used by the LLC sublayer (an extension bit conveyed in the data frame: see 6.2.3.3).

The DATA parameter must contain sufficient information to indicate to the MAC sublayer the length of the data to be transferred.

Figure 4 gives examples of possible exchanges for the primitives associated with the unacknowledged data transfer service.

c) **Operations associated with this primitive**

This primitive is sent by a LLC entity to the MAC sublayer whenever it needs to transfer data in NADT mode (unacknowledged) to one or more remote LLC entities.

Upon receiving this primitive, the MAC sublayer prepares a data frame by adding the frame-specific fields (identifier, command field including the extension bit and RAK bit indicating that acknowledgement is not requested, see 6.2).

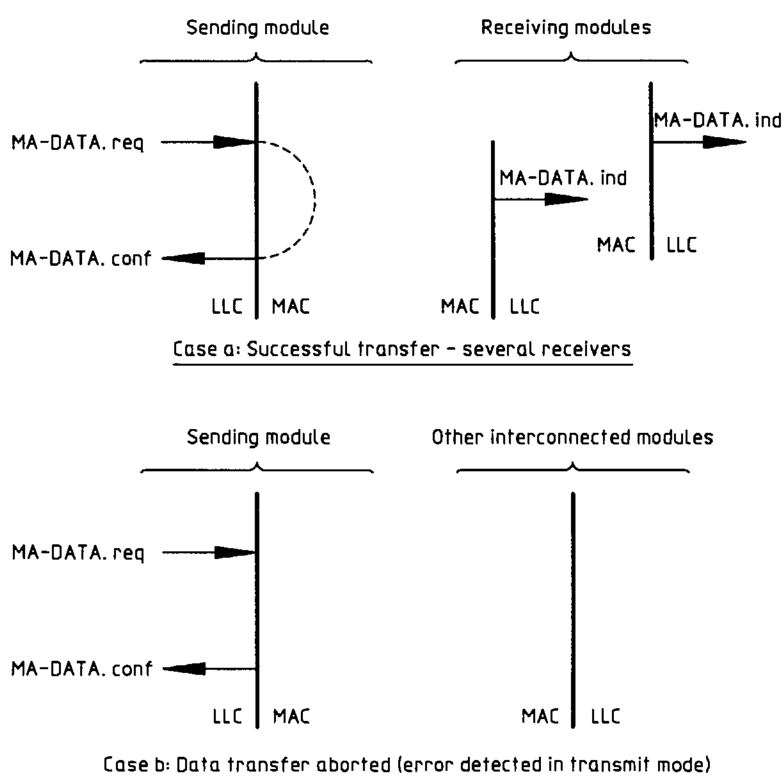


Figure 4 — Example of primitive exchanges in NDT service

6.1.3.3.1.2 MA-DATA.indication

a) **Function**

This primitive allows the MAC sublayer to indicate an unacknowledged data transfer to an entity of the LLC sublayer.

b) **Parameters associated with MA-DATA.indication primitive**

This primitive contains the parameters:

— MA-DATA.indication (IDEN, EXT, DATA, RECEPTION-STATUS).

The IDEN and EXT parameters are the same as those provided in the corresponding request, while the DATA field specifies the data received by the MAC sublayer.

It indicates the arrival of a valid or invalid frame. Notification of invalid received frame to the LLC sublayer is left to the implementer's choice: in case of invalidity, the DATA parameter is not provided.

c) **Operations associated with this primitive**

The MA-DATA.indication primitive is sent by the MAC sublayer to an LLC entity to indicate the arrival of a data frame without request for acknowledgement. A frame of this type must be indicated in this way when the received frame is valid (correct format, reception without error, see 6.2.6) and when the identifier (IDEN) designates the local MAC entity.

If the MAC entity initiating the request (MA-DATA.request) is itself designated by the IDEN field, the MA-DATA.indication primitive must be called by the MAC entity to the local LLC entity (e.g., all frames sent with a reserved broadcast address cause the MA-DATA.indication primitive to be called up on all modules connected to the medium).

d) **Additional comments**

The codes which can be conveyed by the reception-status parameter are as follows:

— RECEPTION-OK: indicates the arrival of a valid frame,

— RECEPTION-ERROR: can be used to indicate the arrival of an invalid frame.

6.1.3.3.1.3 MA-DATA,confirm

a) **Function**

This primitive has a local meaning. It provides the LLC sublayer with an execution report on the MA-DATA, request primitive by indicating, in particular, the success or failure of transmission (e.g., error detected in transmit mode).

b) **Parameters associated with MA-DATA.confirm primitive**

This primitive contains the parameters:

— MA-DATA.confirm (IDEN, EXT, TRANSMISSION-STATUS).

The TRANSMISSION-STATUS parameter is used to pass status information to the local LLC entity which requested data transfer.

It indicates the success or failure of the corresponding MA-DATA.request.

The list of possible values for the TRANSMISSION-STATUS parameter is given in d) below.

c) **Operations associated with this primitive**

This primitive is generated in response to an MA-DATA.request primitive initiated by the local LLC entity. The LLC entity is responsible for placing the MA-DATA.confirm parameter in relation with the corresponding request. The contents of the data whose transfer was requested by this MA-DATA.request primitive must be preserved by the local LLC entity (at least) until this confirm is returned.

d) **Additional comments**

At the LLC sublayer, sufficient information must be available to associate the confirm with the appropriate request (e.g., the MAC sublayer can process only one request at a time or can process requests on a first-in first-out basis: the association is deduced from the order of response).

The list of possible values for the TRANSMISSION-STATUS parameter is as follows:

— TRANSMISSION-COMPLETED: Indicates that the data frame has been sent validly (see 6.2.6);

— LOSS-OF-ARBITRATION[1]: If the MAC entity does not recover in the event of contention, it thereby indicates to the LLC entity that the medium cannot be accessed due to a collision with a message of higher priority;

— TRANSMISSION-ERROR[2]: This concerns only errors detected in transmit mode: e.g., BIT-ERROR, FORMAT-ERROR or ACK-ERROR (see 6.3.4).

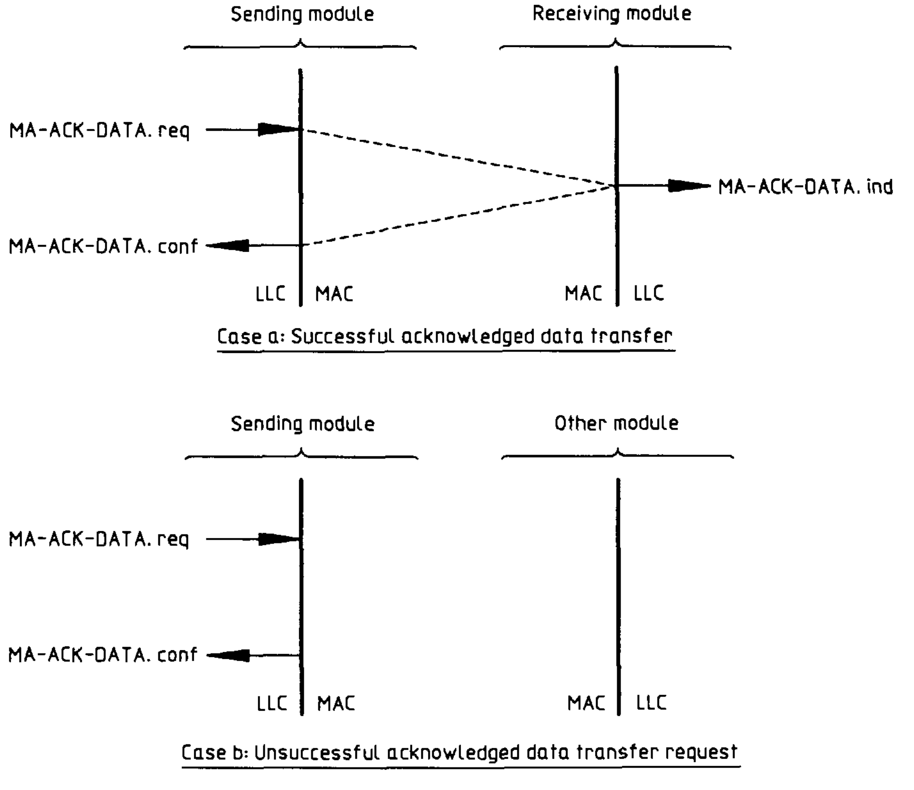


Figure 5 — Example of exchanges of acknowledged data transfer service primitives

1) Recovery on loss of arbitration by the MAC sublayer is optional.

2) In the case of locally detected transmission errors:

CODE-ERROR (code violation),
ACK-ERROR if the acknowledgement field has the value POSITIVE-ACK.

Figure 5 gives examples of exchanges of primitives associated with the acknowledged data transfer service.

6.1.3.3.2 Primitives associated with acknowledged data transfer service (ADT)

6.1.3.3.2.1 MA-ACK-DATA.request

a) **Function**

This primitive allows point to point data transfer with acknowledgement by the remote module receiving the data.

b) **Parameters associated with MA-ACK-DATA.request primitive**

The list of parameters associated with this primitive is as follows:

— MA-ACK-DATA.request (IDEN, EXT, DATA).

The IDEN field can identify the module concerned by the transferred data (DATA). The EXT parameter is an extension field reserved for possible use by the LLC sublayer (one extension bit conveyed in the data frame: see 6.2.2).

The DATA parameter must contain sufficient information to indicate to the MAC sublayer the length of the data to be transferred.

c) **Operations associated with MA-ACK-DATA.request primitive**

This primitive is sent by an LLC entity to the MAC sublayer to activate data transfer in acknowledgement mode to a remote receiving module.

Upon receiving this primitive, the MAC sublayer attempts to send a data frame with explicit acknowledgement request, using the acknowledged data transmission procedures.

6.1.3.3.2.2 MA-ACK-DATA.indication

a) **Function**

This primitive allows the MAC sublayer to indicate acknowledged data transfer to an entity of the LLC sublayer.

b) **Parameters associated with MA-ACK-DATA.indication primitive**

This primitive contains the parameters:

— MA-ACK-DATA.indication (IDEN, EXT, DATA, RECEPTION-STATUS).

The IDEN and EXT parameters are identical to those supplied in the corresponding request. The DATA field specifies the data received by the MAC sublayer.

The RECEPTION-STATUS parameter is used to pass status information up to the LLC sublayer: it indicates the arrival of a valid or invalid frame. Notification of invalid received frame to the LLC sublayer is left to the implementer's choice: in case of invalidity, the DATA parameter is not provided.

c) **Operations associated with this primitive**

The MA-ACK-DATA.indication primitive is passed by the MAC sublayer to an entity of the LLC sublayer to indicate the arrival of an acknowledged data frame. A frame of this type must be indicated in this way when the frame received is valid (e.g., correct format, error-free reception: see 6.2.6) and when the identifier (IDEN) designates the local MAC entity.

d) **Additional comments**

The codes which can be sent in the RECEPTION-STATUS parameter are as follows:

— RECEPTION-OK: indicates the arrival of a data frame with valid data acknowledgement;

— RECEPTION-ERROR: used to indicate the arrival of an invalid frame (the list of errors detected in receive mode is given in 6.3.4).

6.1.3.3.2.3 MA-ACK-DATA.confirm

a) **Function**

This primitive is used to confirm the acknowledged data transfer service.

It can indicate the success or failure of an acknowledged data transfer.

b) **Parameters associated with MA-ACK-DATA.confirm primitive**

MA-ACK-DATA.confirm (IDEN, EXT, TRANSMISSION-STATUS).

The TRANSMISSION-STATUS parameter indicates the success or failure of the corresponding MA-ACK-DATA.request primitive.

The list of possible values for the TRANSMISSION-STATUS parameter is given in d) below.

c) **Operations associated with MA-ACK-DATA.confirm primitive**

This primitive is sent by the MAC sublayer to an entity of the LLC sublayer to indicate the success or failure of a previous request for acknowledged data transfer. The success of the acknowledged data transfer request indicates that the data has been sent correctly and that the remote MAC entity in question has been able to initiate passing of a MAC-ACK-DATA.indication primitive to an entity of the remote-LLC sublayer.

d) **Additional comments**

The user of the MAC service must provide sufficient context information associated with each MA-ACK-DATA.request to be able to correlate the MA-ACK-DATA.confirm primitive with the corresponding request.

The list of possible values of the TRANSMISSION-STATUS parameter is as follows:

— TRANSMISSION-COMPLETED: Indicates that the data frame has been sent successfully and that the acknowledge field has been set to the value "ACK POSITIVE" (see 6.2.6).

— LOSS-OF-ARBITRATION[3]: If the MAC entity does not recover in the event of contention, it thereby indicates to the LLC entity that it is impossible to access the medium due to a collision with a message of higher priority.

— TRANSMISSION-ERROR: These are transmission errors detected locally or through absence of acknowledgement: e.g., BIT-ERROR, FORMAT-ERROR or ACK-ERROR (see 6.3.4).

— ACK-ABSENT: Indicates that the data frame has been sent successfully until the EOD, but that the value of the acknowledge field is not equal to "ACK positive".

6.1.3.3.3 Primitives associated with remote transmission service (RT)

6.1.3.3.3.1 MA-REPLY.request

a) **Function**

The primitive is used to request a remote module to transmit data, i.e. the data at the MAC level must be ready and available for remote access.

b) **Associated parameters**

The primitive includes the parameters:

— MA-REPLY.request (IDEN,EXT).

IDEN is the identifier of the requested data and EXT is an extension bit which can be used by the LLC sublayer.

3) Recovery upon loss of arbitration by the MAC sublayer is optional.

c) **Operation associated with this primitive**

This primitive is sent to the MAC service by a user (entity of the LLC sublayer) to initiate transmission of the data identified by the IDEN field via a remote module which is the owner of this data (producer). Exchange takes place in point-to-point mode with acknowledgement (between the producer module and the request initiator module). In case of multiple consumers, the ACK is delivered by the initiator (see 6.1.2).

Upon receiving this request, the MAC sublayer tries to send a frame requesting remote transmission (request frame), identification (IDEN) and extension (EXT).

Figure 6 shows examples of possible scenarios for primitive exchanges in the remote transmission service.

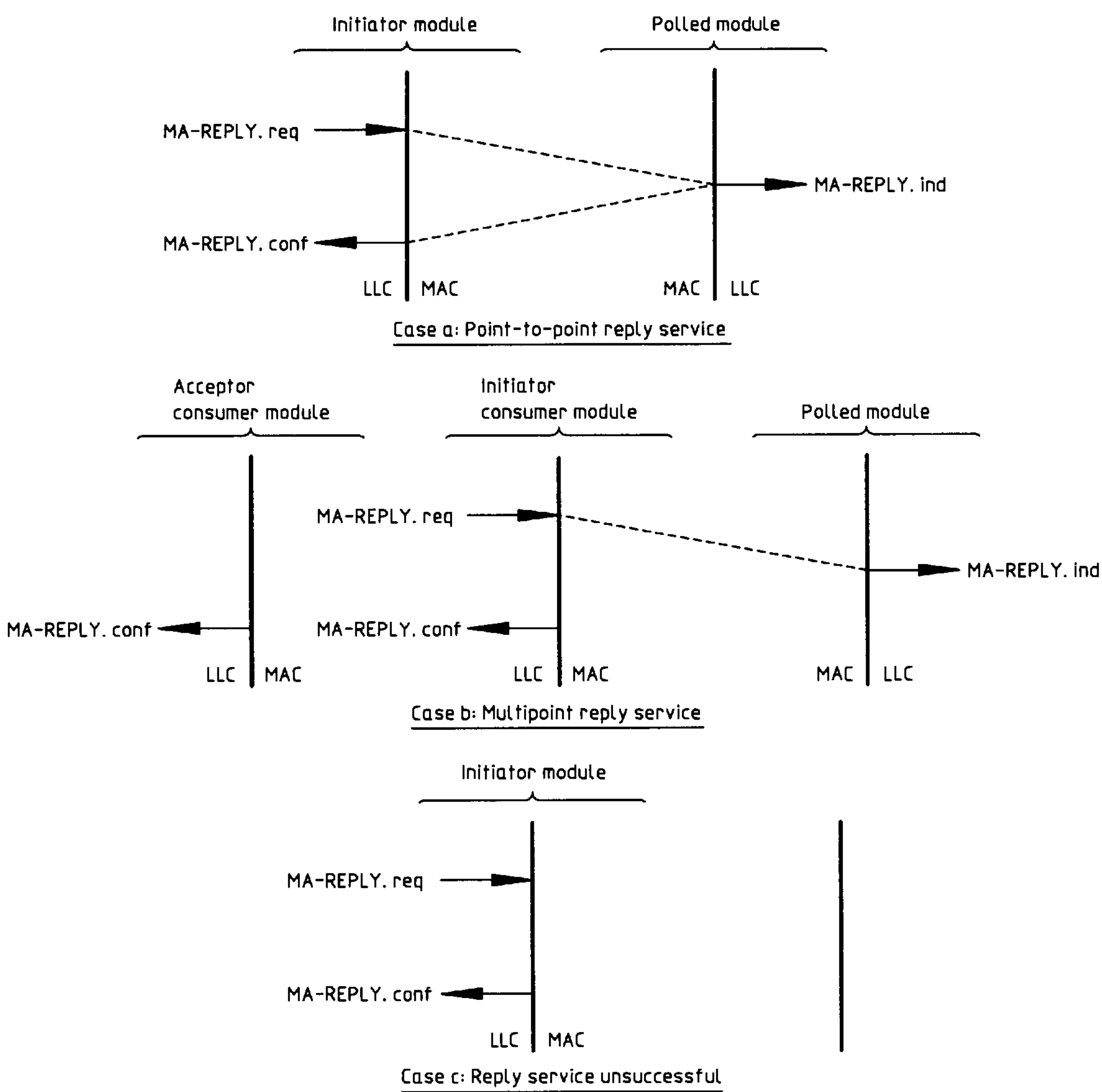


Figure 6 — Example of primitive exchanges associated with remote transmission service

6.1.3.3.3.2 MA-REPLY.indication

a) **Function**

This primitive is the remote transmission service indication primitive.

b) **Associated parameters**

The primitive contains the parameters:

— MA-REPLY.indication (IDEN, EXT, TRANSFER-STATUS).

IDEN is the identifier of the data specified in the corresponding request primitive and EXT is the extension bit value set in the request.

The TRANSFER-STATUS parameter can indicate the result of the operation performed on the polled module: insertion of prepared data or reception of a request frame without response. The values of the specified codes are given in d) below.

c) **Semantics of MA-REPLY.indication primitive**

This primitive is sent by the MAC sublayer to a user receiving the remote transmission request (designated by IDEN) to indicate either the insertion of the requested data (immediate response), or the arrival of a request for transmission of the data identified by IDEN.

d) **Additional comments**

The transfer of prepared data designated by IDEN to a requesting module does not destroy the data possessed by the owner module (polled module). Subsequent requests for remote transmission of this data will cause transfer of the same data, unless an update has been performed in the meantime (see 6.1.3.3.3.4).

The possible values of the TRANSFER-STATUS parameter associated with this primitive are as follows:

— RESPONSE-SENT: The response has been inserted in the frame by the module concerned and the response frame sent is valid (acknowledged by the module initiating the corresponding request);

— RESPONSE-ERROR: An error has been detected during transmission of the response;

— NO-RESPONSE: A valid request frame has been received: the requested data was not ready and the polled module did not immediately send an in-frame response;

— REQUEST-ERROR: An invalid request frame has been received.

6.1.3.3.3.3 MA-REPLY.confirm

a) **Function**

This primitive is the remote transmission service confirm primitive.

b) **Associated parameters**

The primitive contains the parameters:

— MA-REPLY.confirm (IDEN, DATA, TRANSFER-STATUS).

The IDEN parameter designates the data requested by the module which initiated the remote transmission service. The DATA parameter contains the data received by the MAC sublayer. The TRANSFER-STATUS parameter indicates the success or failure of the exchange triggered by the module initiating the remote transmission request. Various status codes are described later [see d)], and one of these codes corresponds to complete success in service execution (requested data received correctly).

c) **Semantics of MA-REPLY.confirm primitive**

This primitive is sent by the MAC sublayer to a user (entity of the LLC sublayer) in the following cases:

— For a service initiator module: to indicate the success or failure of a previous request for execution of the remote transmission service.

— For a transmitted data acceptor module: to indicate the success or failure of a remote transmission service request initiated by another module (notification is given in all cases: success or failure).

d) **Additional comments**

The MAC service user is responsible for providing adequate context information to allow reception of the MA-REPLY.confirm primitive to be correlated to the corresponding request primitive (MA-REPLY.request).

The various possible codes for the TRANSFER-STATUS information are as follows:

— RESPONSE-RECEIVED: The status indicates the complete success of the exchange initiated by the corresponding MA-REPLY.request. It indicates that the initiating module has correctly received the requested data (with acknowledgement set by the request initiator) and that an MA-REPLY.indication primitive has been sent back to the user of the MAC sublayer via the proprietary data module.

— RESPONSE NOT READY: This status indicates that the module which is the owner of the requested data has correctly received the request frame and that an MA-REPLY.indication has reached the user of the receiving MAC sublayer but that no response has been given by the remote MAC sublayer (request frame with acknowledgement by the module owning the requested data).

— ACK-ABSENT: This status indicates failure to receive the request frame or an impossibility of acknowledgement at the level of the data owner module.

— TRANSMISSION-ERROR: This status indicates a transmission error detected locally by the module initiating the MA-REPLY.request request.

— LOSS OF ARBITRATION[4]: When the MAC entity does not recover upon contention, it thereby indicates the impossibility of accessing the medium due to collision with a message of higher priority.

6.1.3.3.3.4 MA-REPLY-UPDATE.request

This primitive allows the user of the MAC sublayer to prepare beforehand the data to be transferred in the event of a remote transmission request.

This primitive contains the parameters:

— MA-REPLY-UPDATE.request (IDEN, EXT, DATA).

Reception of this primitive by the MAC sublayer enables the MAC sublayer to prepare data which will subsequently be accessed remotely.

Figure 4 shows an exchange of interface primitives associated with preparation of remote-accessed data.

6.1.3.3.3.5 MA-REPLY-UPDATE.confirm

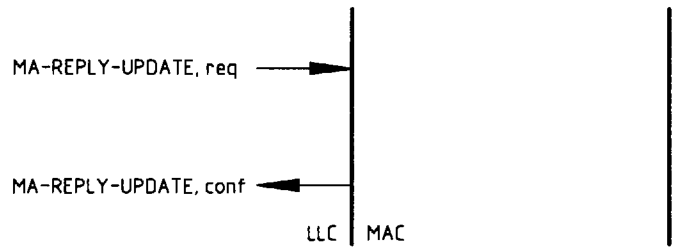
This primitive is a local confirm of execution of a reply preparation request.

It contains the parameters:

— MA-REPLY-UPDATE.confirm (IDEN, STATUS).

The STATUS parameter indicates the success or failure of the previous data preparation request.

4) Recovery upon loss of arbitration by the MAC sublayer is optional.



Case a: Point-to-point remote transmission service

Figure 7 — Exchange of response preparation service primitives

6.2 Structure of MAC frames

6.2.1 Object

This subclause describes in detail the frame structure defined for data transmission between systems using the MAC procedures of the Vehicle Area Network (VAN) type.

Three types of frame are used in MAC procedures:

— **Data frame** (with acknowledgement field): This frame corresponds to a frame sent over the data transmission line which incorporates a specific field (ACK) allowing the receiver to indicate correct reception of the frame sent.

— **Request frame**: This frame, which is initiated by an autonomous module, is sent without data over the data transmission line to access remote data.

— **Response frame**: This frame contains the data requested by a remote module, where

immediate response converts a request frame into a response frame. In this case, the polled module inserts its own data, when available, in the frame;

in the event that the data is not available, the polled module can send a response frame later (store-and-forward).

All the types of frames used by the MAC procedures described in this part of ISO 11519 are structured according to the frame format described in 6.2.2.

6.2.2 Format of MAC frames (encapsulation/decapsulation)

The MAC level frame format is described in figure 10. It consists basically of four fields which are managed and formatted by the MAC sublayer, and the optional acknowledgement field (ACK).

The four fields fully formatted by the MAC sublayer are:

— the frame Identifier (IDEN) which is used to designate the object sent in the frame or the object queried; this field is provided by LLC sublayer;

— the Command field (COM) which specifies the type of frame sent;

— the field containing the LLC-level data to be sent (DAT);

— the Frame Check Sequence (FCS) field, which contains a Cyclic Redundancy Check (CRC) value allowing errors to be detected in received frames.

This format is completed at the physical level by adding Start Of Frame (SOF), End Of Frame (EOF) and End Of Data (EOD) fields before transmission over the bus (figure 8). These fields are encoded and inserted by the physical layer at the request of the MAC sublayer: their contents (encoded symbols) and their length are specified in 7.2.3. Here we shall simply specify their function.

All the MAC level fields are of fixed length except for the LLC data field (DAT).

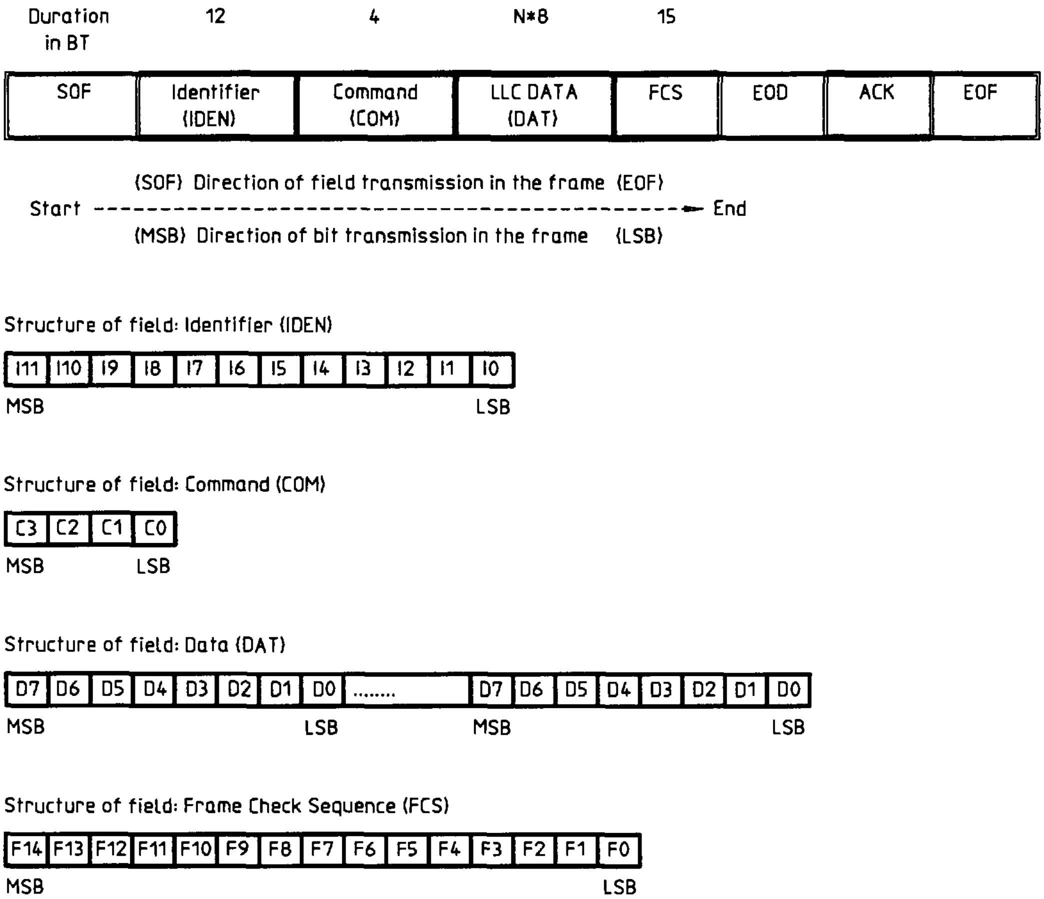
The data field (DAT), which contains the data of the MAC sublayer user, must contain a whole number of bytes (n) ranging between a minimum value and a maximum value specified by each implementation of the MAC access method in this document. Recommended values are specified in 6.3.6.

The Identifier (IDEN), Command (COM), LLC data (DAT) and Frame Check Sequence (FCS) fields consist of binary sequences (strings of bits on 0 or 1) the contents of which are specified below.

The representation conventions in figure 8 are as follows:

— The fields of the frame are sent from left to right and, within each field, the bits are encoded and sent from the Most Significant Bits (MSB) to the least significant bits (LSB).

— The lengths of binary fields (IDEN, COM, DAT, FCS) are expressed in number of bit times (BT = time for transmission of one binary element).



Key

Fields surrounded by a double line are encoded and summed by the physical layer.
MSB = Most Significant Bit(s).
LSB = Least Significant Bit(s).
BT = Time for transmission of one bit.

Figure 8 — Format of MAC frame

6.2.3 Description of MAC frame elements

6.2.3.1 Start Of Frame field (SOF)

The Start Of Frame (SOF) field marks the start of a frame. It is also used at the physical layer level to synchronize the receiver(s) on the received frame.

Its contents are defined in 7.2.4.

6.2.3.2 Identifier field (IDEN)

The Identifier field of the frame is a 12-bit field used to specify the frame's destination address. The address specifies the module(s) for which the frame is destined.

The Identifier field is a random sequence of binary elements (bits).

The identifier is provided by the LLC sublayer.

Two addresses are reserved for specific needs (000 and FFFH).

6.2.3.3 Command field (COM)

The command field contains four command bits whose meaning is as follows (see figure 9):

— One extension bit (EXT): this bit is reserved for subsequent use for the Link layer. In this part of ISO 11519, the extension bit is set to 1 (recessive).

— One request acknowledge bit (RAK): this bit determines whether the sending module requests or not that a receiver module acknowledge successful reception of the frame.

If the RAK bit is on 1, the sending module requests an in-frame acknowledgement: the acknowledgement field (ACK) can be set by a receiver module (see 6.2.3.7) to the value corresponding to a positive acknowledge (POSITIVE ACK).

If the bit is on 0, the ACK field must be set to the value corresponding to non-acknowledgement (see figure 39).

— One read/write bit (R/W) which indicates whether the frame sent is a read or write request; the values of the R/W bit are specified in the tables in figure 9.

— A remote transmission request (RTR).

If the bit is on 0, the frame sent contains LLC-level data (DAT field, see 6.2.3.4).

If the bit is on 1, the frame contains no LLC data and is interpreted by the receiver module(s) as a request to send.

6.2.3.4 Data field (DAT)

The data field consists of a sequence having a whole number of bytes. The contents of this field are a completely arbitrary sequence of bytes provided by the LLC sublayer. The length of this field ranges between a minimum value and a maximum value which are specified by each implementation. Recommended values for these two boundary values are specified in 6.3.6 so as to ensure compatibility of implementation (flexibility).

The MAC sublayer provides transparent LLC-level data transmission (in the sense that the data transmitted consist of random byte strings).

EXT	RAK	R/W	RTR
C3	C2	C1	C0

Field value in Manchester-L encoding

EXT	R	
RAK	R	Acknowledgement requested
	D	No acknowledgement
R/W	D	Read
	R	Write
RTR	D	Data transmission
	R	Remote transmission request

Field value in Enhanced Manchester encoding

EXT	R	
RAK	R	Acknowledgement requested
	D	No acknowledgement
R/W	R	Read
	D	Write
RTR	D	Data transmission
	R	Remote transmission request

Figure 9 — Command field (COM) — General structure

6.2.3.5 Frame Check Sequence field (FCS)

A Cyclic Redundancy Check (CRC) is used in the transmit and receive procedures of the MAC sublayer to detect any individual errors or packet errors in the messages transmitted.

The Frame Check Sequence (FCS) field is a 15-bit field containing the check bits (see figure 10).

The encoding is defined by the following generator polynomial $g(x)$ of order 15:

$$g(x) = x^{15} + x^{11} + x^{10} + x^{9} + x^{8} + x^{7} + x^{4} + x^{3} + x^{2} + 1$$

The CRC calculation concerns the identification, command and LLC data (useful information block) fields.

F14	F13	F12	F11	F10	F9	F8	F7	F6	F5	F4	F3	F2	F1	F0
x^{14}	x^{13}	x^{12}	x^{11}	x^{10}	x^9	x^8	x^7	x^6	x^5	x^4	x^3	x^2	x^1	x^0

Figure 10 — Field: Frame Check Sequence (FCS)

Encoding and check process

Taken as a whole, the identification bits, command bits and data bits numerically correspond to the coefficients of a polynominal having terms ranging from x^{n-1} (where n = total number of bits in a message) to x^{15}, in decreasing order.

This polynomial is subjected to modulo 2 division by the generator polynomial $g(x)$.

The r check bits correspond numerically to the coefficients of terms ranging from x^{14} to x^0 of the polynomial obtained as the remainder of this division operation.

The complete data block corresponds numerically to the coefficients of a polynomial which can be divided by the generator polynomial by the modulo 2 process.

Calculation of CRC in transmit mode

Figure 11 shows a shift-register encoding system. The register's flip-flops are initialized at 1.

Execution of the operation:

— Gates A and B are enabled, gate C is disabled and the k data bits ($n - r = k \rightarrow k$ clock periods), with the most significant bit leading, are counted and entered. They appear simultaneously at output (transmission channel).

— After the data bits have been entered, gates A and B are disabled, gate C is enabled and the register is subjected to 15 new clock periods. During this count, the appropriate check bits appear in inverted form (ones-complemented) in succession at output.

In transmit mode, the data bits are therefore subjected to an encoding process which is equivalent to division by the generator polynomial. The remainder obtained (ones-completed) is sent over the line immediately after the data bits, in decreasing order of terms.

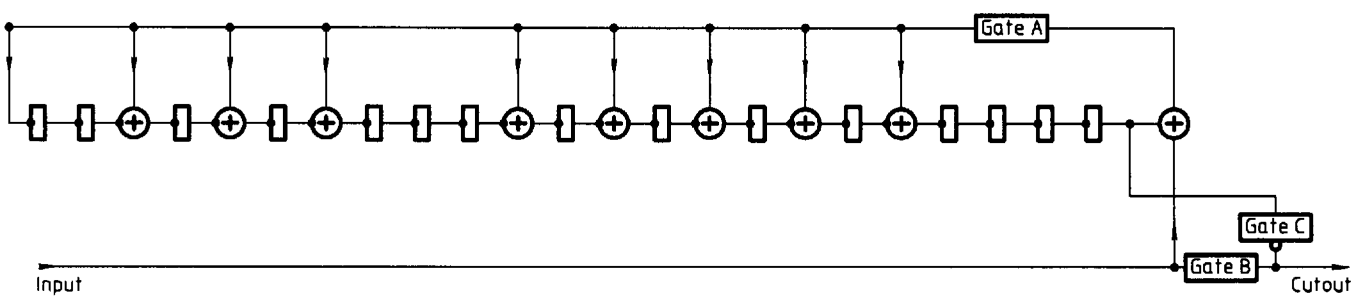


Figure 11 — Block diagram of Frame Check Sequence (FCS) encoder

Calculation of CRC in receive mode

Figure 12 shows a shift-register decoding system.

Execution of the operation:

— Initialization: the flip-flops are set to 1.

— Gates A, D and E are enabled. The k data bits are then counted and entered (most significant bit leading). They appear simultaneously at output (at the MAC-LLC interface level).

— After the k counts, gate E is disabled and the 15 check bits are then counted and entered, and the contents of the register stages are examined. These contents are a fixed remainder equal to 100101100010101 (in order of decreasing powers), if no error occurs. Contents other than this fixed remainder indicate that the received block is erroneous.

At reception, the data bits are therefore subjected to a decoding process which is equivalent to division by the generator polynomial.

If there is no error, this division results in a fixed remainder as defined above, non-null.

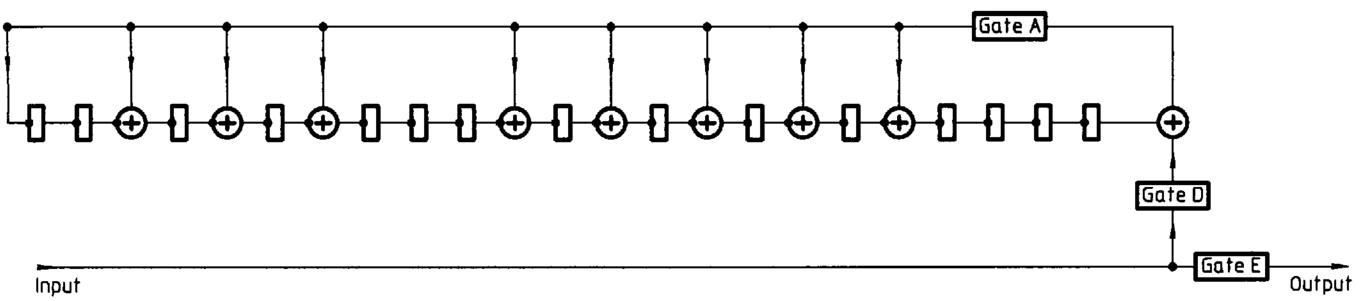


Figure 12 — Block diagram of Frame Check Sequence (FCS) decoder

6.2.3.6 End Of Data (EOD) field

This field is requested by the MAC sublayer and generated by the physical layer.

The End Of Data (EOD) field marks the end of the fields transmitted by the sender, i.e. the portion of the frame including the identifier, command, LLC data and Frame Check Sequence (FCS) fields. It therefore determines the length of the LLC data sent in the frame. The contents of this field are defined at the physical level in 7.2.4.

6.2.3.7 Acknowledgement (ACK) field

The acknowledgement (ACK) field is reserved. It consists of information the contents of which are specified in the physical layer (see 7.2.4).

6.2.3.8 End Of Frame (EOF) field

This field is requested by the MAC sublayer and generated by the physical layer.

The End Of Frame (EOF) field marks the end of the frame and is represented by a symbol of the code defined at the physical level (see 7.2.4).

6.2.4 Types of MAC-level frame

This subclause specifies the different types of frames used in the MAC sublayer procedures.

6.2.4.1 Data frame (with or without acknowledgement request)

The structure of a data frame is shown in figures 13, 14, 15 and 16. The frame-sending module generates the following MAC-level fields: frame identifier, command, data and FCS fields. The R/W bit encoding corresponds to writing (see figure 9).

In the event of acknowledgement, the receiving module(s) set, on the fly, the acknowledgement field to ACK POSITIVE.

6.2.4.2 Request/response frames

The request (or remote transmission request) frame is characterized by an LLC data field of zero length and by a command field for which the R/W and RTR bits have the values specified below:

— The value of R/W designates a read mode (see figure 9);

— The value of RTR is that specified for a remote transmission request (see figure 9).

NOTE 1 For synchronization reasons between the polling module and the in-frame responding module, the value of the bit preceding the RTR bit must be set to the value defined in figure 9.

In the event of an in-frame response, the initiator sends the frame part including identifier and command fields (see figures 17, 18, 23 and 24).

The response part of frame generated by the polled module contains the following fields (see figures 17, 18, 23 and 24):

— A dominant RTR bit whose value is superimposed on the recessive RTR bit of the request;

— A data field (DAT);

— A Frame Check Sequence field (FCS).

Superposition of the two parts of the frame (request and immediate in-frame response) gives a frame on the bus which has a structure in compliance with the general frame structure (see figures 17, 18, 23 and 24).

6.2.5 Bit serializing/deserializing

Within a frame, for the portion included between the identifier field and the End Of Data (EOD) field, frame transmission takes place byte by byte, starting from the left-hand bytes (see figure 25) and within a byte starting from the Most Significant Bits (MSB).

6.2.6 Frame test (validity)

A frame is considered VALID at the MAC sublayer level when all the following conditions are verified:

— The required frame format is complied with, from the SOF character to the EOF character inclusive;

— The portion of the frame between the Identifier field and the Frame Check field (including FCS) corresponds to a length ranging between the minimum and maximum lengths specified for each implementation of the document (for the recommended values, see 6.3.6);

— The identifier, command, LLC data and Frame Check Sequence (FCS) fields consist of an arbitrary string of 0 or 1 bits (no code violation);

— The LLC data field contains a whole number of bytes;

— The bits making up the received frame (not including the FCS field itself) generate a CRC vlaue identical to that received;

— For those frames for which acknowledgement is requested, the acknowledgement must be positive;

— For those frames for which acknowledgement is not requested, the acknowledgement field must be set to "ACK absent".

Any frame not complying with the above criteria is considered INVALID.

The occurrence of an invalid received frame may be signalled to the network administration, whatever type of error was detected.

Notification of an invalid received frame to the LLC sublayer is an implementation choice: if done, notification must not be given when the received frame is too short (identifier and command fields are not complete) or when the frame is not selected (see identifier management in 6.3.2).

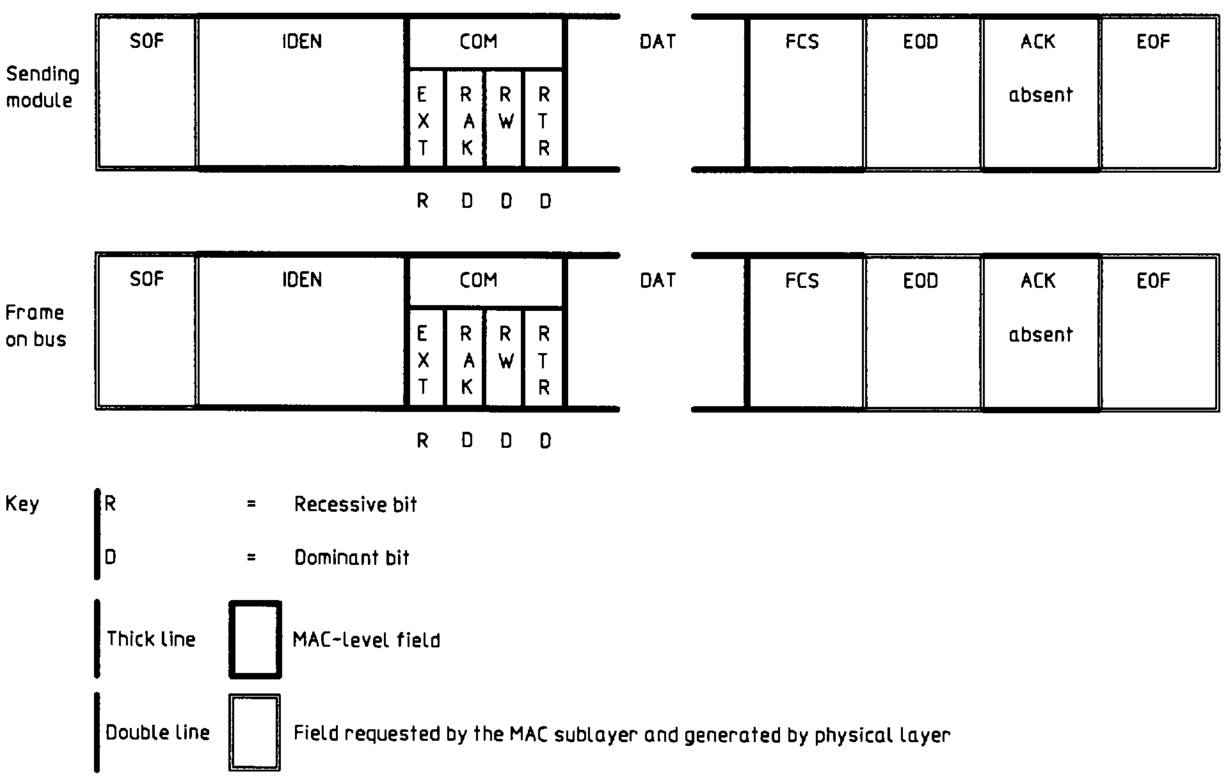


Figure 13 — Structure of data frames without request for acknowledgement — Enhanced Manchester

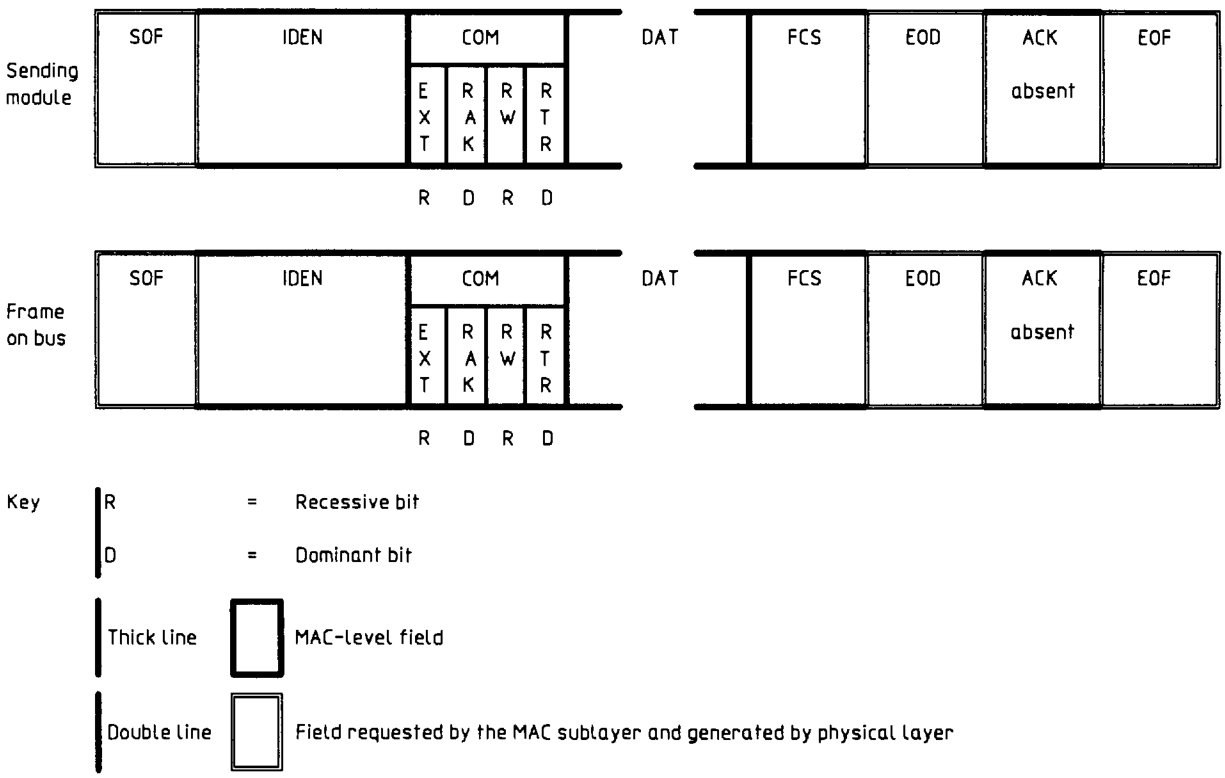


Figure 14 — Structure of data frames without request for acknowledgement — Manchester-L

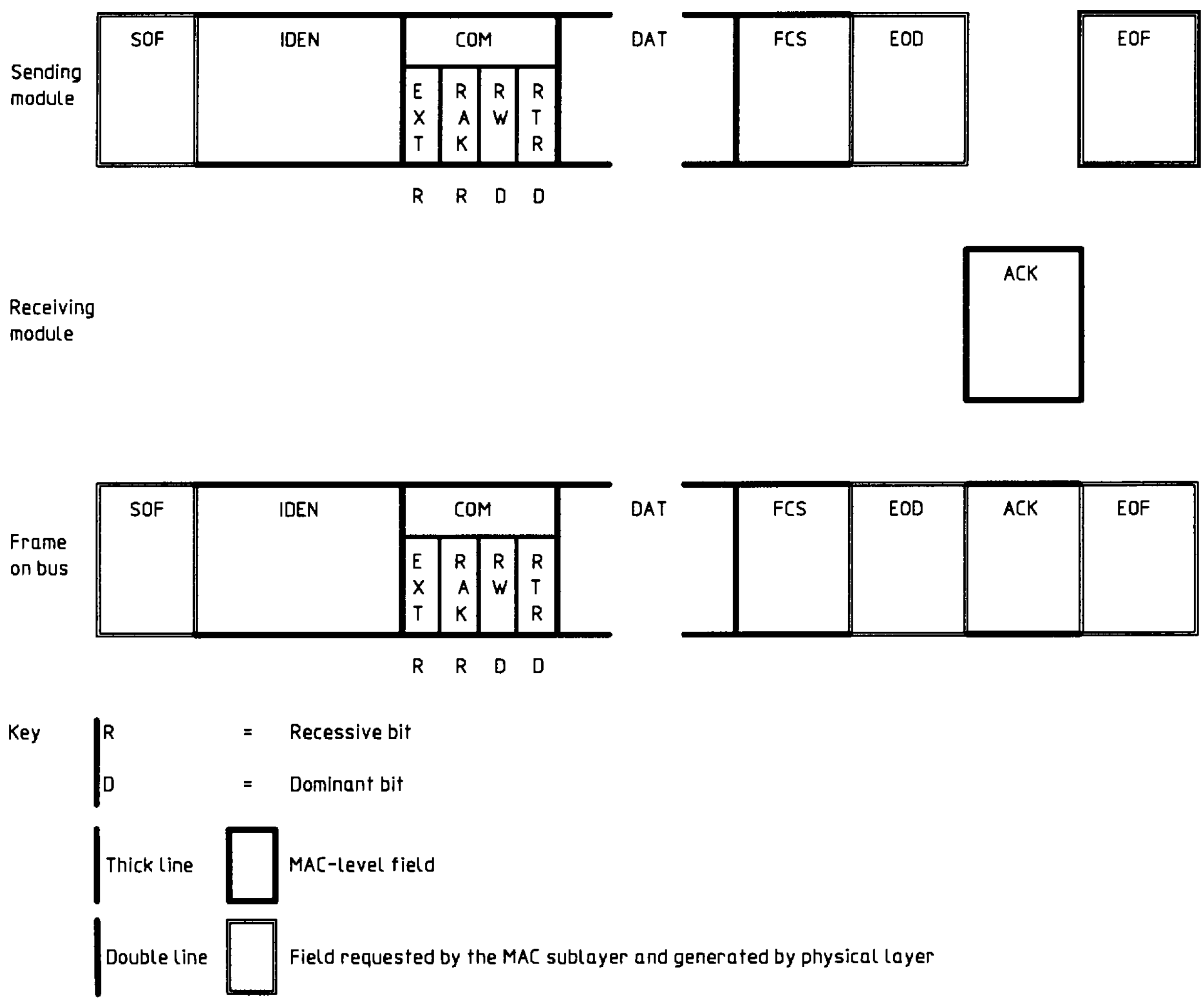


Figure 15 — Structure of data frames with request for acknowledgement — Enhanced Manchester

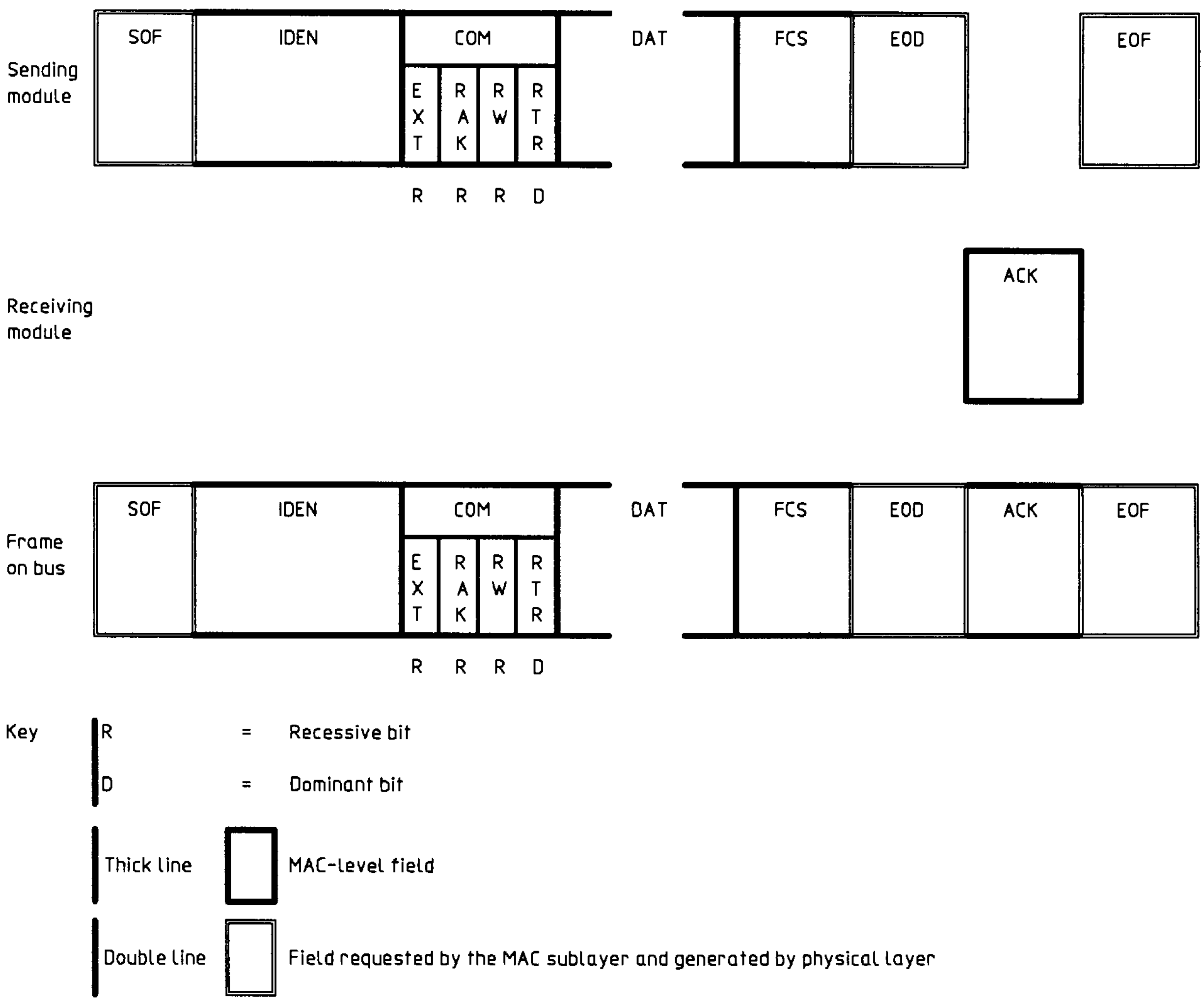


Figure 16 — Structure of data frames with request for acknowledgement — Manchester-L

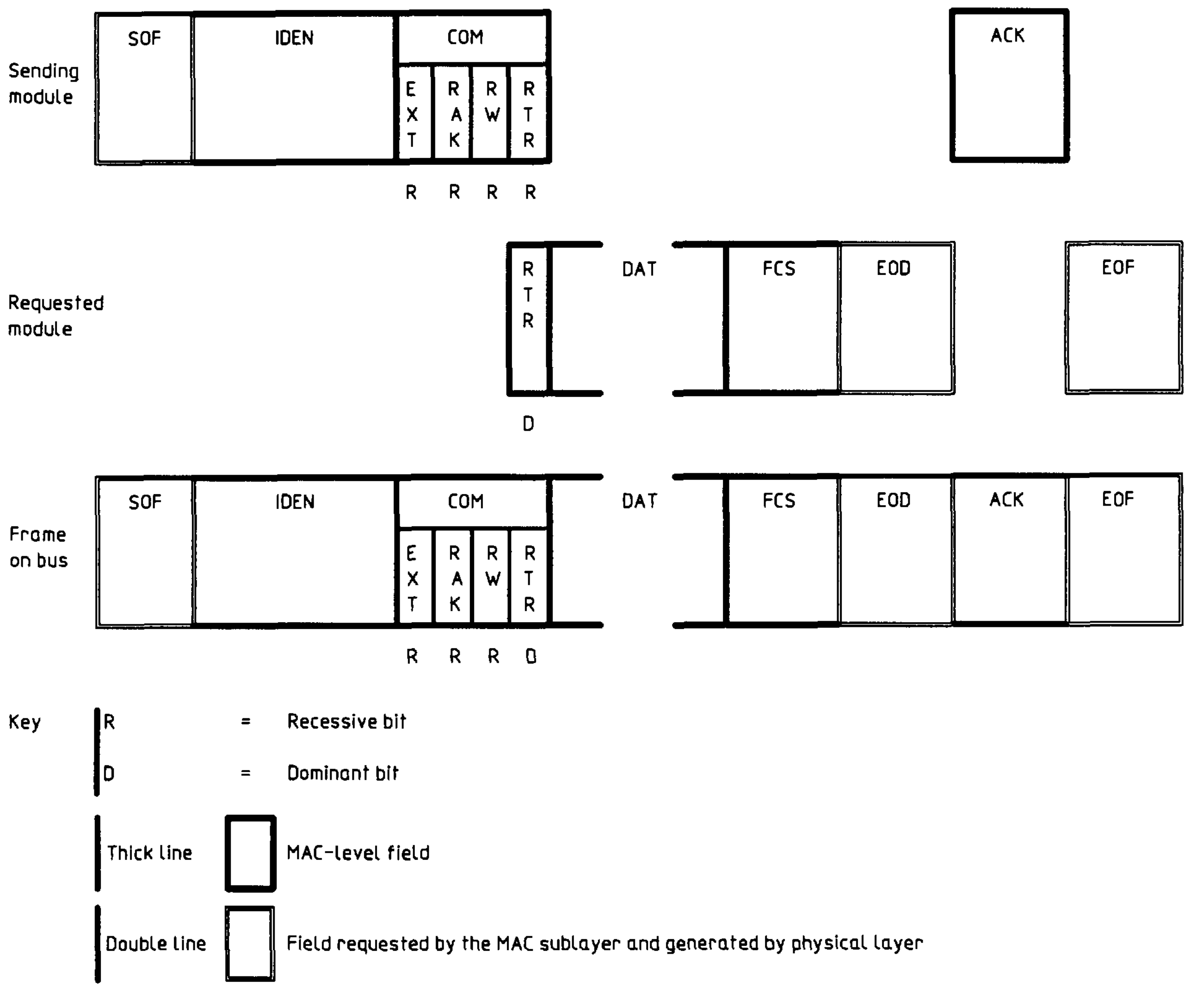


Figure 17 — Structure of remote transmission request and immediate response frames with acknowledgement — Enhanced Manchester

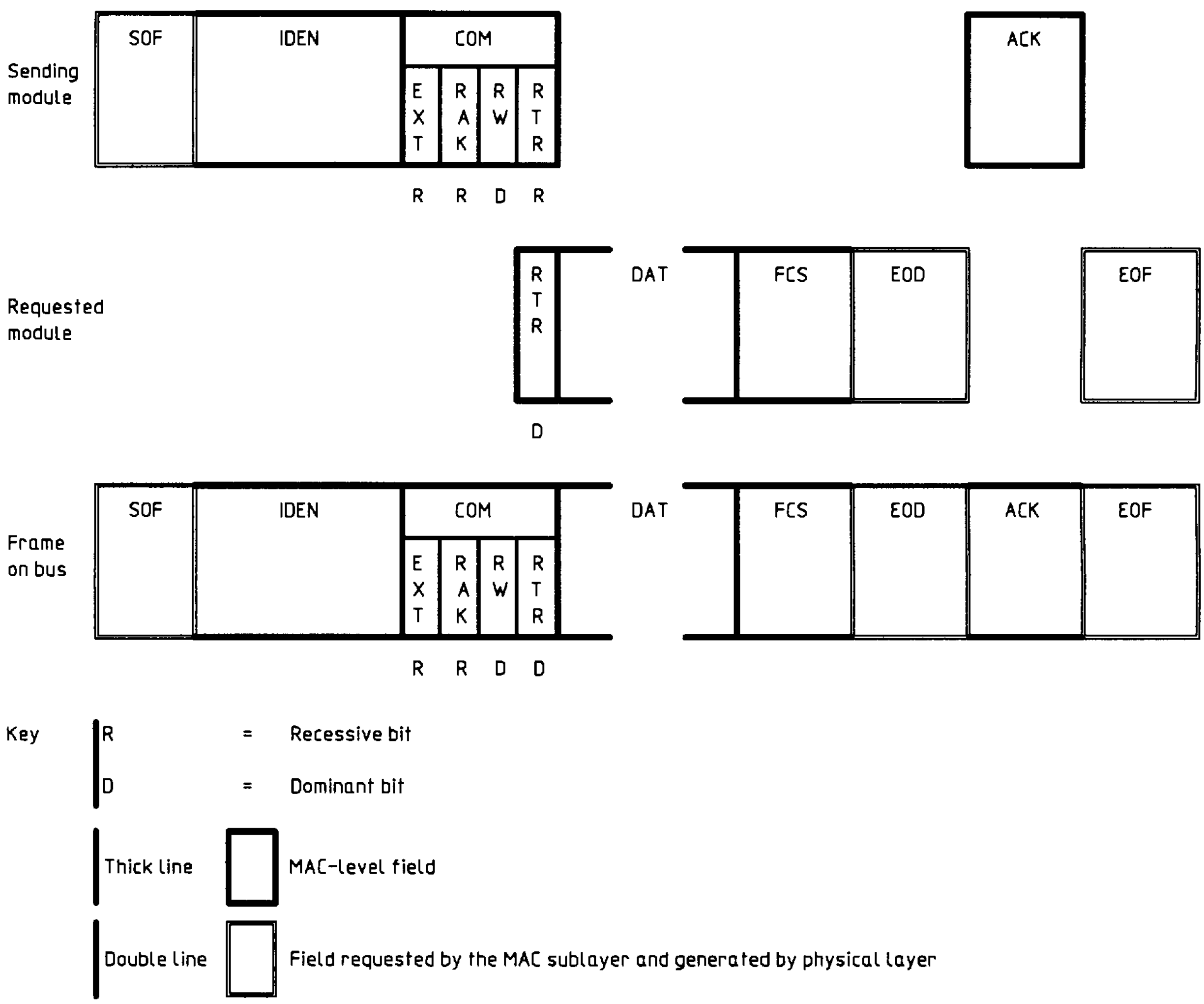


Figure 18 — Structure of remote transmission request and immediate response frames with acknowledgement — Manchester-L

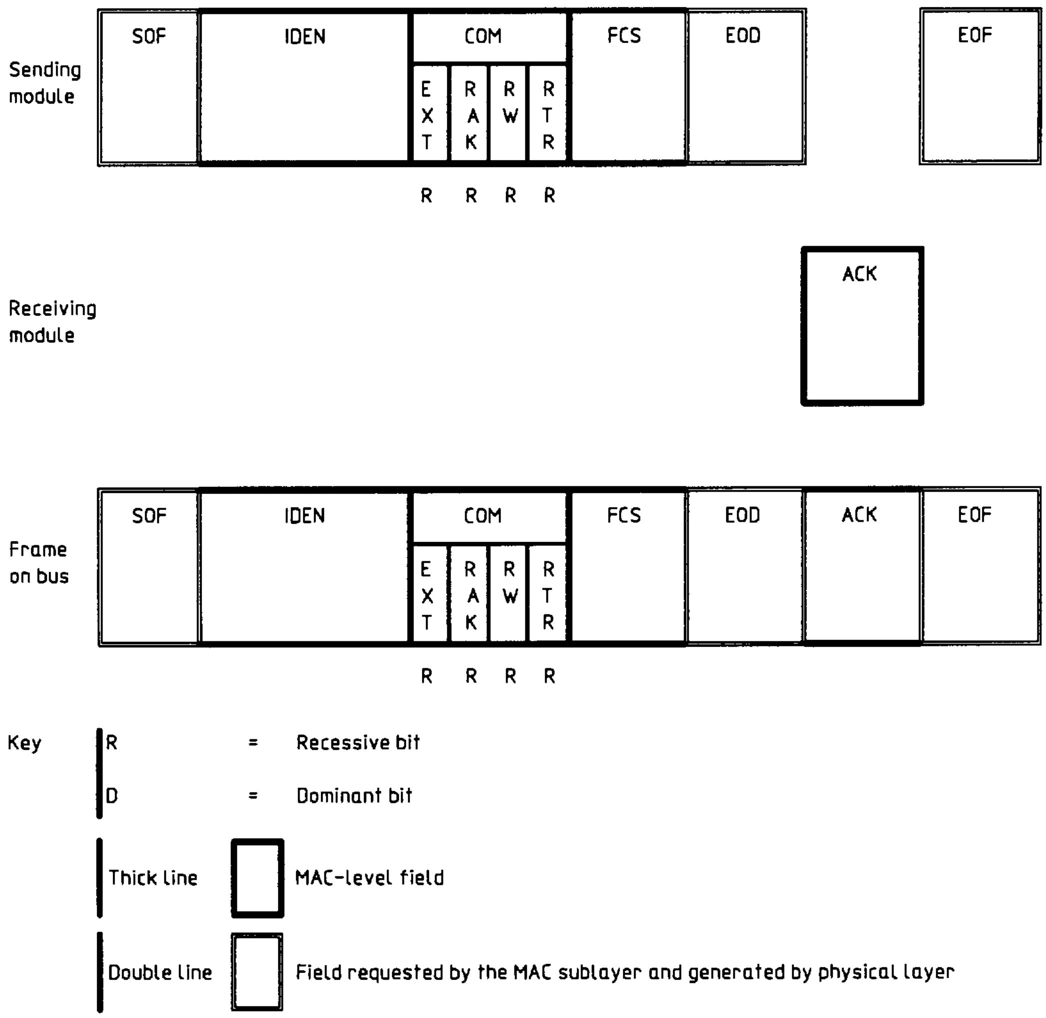


Figure 19 — Structure of remote transmission request and without in-frame response with acknowledgement — Enhanced Manchester

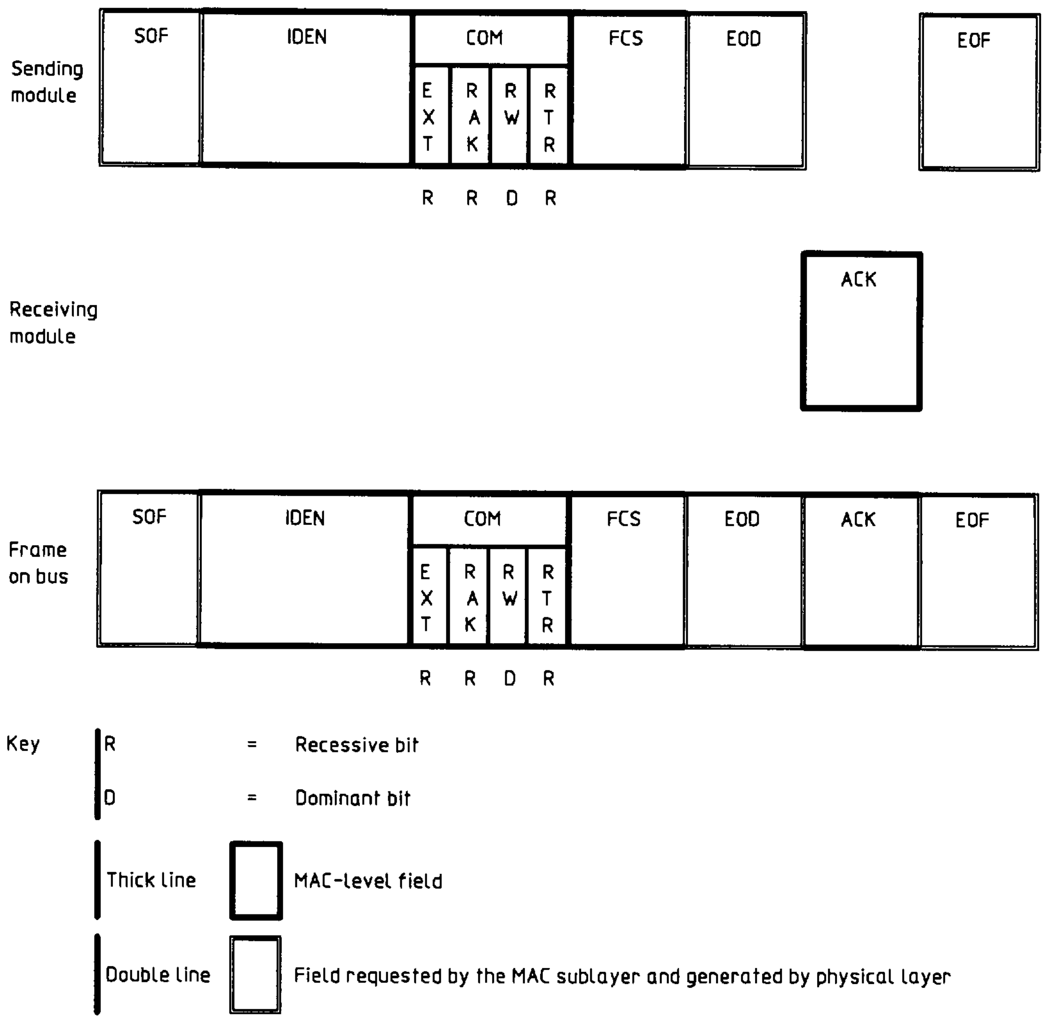


Figure 20 — Structure of remote transmission request and without in-frame response with acknowledgement — Manchester-L

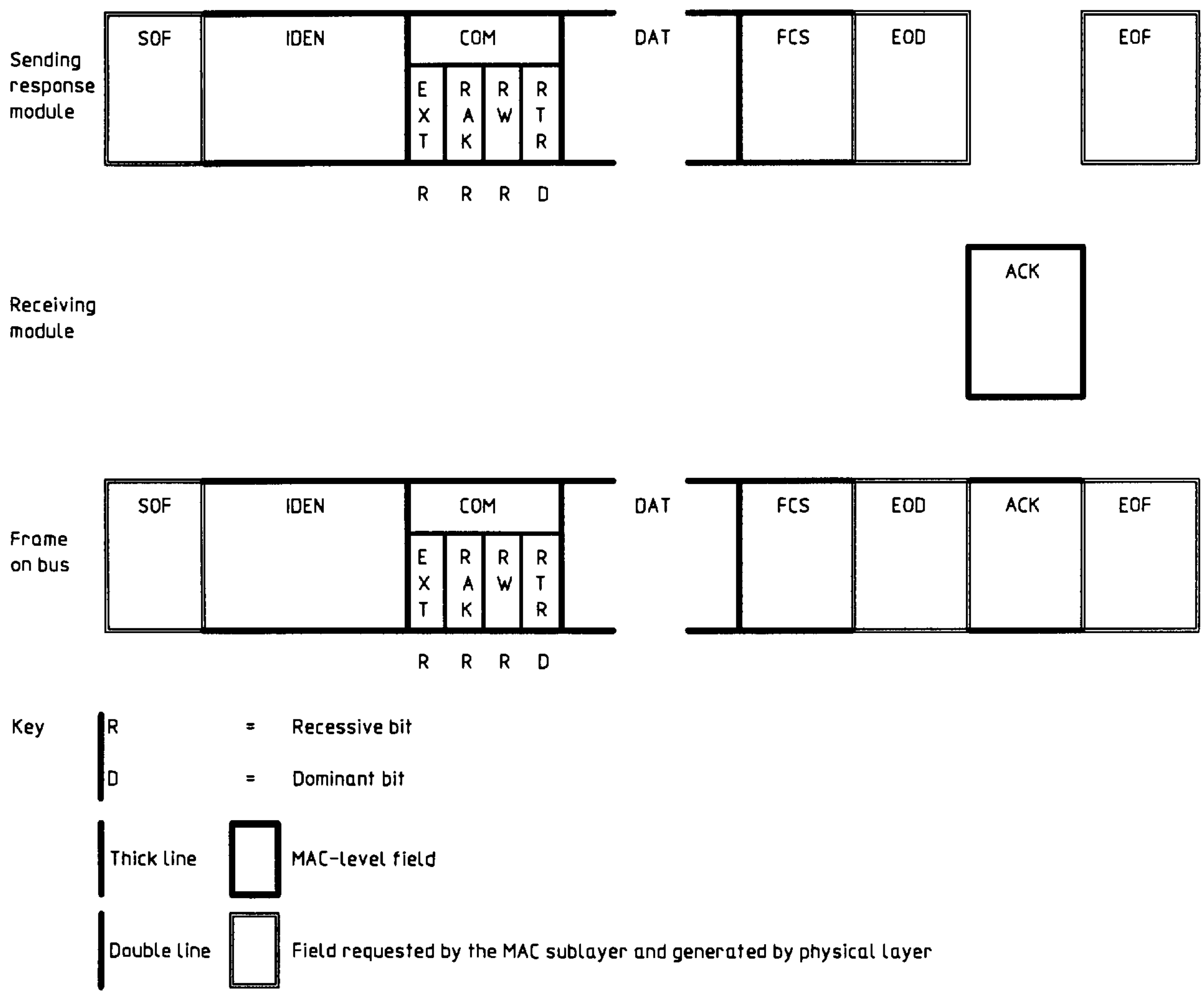


Figure 21 — Structure of deferred response frames with acknowledgement — Enhanced Manchester

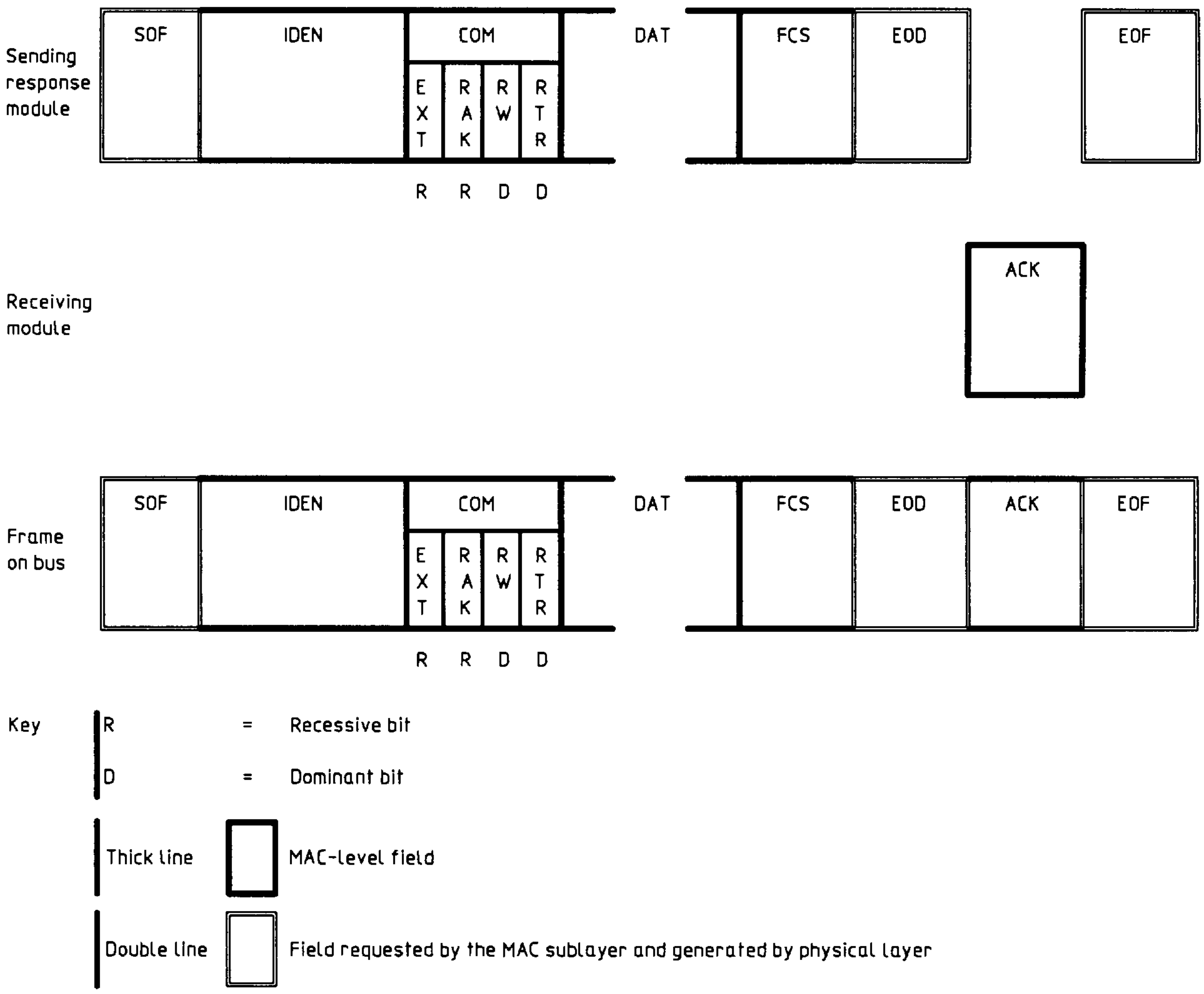


Figure 22 — Structure of deferred response frames with acknowledgement — Manchester-L

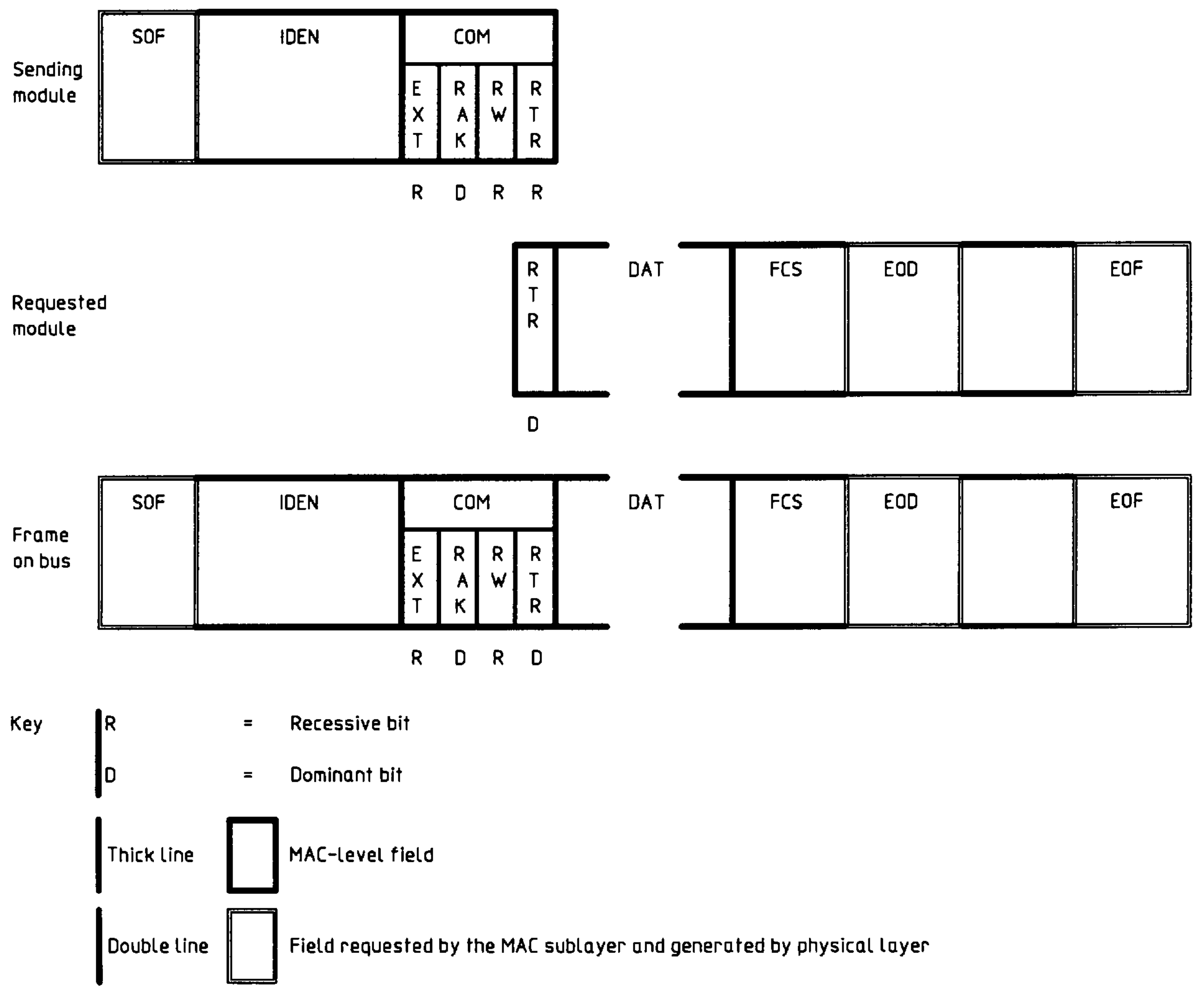


Figure 23 — Structure of remote transmission request and immediate response frames without acknowledgement — Enhanced Manchester

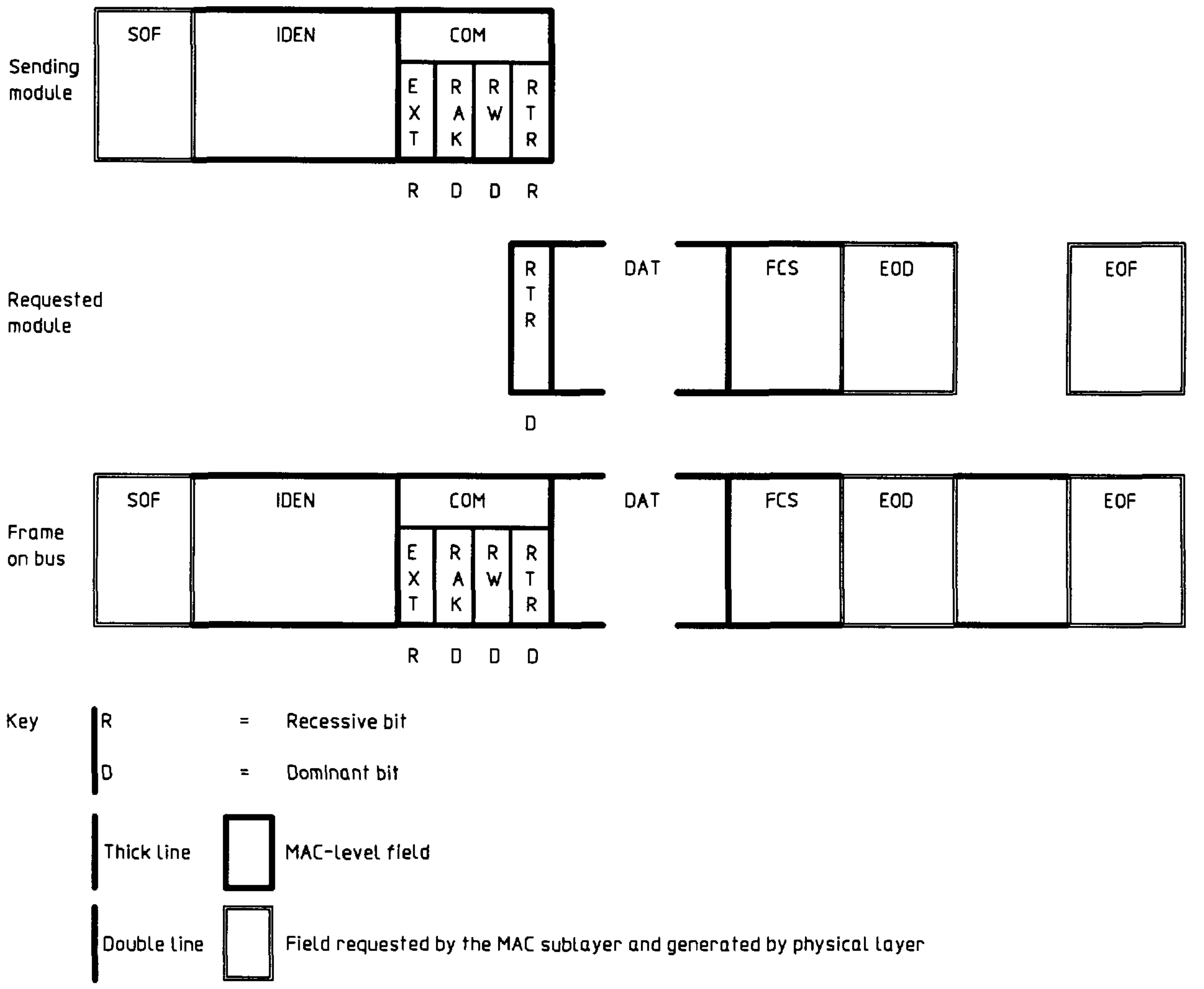


Figure 24 — Structure of remote transmission request and immediate response frames without acknowledgement — Manchester-L

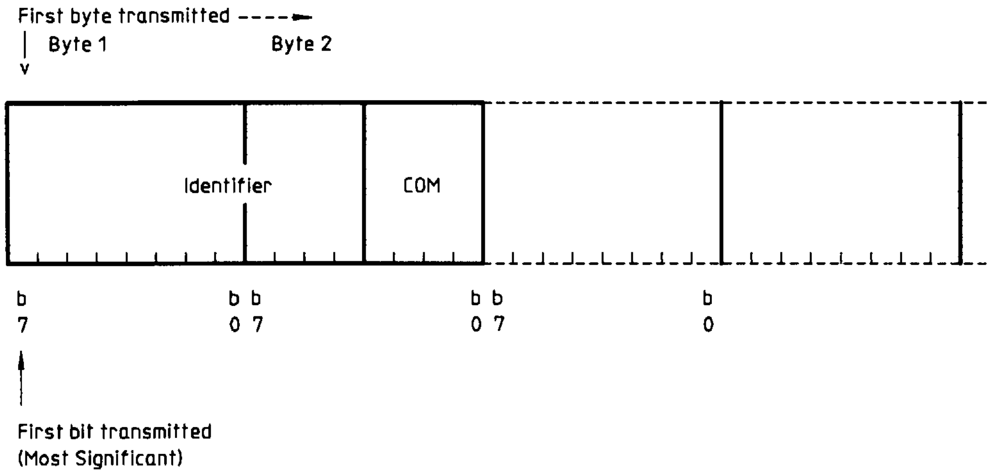


Figure 25 — Principle of frame transmission

6.3 Specification of Medium Access Method (MAC)

6.3.1 Object

This subclause describes the functionalities implemented in the MAC sublayer to provide facilities for sharing the common bus (data transmission medium) between the interconnected modules in a vehicle area network.

In accordance with the architectural model in figure 1, the main functions performed by the MAC sublayer are

a) data encapsulation and decapsulation:

— frame structuring,

— management of logical links (identifiers and associated buffers): selection of received frames which either contain data to be consumed or request for remote transmission (see 6.3.2),

— error detection (errors occurring during in transmission over the physical medium);

b) management of the medium access method:

— medium allocation: prior listening to avoid collisions and detection of interference and forced setting of one of the transmitters in the event of a contention;

— conflict solving.

The MAC sublayer is based on the physical layer which provides it with an interface for serial bit transmission over the physical medium. The characteristics of this interface are described in detail in 7.2.2 and illustrated by a set-up example in annex B.

This sublayer provides the LLC sublayer with a medium access service independent of the methodology used for sharing the communication bus.

The breakdown of standard functions of the data link layer between the MAC and LLC sublayers should therefore allow a greater variety of access methods to be supported (flexibility).

6.3.2 Management of identifiers

The MAC sublayer needs for its operation to access two tables which are called respectively consumer table and producer table. These tables are defined below.

6.3.2.1 Producer table

6.3.2.1.1 Definition

This table is used by the MAC sublayer to perform the functions related to frame transmission (data production).

6.3.2.1.2 Function

The producer table allows the MAC sublayer to recognize that it is the producer of some data which are requested remotely and thus to transmit these data to the physical layer for immediate response transmission.

6.3.2.1.3 Contents

The producer table contains elements which are composed of two fields:

a) IDENTIFIER: this field allows the MAC sublayer to recognize that it is the producer of data named by this identifier;

b) DATA: this field is used by the MAC sublayer

   — to send the data named by the associated identifier when requested by a remote module,

   — to store the previously prepared data, which are passed by the LLC sublayer.

The IDENTIFIER field is accessed on read-only mode by the MAC sublayer, while the DATA field can be read and written (updated).

6.3.2.2 Consumer table

6.3.2.2.1 Definition

This table is used by the MAC sublayer to perform the functions related to the frame reception (data consumption).

6.3.2.2.2 Function

The consumer table allows the MAC sublayer to recognize that a received frame is relevant for this module and to transmit the data to the LLC sublayer.

6.3.2.2.3 Contents

The consumer table contains elements which are composed of a unique field:

— IDENTIFIER: this field allows the MAC sublayer to recognize that it is the consumer of data named by this identifier.

This table cannot be modified by the MAC sublayer: an element of this table (identifier field) is accessed on a read-only mode.

6.3.3 Principle of Medium Access Control (MAC)

6.3.3.1 Access control in multiple-master/slave mode

The MAC method proposed in this document provides a module operating method for accessing the data transmission bus by a multiple-master/slave technique.

A slave module can access the bus only when it has been enabled to transmit. Polling of the slave modules is performed at any time by any one of the autonomous modules. An autonomous module is always enabled to transmit.

At any given time, if the bus is available, several modules enabled to transmit may start transmitting simultaneously. Management of competing accesses between these modules is performed by a multiple random access method. In the event of contention, this method allows non-destructive collision, enabling forced setting of one of the conflicting modules (bitwise arbitration).

This method allows one frame to be transmitted at each bus access attempt even in case of contention (in the absence of error).

6.3.3.2 Collision avoidance

A module enabled to transmit may attempt to access the bus at any time. The following procedure is implemented for bus acquisition by a module; the objective is to avoid collision: the module listens beforehand to check that the bus is not busy.

If the bus is busy, it postpones its attempt. As soon as the bus becomes free again, access to the bus is again enabled after a time-out known as interframe spacing (during which the bus must remain free).

6.3.3.3 Contention

A contention situation arises when several modules enabled to transmit try to access the bus simultaneously.

A sending module keeps listening to the bus while it transmits. In this way, it can detect a conflict when interference on the bus occurs (detected signal different from that transmitted by it). The conflict is solved by bitwise arbitration on the part of the frame known as the arbitration area.

6.3.3.4 Interference detection

Interference (or collision) detection on the bus is based, at the physical layer level, on the use of two types of representation (values) of the bus state called dominant state and recessive state.

When several dominant and recessive states are transmitted simultaneously, the resultant state on the bus is dominant. As soon as a transmitting station detects an interference at the physical level (state on bus different from state transmitted), it immediately ceases transmitting. Priority is thus granted to modules sending a dominant bit over those sending a recessive bit.

An interference detected on the bus may also result either from a transmission error of internal or external origin (see error management) or from a dominant physical level code symbol on the bus (e.g. End Of Data).

6.3.3.5 Bitwise arbitration

The arbitration system defined below can settle conflicts in the event of simultaneous accessing. It is applied to each bit of the arbitration area successively, starting with the initial fields (those at the head) of the frame.

During the arbitration phase, the sending module compares the value of the bit sent with the value of the bit passing over the bus. If both these values are equal, it continues to send the next bit. If it detects interference on the bus, it immediately stops sending its series of bits. When one module sends a recessive bit while another module is sending a dominant bit, it loses arbitration and abandons its current send attempt.

At each stage of the arbitration process, all conflicting modules which detect a collision will therefore stop sending. The remaining modules will continue to send their bits until the next interference. The module which obtains access to the medium is that which forces the bits of the arbitration area without detecting any interference. All the other modules become receiver of this frame (see figure 26).

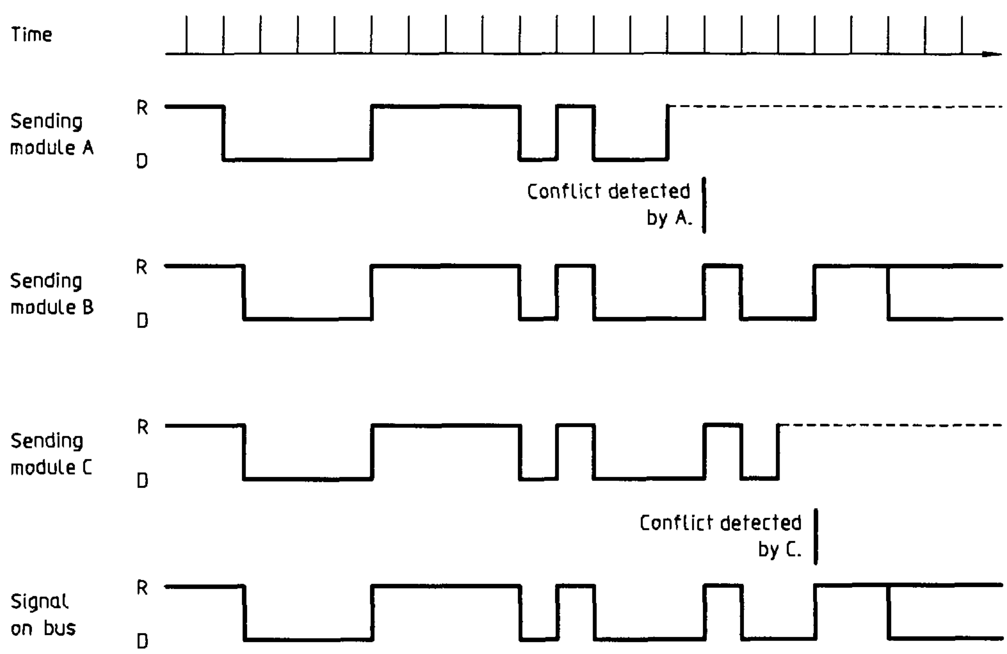


Figure 26 — Mechanism of bitwise arbitration on bus in Enhanced Manchester

6.3.3.6 Arbitration area

The arbitration process described above covers the fields of the frame, starting from the identifier field.

6.3.3.7 Loss of arbitration

A module which detects a contention aborts its current transmission: loss of arbitration is generally unintentional and gives rise to a recovery procedure after conflict detection. However, in the case of an autonomous module polling a slave module, transfer of bus control to the polled module may be intentional if it occurs at the RTR bit level; the in-frame response mechanism is specified (see 6.3.3.10).

6.3.3.8 Priority

The arbitration mechanism allows implementation of a priority system defined as a function of the bit configuration in the frame arbitration area. Priority is granted to those modules sending dominant bits (logical "0").

In normal operating mode (no transmission error or deadlock), this method ensures access to the medium for the module sending the highest priority frame.

6.3.3.9 Overlap in event of loss of arbitration

When the sending module has interrupted sending following an (unintentional) loss of arbitration, the overlap procedure involves repeating the transmit attempt immediately after bus release.

The recovery algorithm above does not apply in the event of a response inserted in the frame by a polled module (see 6.3.3.10).

6.3.3.10 In-frame response

A module which has obtained access to the medium can request transmission by a remote module:

— in the event of immediate response by the polled module, the communication bus is not released and thus escapes the usual mechanism of data transmission bus allocation;

— if the polled module is not ready to send, the polling module sends a complete frame (request frame without immediate response).

Upon a (remote) request for data by an autonomous module, the polled module is enabled to send its in-frame response by accessing the medium starting from a specific bit (RTR): the polled module sends a dominant bit (logical "0") over the recessive bit sent by the autonomous module in the request (logical "1"). The autonomous module thereby loses arbitration and automatically stops sending to receive the requested data.

6.3.3.11 Principle of in-frame acknowledgement

When a module has received data sent to it, it must indicate, upon an explicit request (RAK), that the frame has been correctly received, using the acknowledgement field reserved for this purpose on the bus.

6.3.3.12 Restriction

A slave module which cannot initiate data transmission autonomously is not enabled to request data from a remote module.

6.3.4 Functions performed by MAC sublayer

The main functions performed by the MAC sublayer are as follows.

a) For frame transmission:

   — acceptance of data coming from the LLC sublayer;

   — frame formatting by adding the specific fields of the frame;

   — presentation to the physical layer of a data stream consisting of serialized bits for transmission via the medium.

   NOTE 2 Data sent via the LLC sublayer are in byte multiples.

b) For (valid) frame reception:

   — receiving a data stream from the physical layer formed of serialized bits;

   — recompiling the structure of the complete frame (frame fields);

   — checking the frame identifier and selecting frames concerning the module (acceptance filter);

   — passing useful information to the LLC sublayer (identifier, command and data).

   A module which starts transmission via the bus manages in parallel the operations relating to frame reception up until final acquisition of the bus.

c) Deferment of frame transmission so long as the physical medium is busy.

d) Management of the transmission time-out to comply with the Interframe Spacing (IFS).

e) Implementation of the arbitration mechanism during transmission of the part of the frame on which this mechanism is applied.

f) Termination of transmission upon collision detection.

g) Notifying the LLC sublayer of any transmission error (see in particular 6.3.5).

h) Scheduling a resend in the event of (unintentional) loss of arbitration.

i) In the event of loss of arbitration (whether intentional or not), continuing to receive the frame passing via the bus until the end of the frame.

j) Adding the calculated Frame Check Sequence (FCS) field to the frame sent.

k) Checking frames at reception by means of the FCS and checking alignment on a whole number of bytes.

l) Insertion of an acknowledgement field in the frame after data checking.

m) Rejecting excessively short received frames (incomplete identifier and command fields).

n) Where applicable, notifying the network administration of the arrival of an invalid frame [other than case m)] or a non-selected frame (address).

o) Removal of specific fields from the received frame.

6.3.5 MAC-level error management

Errors of transmission via the physical medium are signalled by the MAC sublayer to the LLC sublayer.

Error recovery procedures are managed by the LLC sublayer.

Table 5 shows the various types of error indicated by the MAC sublayer.

Table 5 — Types of error

Type	Meaning
Bit error	Error on one of the fixed or symbol fields of the frame (SOF, EOD, ACK, EOF)
Code error	Error on one of the bits of the binary or data fields (IDEN, COM, DAT, FCS): Violation of the Manchester-L or Enhanced Manchester coding.
Format error	Encoding or synchronization error on one of the fixed or symbol fields of the frame (SOF, EOD, ACK, EOF)
CRC error	The value of the CRC remainder calculated by the receiving module does not correspond to the value defined in 6.2.3.5.
ACK error	The ACK field does not comply with the definition of a positive ACK or an absent ACK. The value is ABSENT ACK, and the acknowledgement is requested, or the value is POSITIVE ACK, and the acknowledgement is not requested.

6.3.5.1 Errors in transmit mode

When a module is sending, it can detect an error:

— either when a code violation is detected on the bus;

— or through the absence of the requested acknowledgement field.

In the event of error detection on the bus, the sending module interrupts transmission and waits for:

— the end of the VIOLATION TIME-OUT to enable frame reception;

— the end of the VIOLATION TIME-OUT followed by the IFS to enable the LLC sublayer to send a new frame.

The VIOLATION TIME-OUT is managed by the physical layer.

6.3.5.2 Errors in receive mode

In receive mode, the errors signalled by the MAC sublayer correspond to:

— either a code violation (detected by the physical layer),

— or an invalid frame: the error management mechanisms proposed in MAC are based on the frame check and integrity check (FCS).

In the event of error detection, the frame is ignored and the type of error detected is signalled to the LLC sublayer for possible counting by the network administration.

6.3.6 Implementation parameters

This subclause provides the specific values of the access method parameters which can be used for implementation of this part of ISO 11519.

Table 6 gives the compatible values of access method parameters to be used for transmission over a differential pair having a bit rate of up to 125 kbits/s, assuming that the medium has the properties described for the physical layer of this part of ISO 11519.

Table 6 — Access method parameters

Parameter	Value
Maximum length of frame	255 bits (16 + 28x8 + 15)
Maximum length of frame	31 bits
Maximum number of specified transmit attempts	not specified
Identifier length	12 bits
IFS	61 corrected clock periods

7 Description of physical layer

This clause gives the functional description of the physical layer. A setup example is described in annex A.

The physical layer consists of three main sublayers:

— encoding/decoding and synchronization;

— line transmitter/receiver;

— connections.

7.1 Specification of physical service

7.1.1 Object and notations

This subclause specifies the services supplied by the physical layer to the MAC sublayer within the framework of the VAN architecture described in this part of ISO 11519 (see figure 1 for the breakdown by layers).

7.1.2 General description of physical service

The services provided by the physical layer are designed to allow data exchange between a local user entity (MAC sublayer) and peer entities (MAC sublayers) connected to the communication bus.

The physical services proposed in this part of ISO 11519 are of three types:

— **character sending service:** This service allows the local entity to send a character (bit or symbol) over the transmission medium without a call connection;

— **character reception service:** This service allows the local entity to receive a character (bit or symbol) over the transmission medium;

— **status bus service:** This service allows the local entity to know the status of the bus (free or busy).

7.1.3 Detailed definition of physical service

7.1.3.1 Service specification method and notations

This subclause describes the aspects of the physical layer service (corresponding to the various functionalities of the physical layer available to users located in the higher-level layer).

This is an abstract representation (or model) of an interface between the physical layer and a user of the physical service, independent of any particular application.

The definition of physical services proposed complies with the ISO reference mode. In particular, it employs the associated service notations in ISO/TR 8509.

7.1.3.1.1 Classification of service primitives

The service primitives used are of three types: request, indication, and confirm. Their summary definitions are given below:

Request The REQUEST primitive is passed by a service (N)-user to the (N)-layer or sublayer to request initiation of a service.

Indication The INDICATION primitive is passed by the (N)-layer or sublayer to a service (N)-user to indicate that an internal event in (N)-(sub)-layer has occurred, and that said event is significant for the service (N)-user. An indication can be triggered by a service request executed previously or by an internal event in (N)-(sub)-layer.

Confirm The CONFIRM primitive is sent by the (N)-layer to a service (N)-user to retrieve the results associated with the previous service request.

Figure 27 gives typical examples of service primitives exchanges. In case 1, the indication is directly related to a remote service request: e.g. character transfer initiated by a sending module (see PL DATA request definition).

In case 2, the indication denotes an internal change of the physical layer which may be signalled to its service user: e.g. a change in the bus state (see 7.1.3.3.3).

This event may be caused by a module which starts transmitting or may be caused by a bus perturbation (EMC) or any internally detected event. In any case, there is no relation established between a possible cause and its effect.

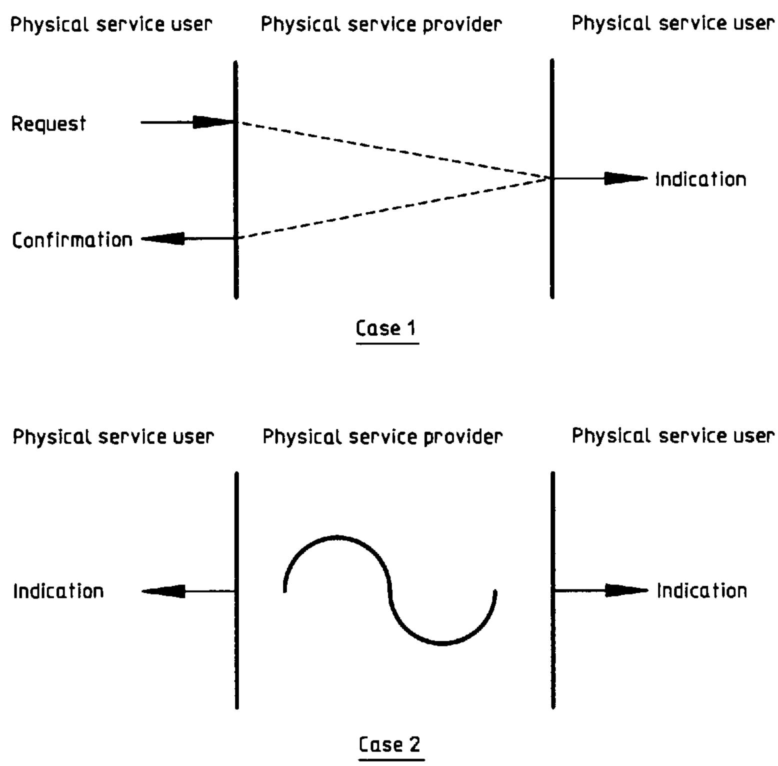


Figure 27 — Schematic diagram of interactions between adjacent layers

7.1.3.2 Description of service interactions

Table 7 gives a list of the service primitives characterizing each element of the physical service described in this document. Subclause 7.1.3.3 gives a detailed description of each of these service primitives.

Table 7 — List of physical service primitives

Character transfer service	
Character sending service	
PL-DATA.Request	Request to send a character.
PL-DATA.Confirm	Confirmation of character sending.
Character reception service	
PL-DATA.Indication	Character reception indication.
Status bus service	
PL-BUS-STATUS.Indication	Indication of bus free or busy.

7.1.3.3 Detailed specification of physical services

This subclause gives a detailed description of the primitives and associated parameters corresponding to the various physical services described in 7.1.3.2.

The CHARACTER parameter represents the value of the character (binary value of the bit or type of symbol) to be transmitted.

The TERMINATION character can indicate the method for start and end of data transmission.

The RECEPTION-STATUS parameter specifies whether or not the transmission of a character (bit or symbol) has been successful, and the status of the medium (busy or free).

7.1.3.3.1 Primitives associated with character sending service

7.1.3.3.1.1 PL-DATA.request

a) **Function**

This primitive allows transmission of a character between the local MAC entity and one or more destination MAC entities.

b) **Parameters associated with PL-DATA.request primitive**

This primitive includes the following parameter:

— PL-DATA.request (CHARACTER,TERMINATION).

Figure 28 gives examples of possible exchanges of primitives associated with the character sending service.

c) **Operation associated with this primitive**

This primitive is passed by a MAC entity to the physical layer whenever it needs to send a character over the medium to one or more remote MAC entities.

Upon receiving this primitive, the physical layer encodes the character in a specific format and transfers it to the medium.

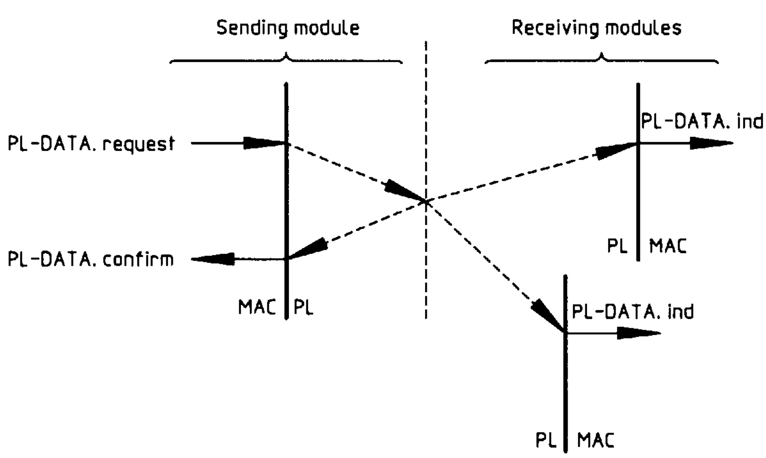


Figure 28 — Example of primitives exchanges

7.1.3.3.1.2 PL-DATA.confirm

a) **Function**

This primitive allows confirmation to the MAC sublayer of the transmission of a character.

b) **Parameters associated with the PL-DATA.confirm primitive**

This primitive includes the parameters:

— PL-DATA.confirm (TRANSMISSION-STATUS).

The TRANSMISSION-STATUS parameter indicates the success or failure of the transmission of a character:

— TRANSMISSION-OK

— INVALIDITY-TRANSMISSION (Code violation)

— ARBITRATION-LOSS

7.1.3.3.2 Primitives associated with the character receiving service

7.1.3.3.2.1 PL-DATA.indication

a) **Function**

This primitive allows the physical layer to indicate to the MAC sublayer the reception of a character.

b) **Parameters associated with the PL-DATA.indication primitive**

This primitive includes the parameters:

— PL-DATA.indication (CHARACTER, RECEPTION-STATUS).

The CHARACTER parameter is identical to that supplied in the corresponding request. The CHARACTER specifies the bit or symbol received by the physical layer.

The RECEPTION-STATUS parameter is used to pass up status information to the MAC sublayer. It includes a code which can indicate the validity of a character, and information concerning the medium's status.

c) **Operation associated with this primitive**

The PL-DATA.indication primitive is passed by the physical layer to a MAC entity to indicate the arrival of a character.

d) **Additional comments**

The codes which can be conveyed by the RECEPTION-STATUS parameter are as follows:

— CHARACTER-OK: indicates the arrival of a valid character.

— CHARACTER-ERROR: can be used to indicate the arrival of an invalid character (Code violation).

7.1.3.3.3 Primitives associated with status of bus

7.1.3.3.3.1 PL-BUS-STATUS.indication

a) **Function**

This primitive allows the physical layer to indicate the bus status.

b) **Parameters associated with PL-DATA.indication primitive**

This primitive includes the parameters:

— PL-BUS-STATUS.indication (BUS-STATUS).

The BUS-STATUS parameter indicates the status of the bus, free or busy.

c) **Operations associated with this primitive**

The PL-BUS-STATUS.indication primitive is passed by the physical layer to a MAC entity to indicate the status of the bus.

7.2 Encoding/decoding and synchronization sublayer

7.2.1 Function

The function of this sublayer is to:

— implement links between the MAC layer and the line transmitter/receiver sublayer;

— encode/decode bits depending on the selected code (Manchester-L or Enhanced Manchester);

— generate the SOF, EOD and EOF symbols upon a request by the MAC sublayer;

— generate the internal clocks used by this sublayer (**Hcor** which is the base clock used for sampling and sequencing);

— generate the various clocks used by the MAC sublayer:

Hsymb which is the symbol synchronization clock,

Hpl which is a submultiple of the base clock, corresponding to:

16 Hcor periods in Enhanced Manchester,
32 Hcor periods in Manchester-L.

7.2.2 Breakdown by layers

Figure 29 gives the functional breakdown by layers and the various signals allowing interfacing of the MAC sublayer on the one hand and the line transmitter/receiver on the other hand.

7.2.3 Bit encoding

Two types of encoding can be selected at the level of this sublayer: Manchester-L or Enhanced Manchester code. On a given network, all modules must use the same code.

Figures 30 to 32 show the various symbols in relation to Hcor for the two codes.

The Enhanced Manchester code consists of one Manchester bit inserted every 3 NRZ bits (see figure 32). Insertion of the Manchester bit allows receiver resynchronization.

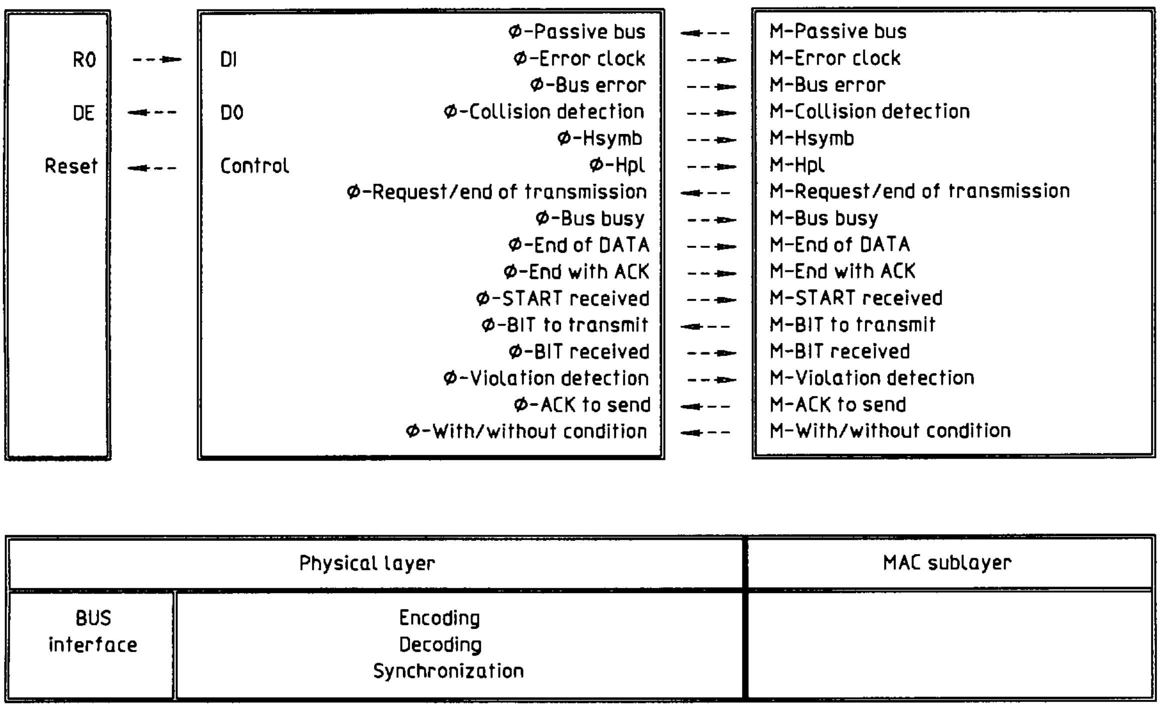


Figure 29 — Interconnections between physical layer and MAC sublayer

Table 8 — Dialogue signals between MAC sublayer and physical layer — Interface I-3

Name	Direction	Definition
Bus inhibited	from MAC	Signal setting DO output to recessive state
Clock error	to MAC	Signal indicating a clock corrector overflow
Err bus	to MAC	Signal indicating the presence of a dominant level of longer duration than the Start Of Frame signal
Collision Detection	to MAC	Signal indicating collision in progress
Hsymb	to MAC	Clock signal that synchronizes, serializes and deserializes the various symbols generated by MAC
Hcor	to MAC	Corrected base clock signal
Hpl	to MAC	Clock signal which is a sub-multiple of Hcor, allowing time delay generation by MAC
Request to send	from MAC	Transmission initialization and finalization signal
With/without condition	from MAC	Signal which, in conjunction with the request to send signal, can generate the sequences required for in-frame response or for line abort after detecting a collision or a code violation
Bus busy	to MAC	Signal indicating presence of signals on DI
Transmit enable	to MAC	Signal indicating the capability to begin a transmission
End of data	to MAC	Signal indicating reception of an EOD
End with ACK	to MAC	Signal indicating reception of an ACK
START received	to MAC	Signal indicating reception of a Start Of Frame
Bit to transmit	from MAC	Signal indicating value of a bit to transmit
Bit received	to MAC	Signal indicating value of a bit received
Violation	to MAC	Signal indicating presence of elementary states on the bus which cannot be interpreted as a (1) or (0) bit or a symbol (SOF, EOD, EOF, ...)
ACK to transmit	from MAC	Request to send an acknowledgement

Table 9 — Dialogue signals between line interface and encoding/decoding synchronization sublayer — Interface I-2

DI	from line transmitter/receiver (RO)	Received data
DO	to line transmitter/receiver (DE)	Transmitted data
Control	to line transmitter/receiver	Signal changing reception comparators to recessive state

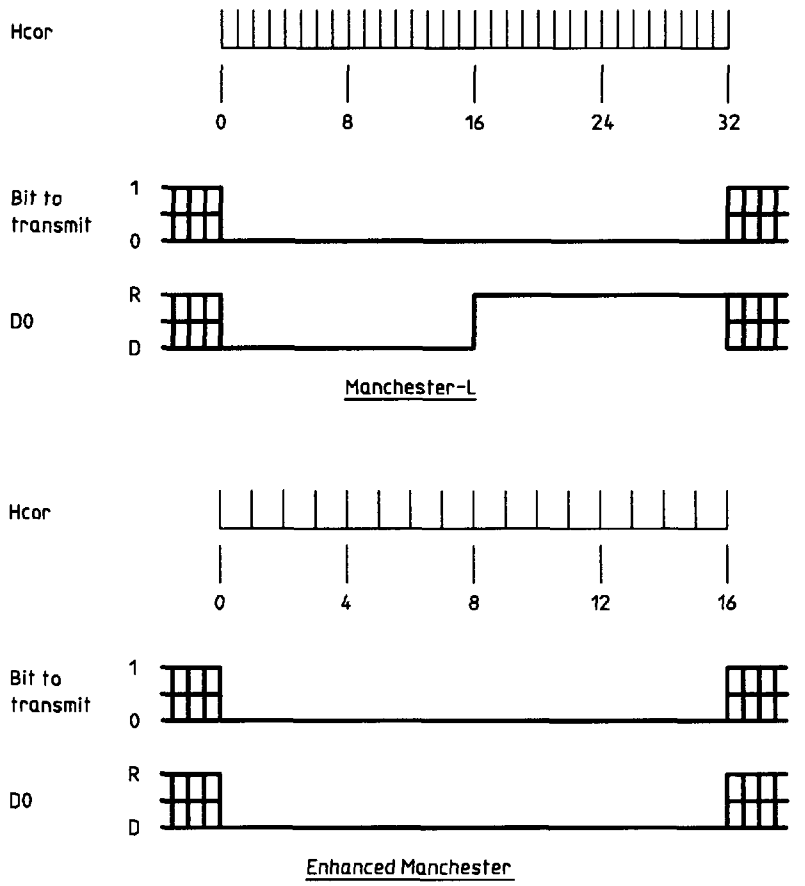


Figure 30 — Description of "0" bit or dominant bit

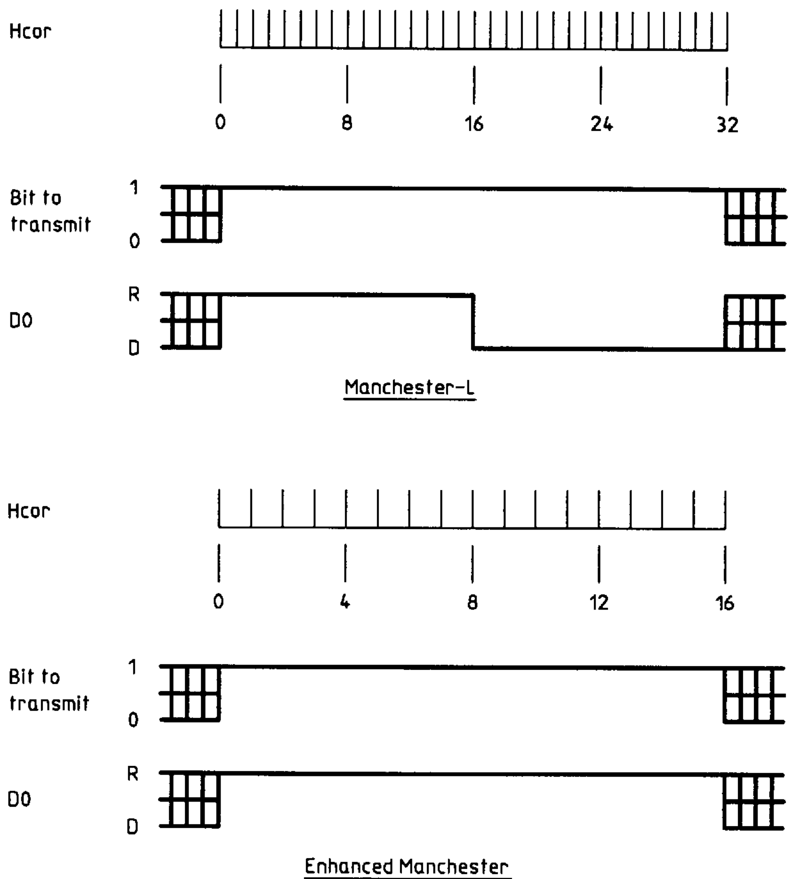


Figure 31 — Description of "1" bit or recessive bit

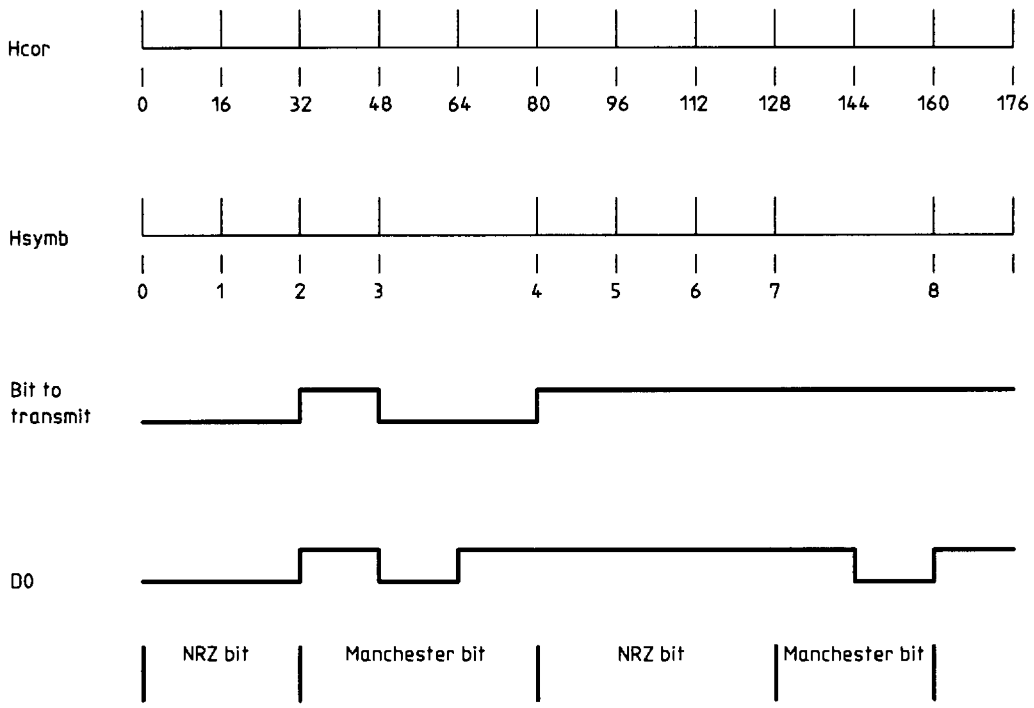


Figure 32 — Description of bit string: Enhanced Manchester

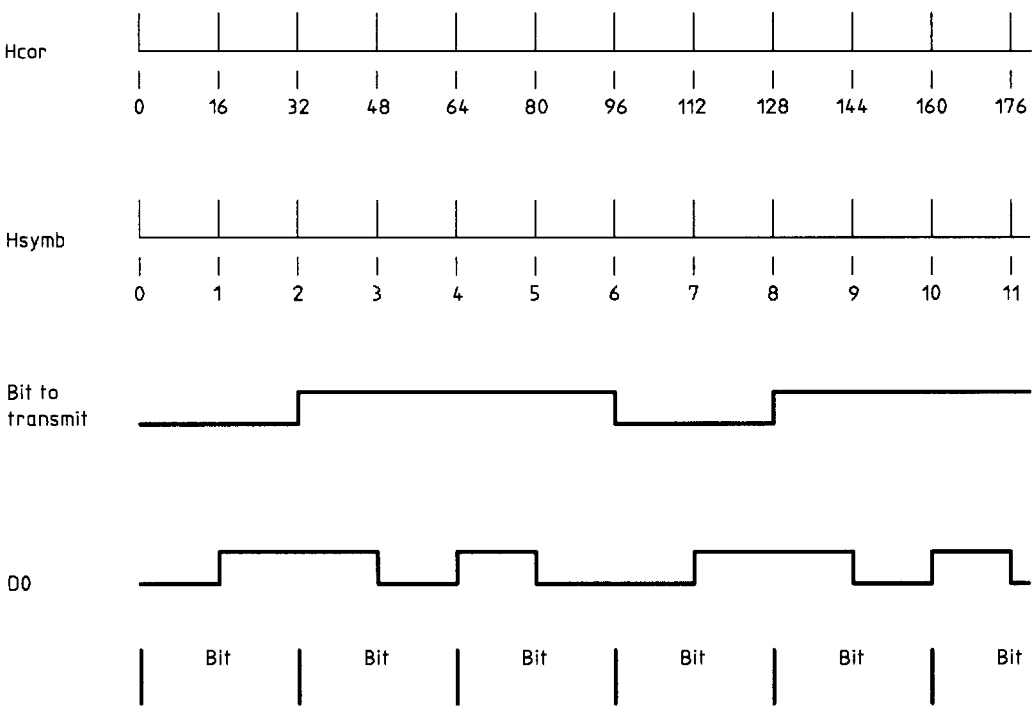


Figure 33 — Description of bit string: Manchester-L

7.2.4 Synchronization

This unit contains three sub-units:

— a sub-unit for generation of the various clocks;

— a bit synchronization sub-unit;

— a sub-unit for synchronization of the various symbols.

7.2.4.1 Clock generation

Each module includes a clock generator circuit (Hlocal) whose accuracy depends on the class of module and correction device used. This local clock generates the symbol clock Hsymb, the bit clock Hpl and the corrected clock Hcor.

— In the case of autonomous modules, the clock precision must be at least ± 3 %;

— For slave modules and synchronous access modules, the clock accuracy must be better than ± 20 %.

In both cases, a correction device allows generation of a corrected local clock (Hcor) using the SOF as a reference time. The purpose is to obtain a relative accuracy of ± 1 % between all the modules in communication on the same bus.

If the relative clock accuracy of each module is better than ± 1 %, the correction device is not required.

7.2.4.2 Bit synchronization

This subclause describes the synchronization rules employed, with special emphasis on three important points:

— the resynchronization point;

— the sampling point;

— the transmission point.

The synchronization rule is based on the "received" signal edges (see figure 36) and is the same for all modules whether they be sending or receiving. During preamble sending, the synchronization rule is disabled for the preamble initiator.

A signal edge detected at the moment t of Hcor clock of the receiver involves a new resynchronization at $t + 1$ ($t + 1$ Hcor period). At $t + 1$ the value of the synchronization counter is forced as shown in table 10.

7.2.4.3 Synchronization of various symbols

Figures 36 to 39 show the timing diagrams of the various symbols used.

7.2.5 Rules for signal exchange between MAC sublayer and the physical layer

The rules for exchange between the MAC layer and the physical layer are based on the use of the H symbol clock:

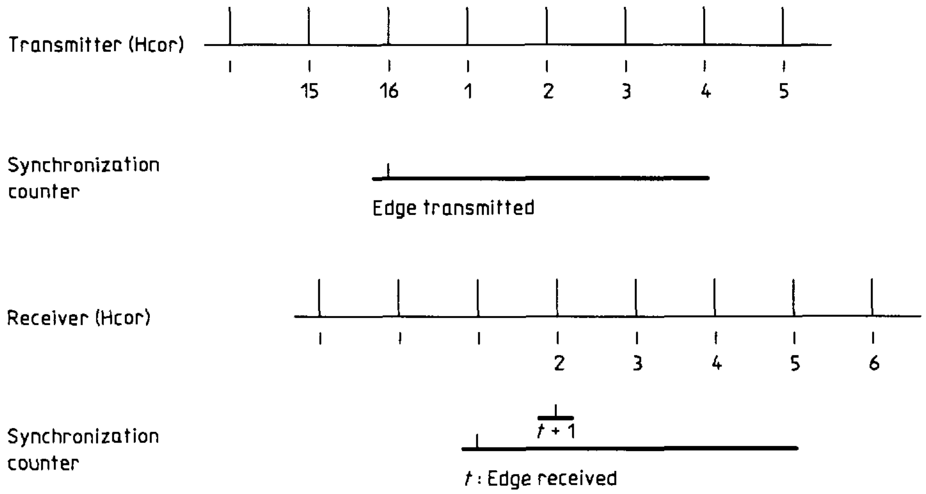
— Signal acquisition by the MAC layer for signals coming from the physical layer must take place on the rising edge of H symbol (Hsymb).

— These signals must be present at the interface level at least 1/2 corrected clock period (Hcor) before the appearance of H symbol (Hsymb). See figure 40.

— Signal acquisiton by the physical layer for signals coming from the MAC layer must take place on the falling edge of H symbol.

— These signals must be present at the interface level at least 1/2 corrected clock period (Hcor) before the appearance of H symbol (Hsymb). See figure 40.

Table 10 — Specific points of synchronization counter



	NRZ bit	Manchester bit 1st TS	Manchester bit 2nd TS
Value of synchronization counter forced at $t+1$	2	2	18
Sampling of collision detection and TS level	11	11	
Sampling of bit value and TS level	11		27
Transmission point	16	16	32

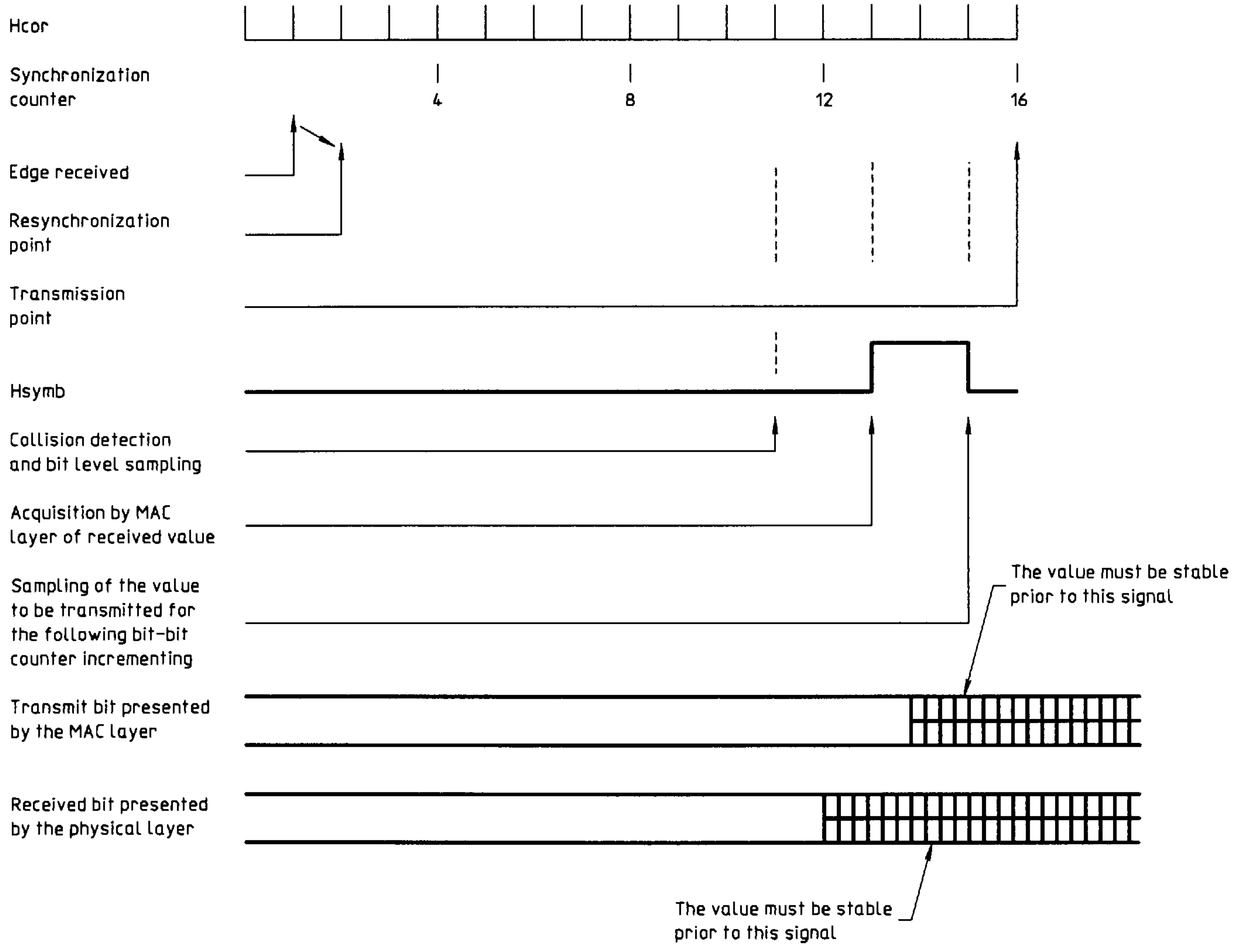


Figure 34 — Bit synchronization: Bit NRZ

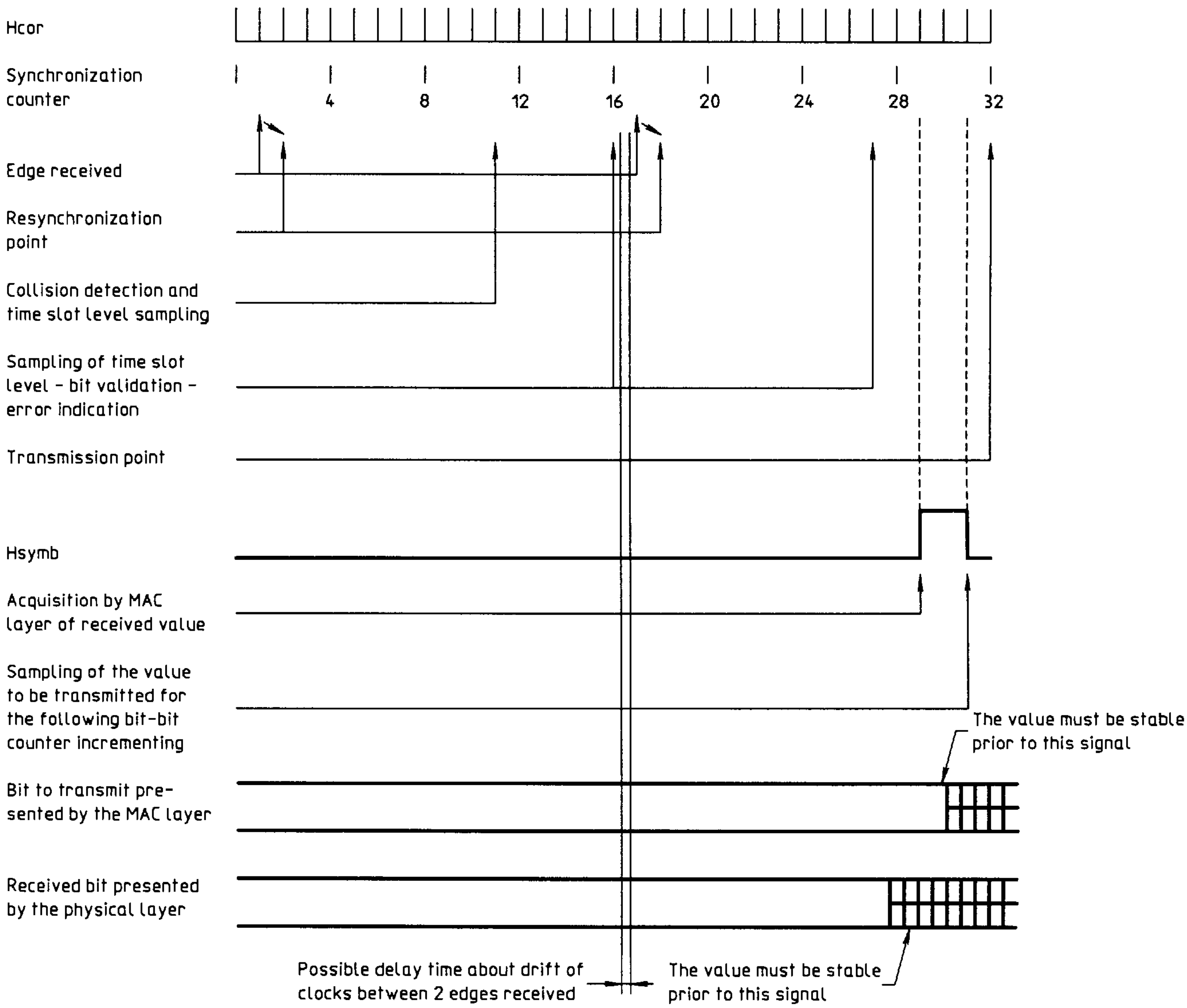
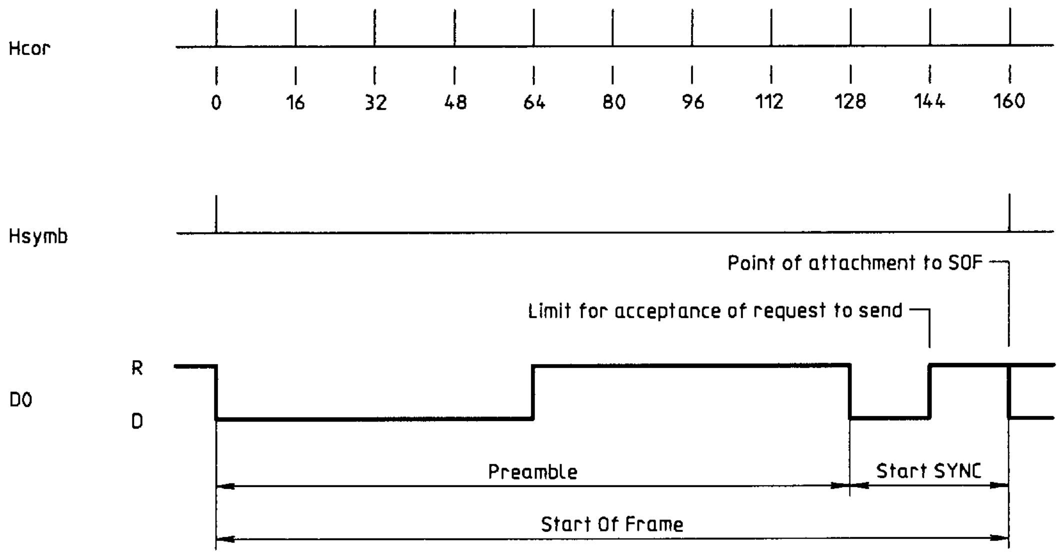


Figure 35 — Bit synchronization: Bit Manchester



The Start Of Frame delimiter must be preceded by a 128 Hcor periods idle character.

Figure 36 — Description of Start Of Frame (SOF) delimiter — Manchester-L and Enhanced Manchester

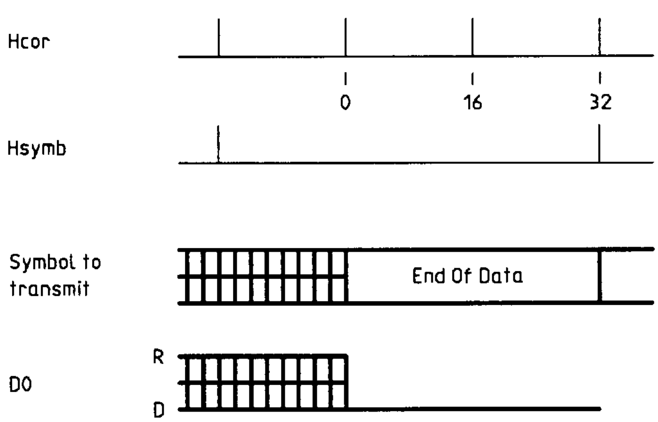


Figure 37 — Description of End Of Data (EOD) delimiter — Manchester-L and Enhanced Manchester

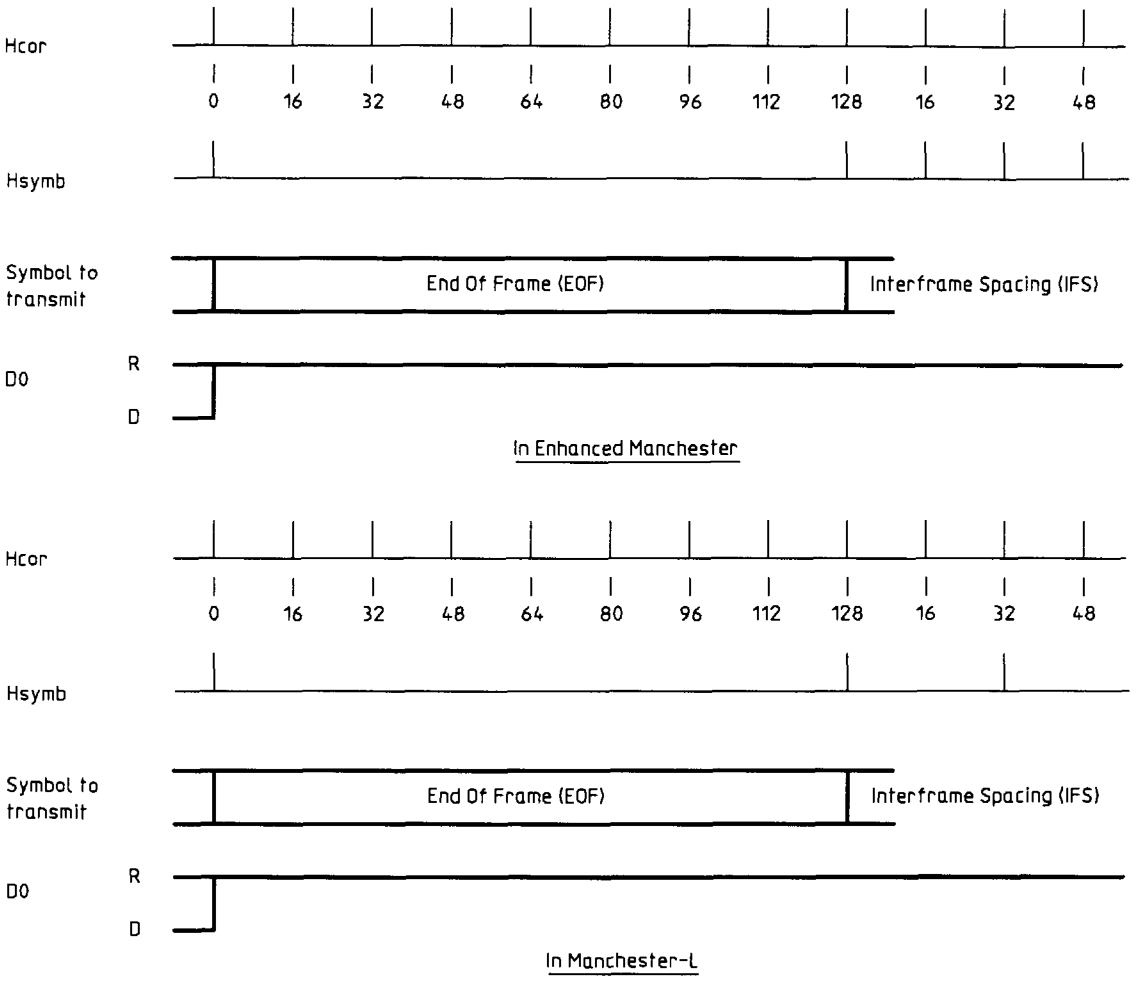
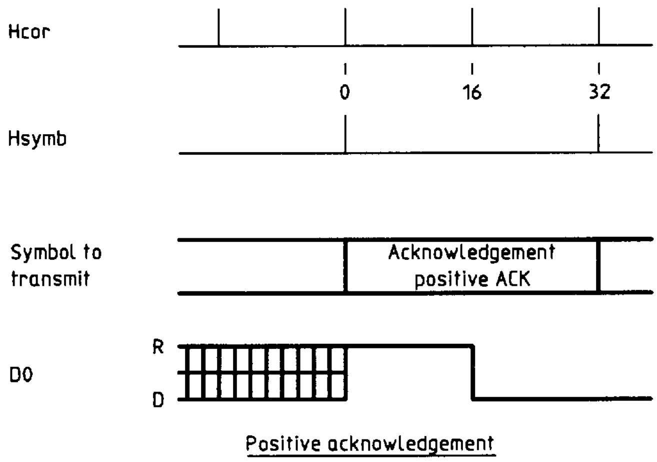


Figure 38 — Description of delimiter End Of Frame and Interframe Spacing



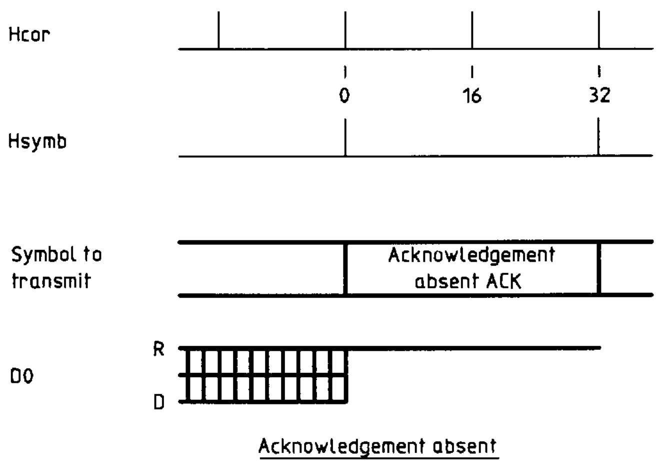


Figure 39 — Description of acknowledgement field — Manchester-L and Enhanced Manchester

7.2.6 Exceptions to this signal exchange rule

Three signals can be acquired or generated asynchronously:

— the passive bus signal,

— the clock error signal,

— the bus error signal.

7.2.7 Timing diagram for various signals

Frame sending initialization rule:

A transmit attempt requested by the MAC sublayer can be initiated by the physical layer with the following rules[5]:

a) either the bus is not busy and the frame is initialized by an SOF symbol;

b) or the bus is busy with an SOF; in this case, frame sending starts at the end of the SOF symbol beginning from the first bit in the identifier field (attachment point).

The transmit attempt, indicated by the "Request/End of Transmission" signal, is only taken into account if it reaches the physical layer before the transmit request acceptance deadline; i.e. before the falling edge of the signal "transmit enable".

See figure 43 for the definition of the attachment point and the transmit request acceptance deadline. For the principles of transmitting, see figures 42 and 43.

Timing diagrams:

Figures 41 to 46 give the timing diagrams for the various interface signals between the physical layer and the MAC layer.

7.2.8 Details concerning "violation" information

The "violation" information indicates:

— a transgression of the bit encoding (Manchester-L or Enhanced Manchester) or symbol encoding rules,

— a start too short or too long,

— a symbol incorrectly inserted in a frame,

— a symbol of abnormal length (e.g. EOF),

— a transmitted dominant bit converted into a read recessive bit,

— a bit of abnormal length (i.e. more than one transition per Time Slot).

Once a violation is detected, the clock Hsymb is generated to allow an End Of Frame (EOF) and then systematically with a period of 1-bit duration.

When a violation is detected, the DO output must be set to 1 pending retrieval of the bus clear information indicated by the bus busy link (after a time-out of 128 clock periods) before a new transmit attempt can be accepted.

A new transmit attempt can only be made after waiting for a further 64 corrected clock periods.

5) Autonomous modules comply with rules a) and b). Synchronous-access modules comply with rule b). These rules do not apply to slave modules.

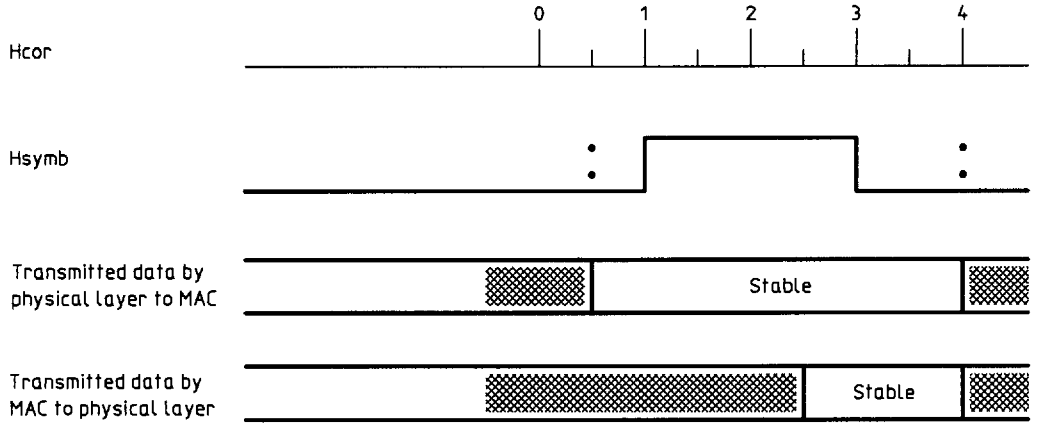


Figure 40 — Information exchange rule between MAC sublayer and physical layer

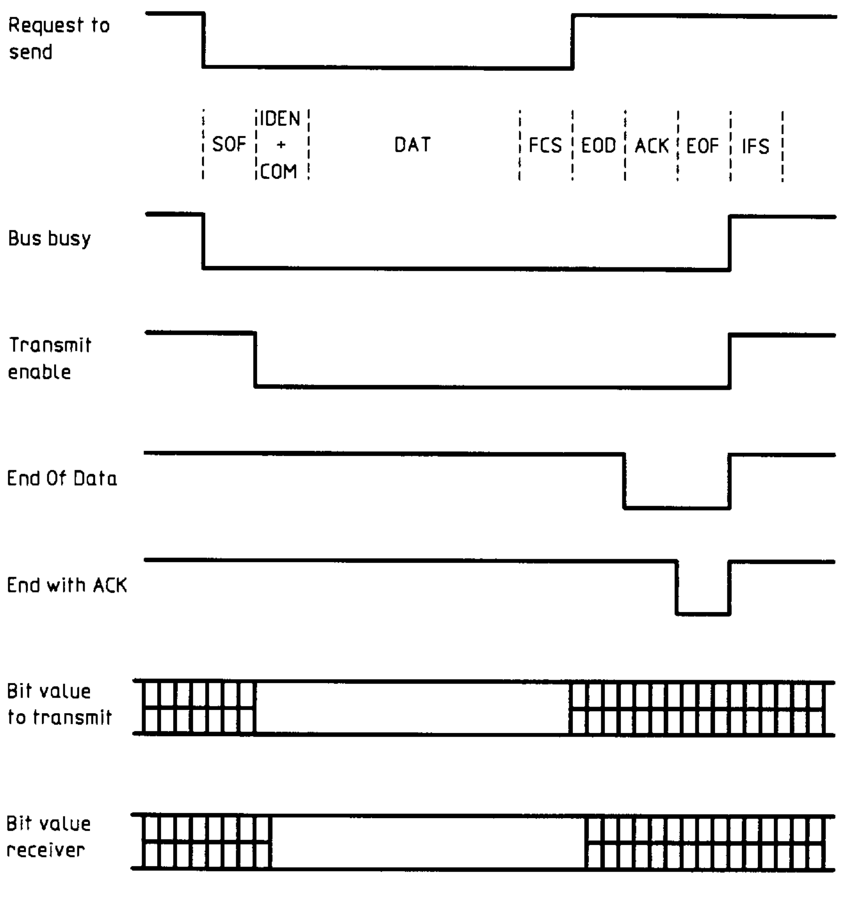


Figure 41 — Symbol in frame

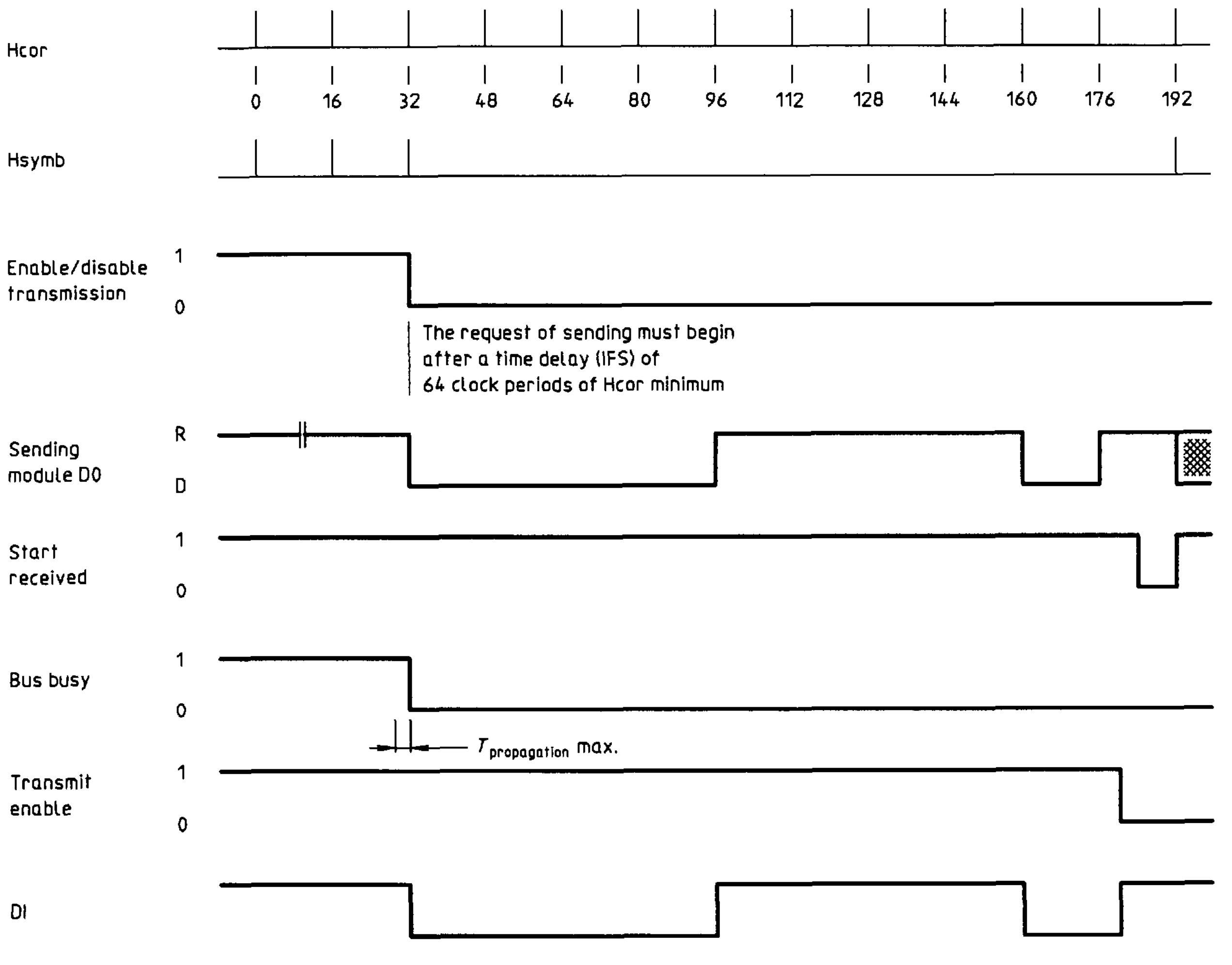


Figure 42 — Timing diagram of "request to send" signal in Enhanced Manchester

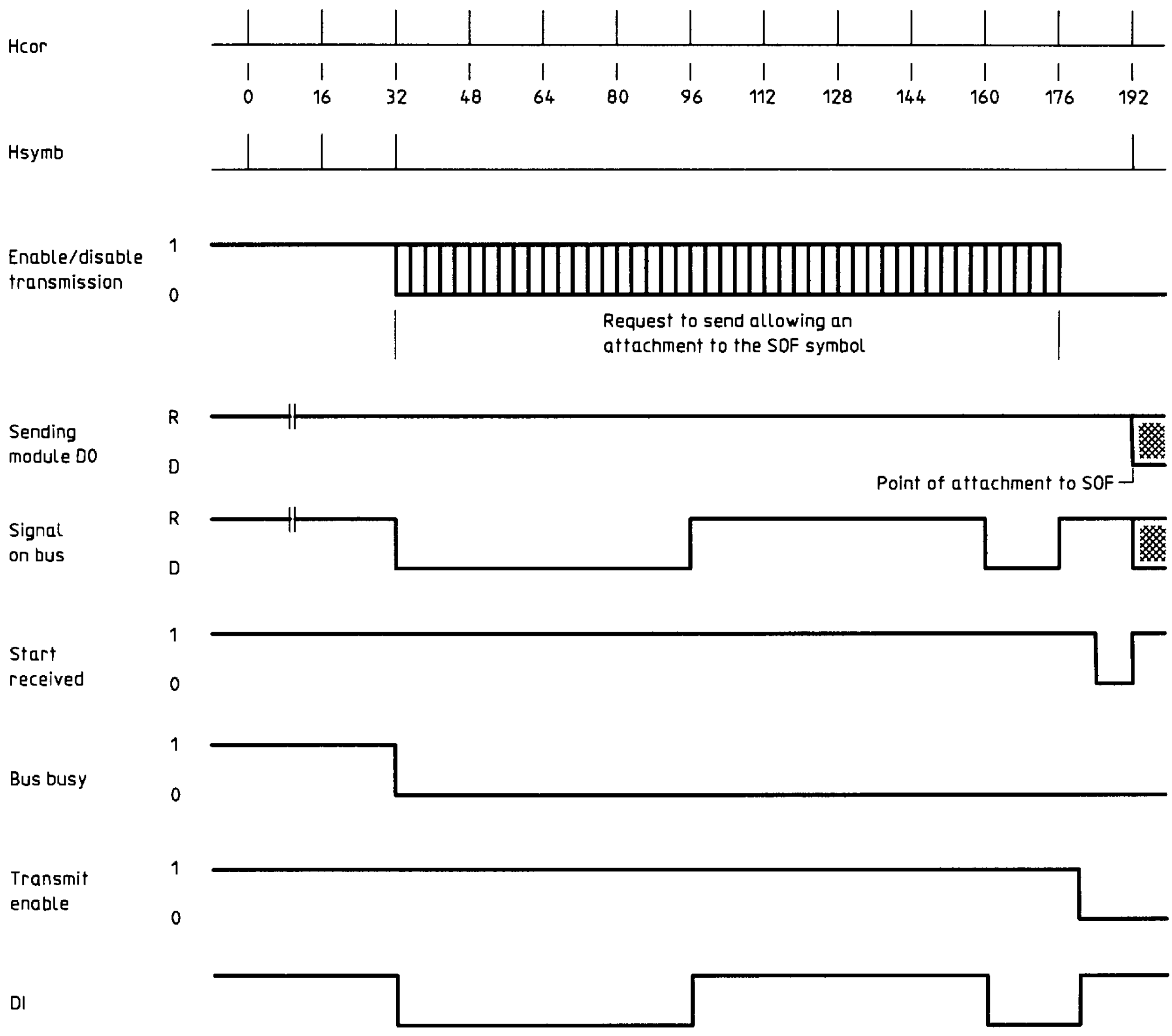


Figure 43 — Timing diagram of "request to send" in Enhanced Manchester — Attachment to SOF symbol

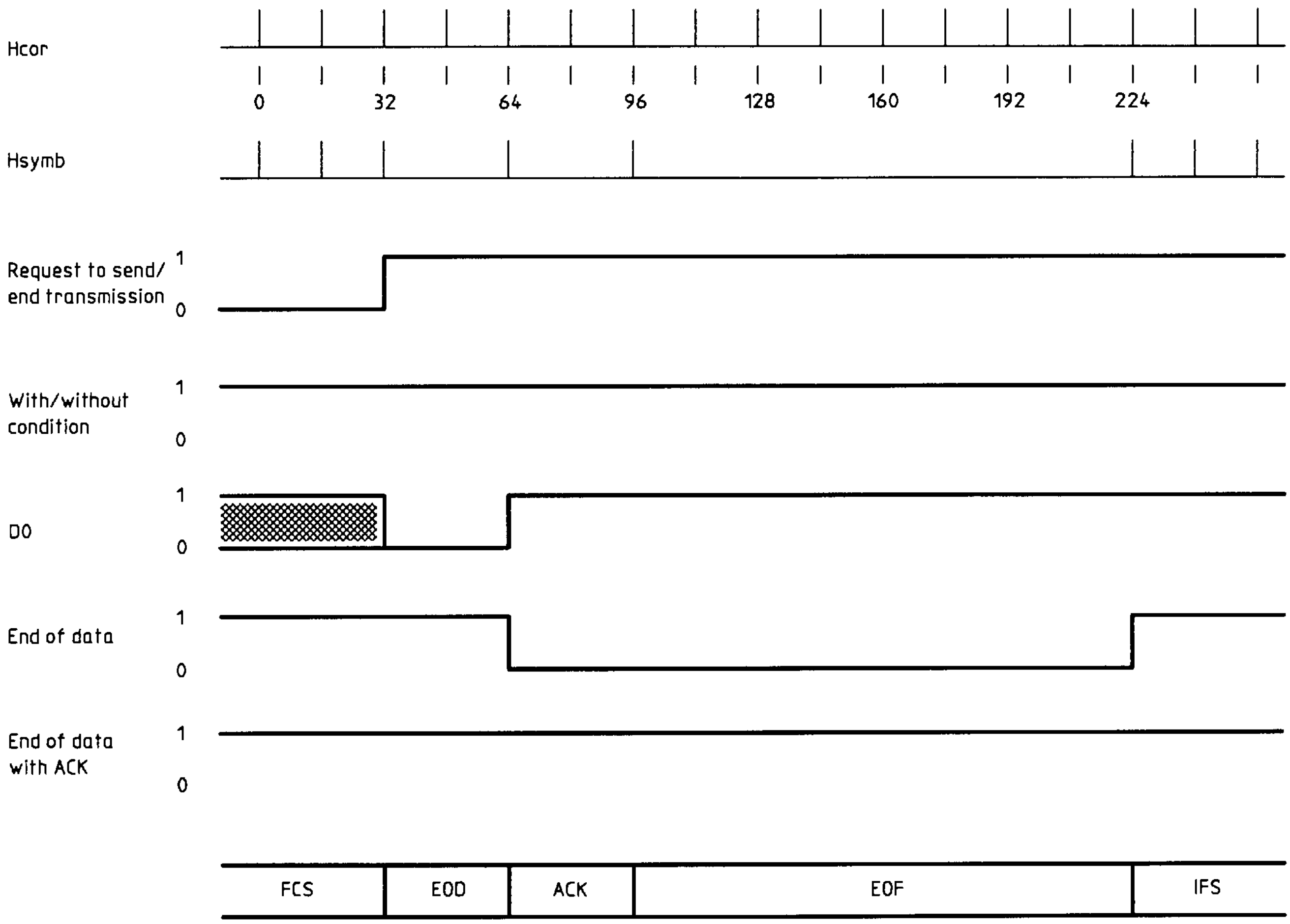


Figure 44 — Timing diagram of "request to send, end of data without request for acknowledgement" signal in Enhanced Manchester

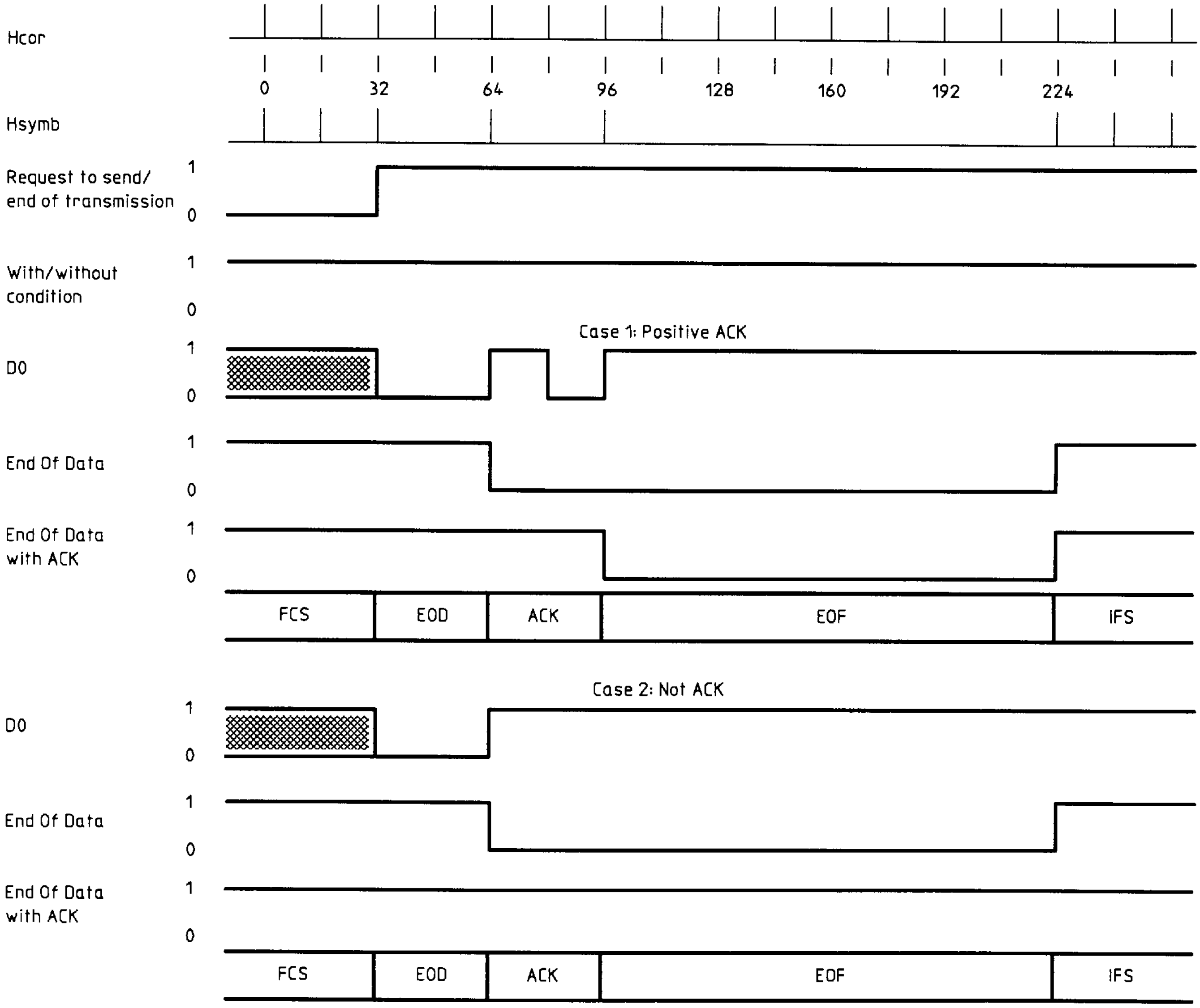


Figure 45 — Timing diagram of "request to send, end of data with request for acknowledgement" in Enhanced Manchester

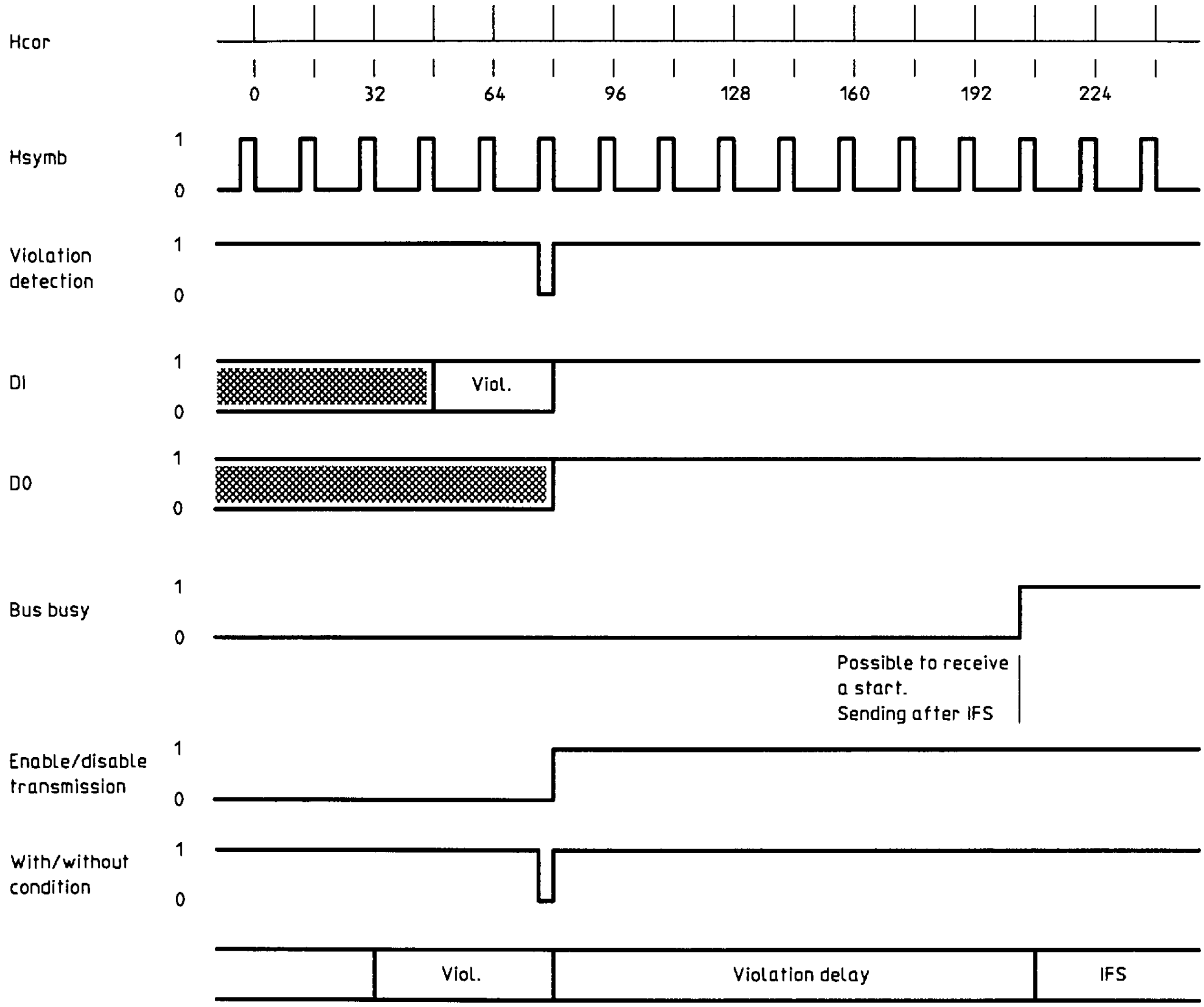


Figure 46 — Timing diagram of signals during violation code in Enhanced Manchester

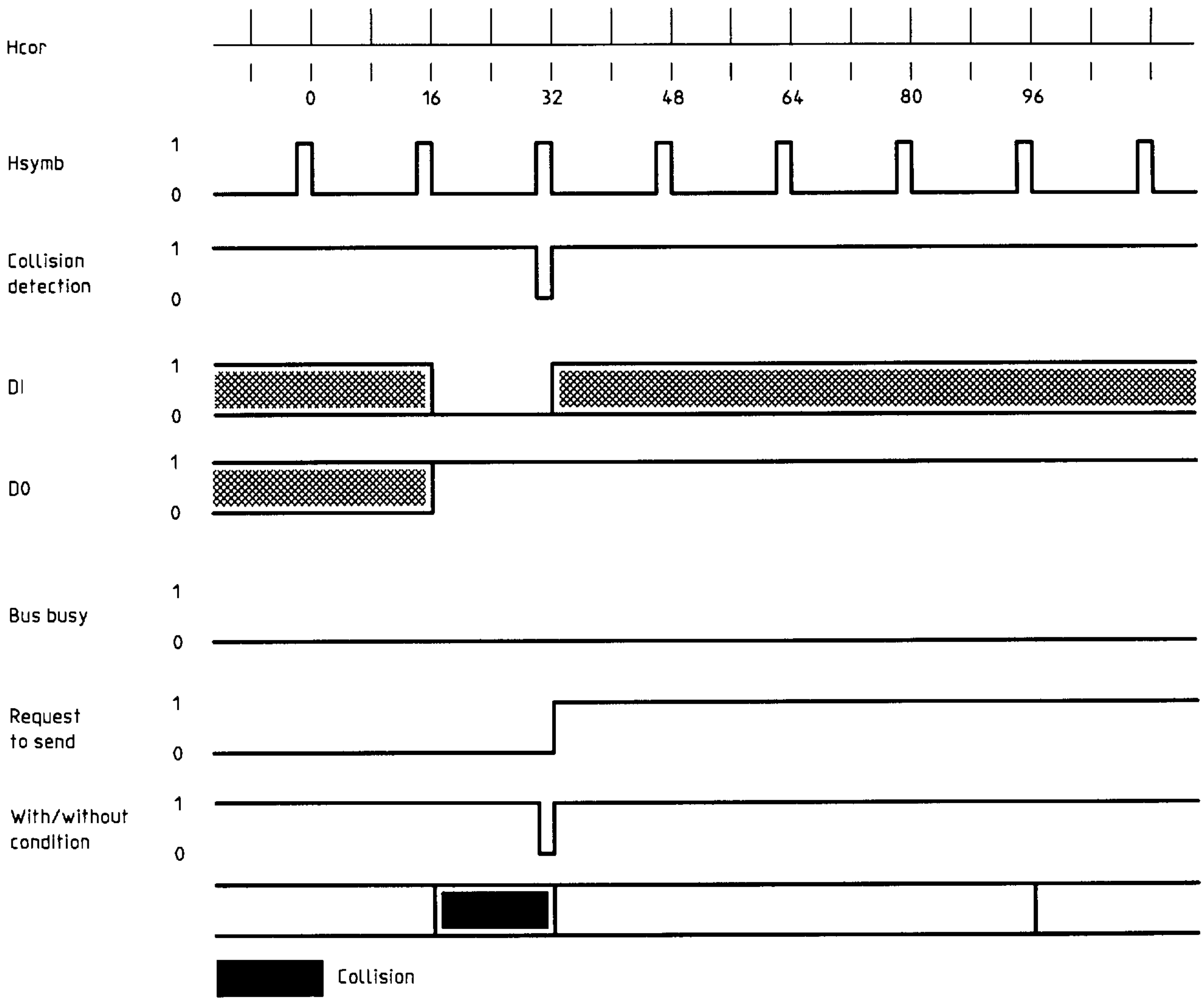


Figure 47 — Timing diagram of signals in event of collision in Enhanced Manchester

7.3 Line transmitter/receiver

7.3.1 Description

The line interface unit consists of four current generators for the sending section, and a comparator and an analogue filter for the reception section. It allows differential data transmission and reception, thus minimizing noise emission.

Transmission in fallback mode on a single wire is also possible.

7.3.2 Breakdown by layers

See figure 48.

7.3.3 Designations

Name	Direction	Definition
Reset	To line transmitter/receiver	Signal of comparator setting to recessive state
DE	To line transmitter/receiver	Level to be sent
RO	To MAC level	Level received by the differential comparator
DATA	To line transmitter/receiver	Data line whose signals are received in phase with DE
DATA	To line transmitter/receiver	Receiver in reverse phase to DE

7.3.4 Description

The sending section includes four current generators, two used to generate the recessive state and the other two to generate the dominant state. These four generators are actuated by the DE input.

A logical "1" on the DE input generates a recessive state, while a logical "0" generates a dominant state.

NOTE 3 The current corresponding to the recessive state is always applied to the bus, even during a reset. The reception section consists of an analog filter and a comparator as shown in figures 49 and 50. The analog filter consists of two parts: a line matching part and a filter. The diagram is given in figure 50.

7.3.5 Timing diagram of signals

See figure 51.

7.3.6 Security constraints

The transmission system is designed to operate on a single data wire in the event of disconnection, battery short-circuiting, or grounding of the other communication medium wire.

Disconnection of the DE/DO link should not generate a persistent dominant level on the transmission lines.

Disconnection of the reset link should not inhibit reception.

7.3.7 Parallel connection constraints

The line interface should be able to energize one or more circuits implementing the upper layers under a specific interfacing.

NAME	DIRECTION	DEFINITION
Reset	To line transmitter/receiver	Signal of comparator setting to recessive state
DE	To line transmitter/receiver	Level to be sent
Ro	To MAC level	Level received by the differential comparator
DATA	To line transmitter/receiver	Data line whose signals are received in phase with DE
$\overline{\text{DATA}}$	To line transmitter/receiver	Receiver in reverse phase to DE

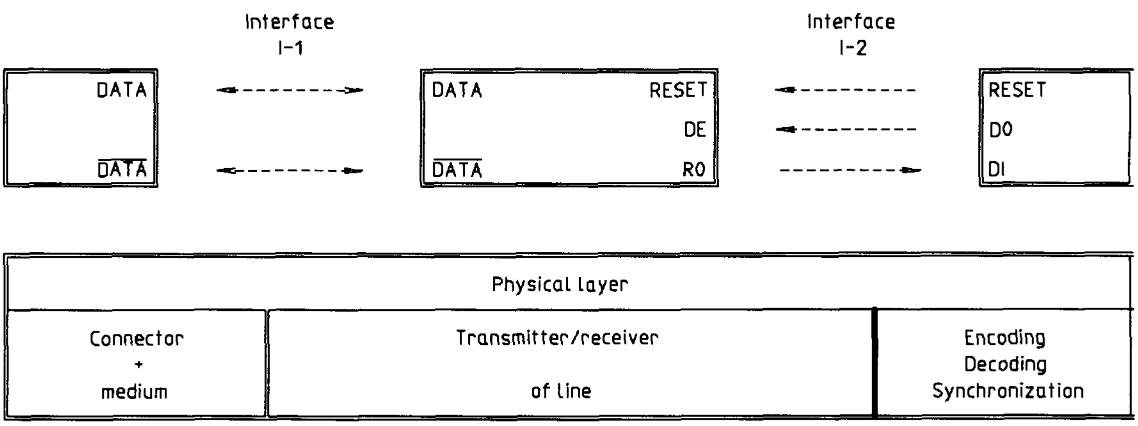


Figure 48 — Interconnections between physical layer levels

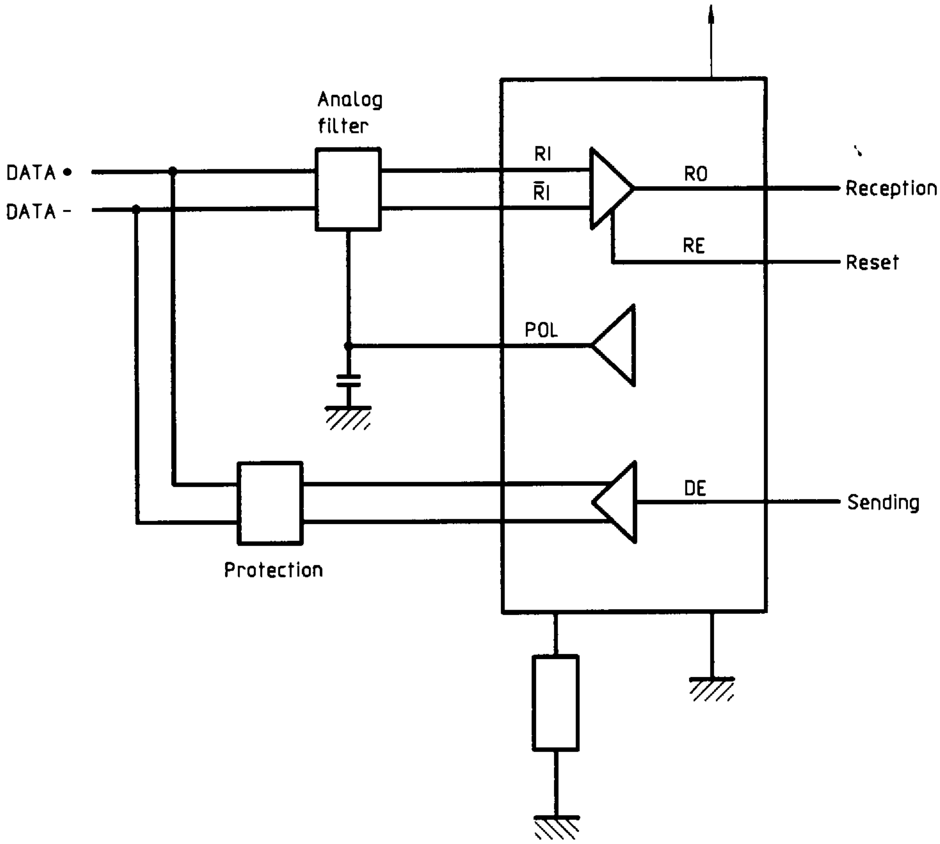


Figure 49 — Block diagram of line transmitter/receiver

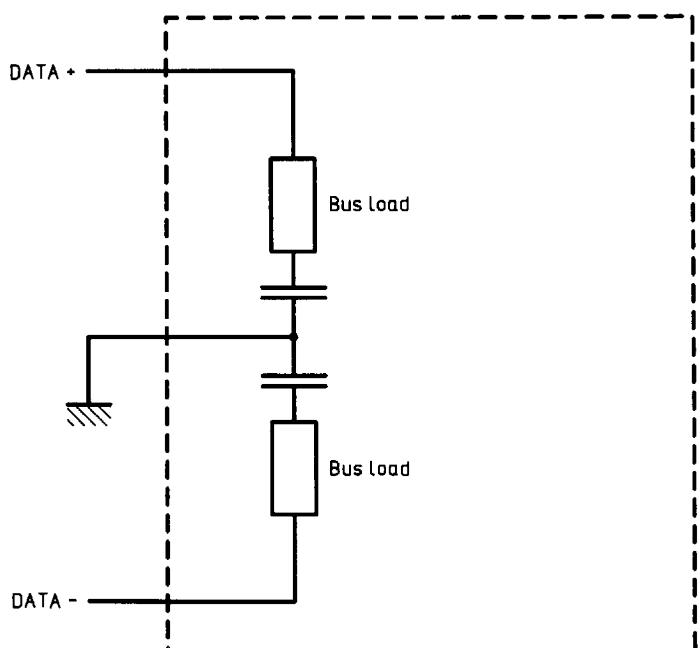


Figure 50 — Option: Analog filter

Transmitter/receiver input/output	State	Level
DATA sending Input DE	Recessive Dominant	"1" "0"
DATA output	Recessive Dominant	
$\overline{\text{DATA}}$ output	Dominant Recessive	
DATA reception Output RO	Recessive Dominant	"1" "0"
Enable/reset Input		"1" "0"

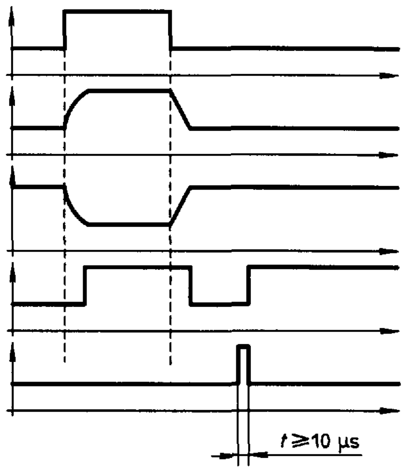


Figure 51 — Timing diagram

7.4 Connector

Connectors are not described as such in this part of ISO 11519 except at the level of data transmission parameters:

— connector pin spacing (for a flat ribbon cable);

— electrical resistance;

— minimum current.

The numerical values of these parameters are given in 8.3.4.

7.5 Communication medium

The communication medium consists of solutions which can easily be industrialized in the automotive world. A conventional twisted pair can be used, or else a single flat ribbon cable consisting of two transmission wires. Future extensions employing optical fibres or coaxial cables are not described in this part of ISO 11519.

The characteristics of the communication medium are given in 8.3.5.

8 Electrical parameters

8.1 At LLC sublayer level

In the case of hardware implementation, it is requested that TTL/CMOS compatibility be ensured.

8.2 At MAC sublayer level

In the case of hardware implementation, it is requested that TTL/CMOS compatibility be ensured.

8.3 At physical layer level

This subclause gives the electrical parameters which can characterize the physical layer at the level of:

— the encoding/decoding/synchronization sublayer;

— the transmitter/receiver;

— the filter;

— the connector;

— the communication medium.

8.3.1 Electrical characteristics of encoding/decoding/synchronization sublayer

NOTE 4 These characteristics will be supplied later.

8.3.2 Electrical characteristics of transmitter/receiver

8.3.2.1 Transmitter characteristics

Symbol	Parameters	Value				Conditions
		min.	typ.	max.	unit	
VoD	Differential output voltage (peak-to-peak amplitude)				V	
VMCE	Transmitter common-mode voltage				V	
DVMCE	Transmitter common-mode voltage variation					between 4,5 V and 5,5 V power supply
ILD	Current limiting in dominant mode	40	50	60	mA	between 4,5 V and 5,5 V power supply
ILR	Current limiting in recessive mode	0,8	1	1,2	mA	between 4,5 V and 5,5 V power supply
IF	Leakage current of current generator outputs			10	µA	in the event of standby mode implementation
MI	Divergence between two current generators of identical type	− 10 %		+ 10 %	%	
tpLH and tpHL	Input/output (current) propagation time			0,5	µs	
Etp	Divergence between propagation times for current generators of identical type			0,15	µs	
RRE	Current adjustment resistance	9,9	10	10,1	kΩ	

8.3.2.2 Characteristics of differential receiver

Symbol	Parameters	Value				Conditions
		min.	typ.	max.	unit	
PV	Reference excursion				V	on RI and RIB
VMCR	Common-mode voltage				V	on RI and RIB relative to polarization voltage
VSH	Output high level	Vdc – 1,5		Vdc	V	
VSL	Output low level			0,5	V	
ZMC	Common-mode input impedance	100			kΩ	between RI, RIB and Polar
CMC	Differential input capacitance relative to frame ground			10 pF		
CMD	Differential input capacitance relative to the other differential input			10 pF		
OFF	Offset	– 30	0	+ 30	mV	includes voltage and current
HYS	Hysteresis	100		400	mV	should be centred relative to filter reference voltage
TEDL	Time delay between input and output at 0			200	ns	for an overdrive input of 50 mV
TEDH	Time delay between input and output at 1			200	ns	
DTEO	Divergence between delay times			100	ns	
TM	Rise time			50	ns	
TD	Fall time			50	ns	CL = 50 pF
DTMO	Divergence between rise and fall times			25	ns	
RRMC	Common-mode rejection					$f = 10$ MHz

8.3.2.3 Characteristics of polarization voltage generator

Symbol	Parameters	Value				Conditions
		min.	typ.	max.	unit	
ZGT	Output impedance				Ω	at 1 MHz
					Ω	DC
VGT	Output voltage				V	at 5 V power supply
DVGT	Precision on VGT				%	

8.3.2.4 Consumption of circuit performing this function

In receive mode, this should be less than 25 mA.

In transmit mode, this should be less than ILD + 25 mA.

8.3.2.5 Input/output characteristics

Name	Type	Function	Mode	Compatibility
RI	Input	+ Input of reception comparators		
RIB	Input	− Input of reception comparators		
POL	Output	Output of filter polarization generator		
DH1	Output	Output of "High Side" current generator 1 mA		
DL1	Output	Output of "Low Side" current generator 50 mA		
DH2	Output	Output of "High Side" current generator 50 mA		
DL2	Output	Output of "Low Side" current generator 1 mA		
Ajcour	Output	Reference current output		
RO	Output	Data output		TTL or CMOS
DE	Input	Data input		pull up
Reset	Input	Input "1" on this input sets the comparator to 1		CMOS pulldown/10 k

8.3.3 Electrical characteristics of filter

See figure 50.

8.3.4 Electrical characteristics of connector

Symbol	Parameter	Value				Conditions
		min.	typ.	max.	unit	
EOD	Spacing between data pins		5		mm	
Rcon	Contact resistance				Ω	
Icon	Current of signals passing through the contact	0,5		60	mA	
Vcon	Voltage of signals passing through the contact		3		V	

8.3.5 Characteristics of transmission medium

Symbol	Parameter	Value				Conditions
		min.	typ.	max.	unit	
Lm	Medium inductance per metre			0,5	µH	
Cm	Medium capacitance per metre		30		pF	
Iso	Insulation				V	

9 Conformance

9.1 Conformance at MAC layer level

The conformance rules expressed in this paragraph are static conformance rules resulting from the various options available at the physical layer level:

Choice of coding: MANCHESTER-L or ENHANCED MANCHESTER.

Once the physical level options have been selected, for an implementation to be in conformance it must comply with all the values shown in this part of ISO 11519 concerning the options selected (values relating to coding, for example).

The recommended access method parameter values for this part of ISO 11519 are specified in 6.3.5.

9.2 Conformance at physical layer level: Line transmitter/receiver tests

The various tests are shown in figures 52 to 57.

These tests allow checking of:

— differential voltage;

— common-mode voltage variations;

— rise and fall times;

— recessive and dominant currents.

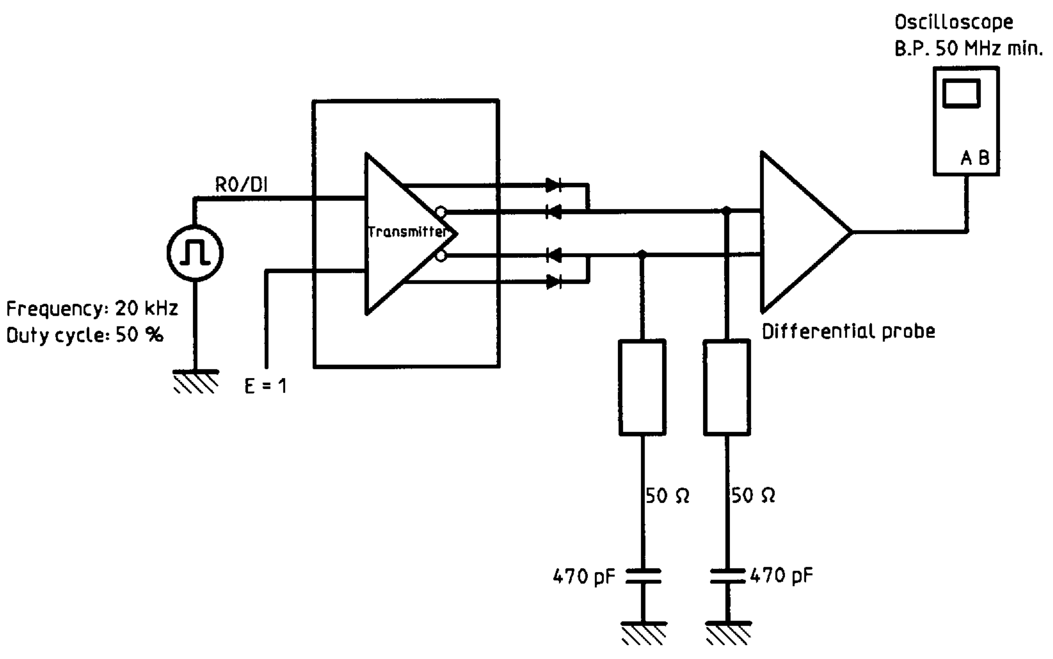


Figure 52 — Differential voltage measurement

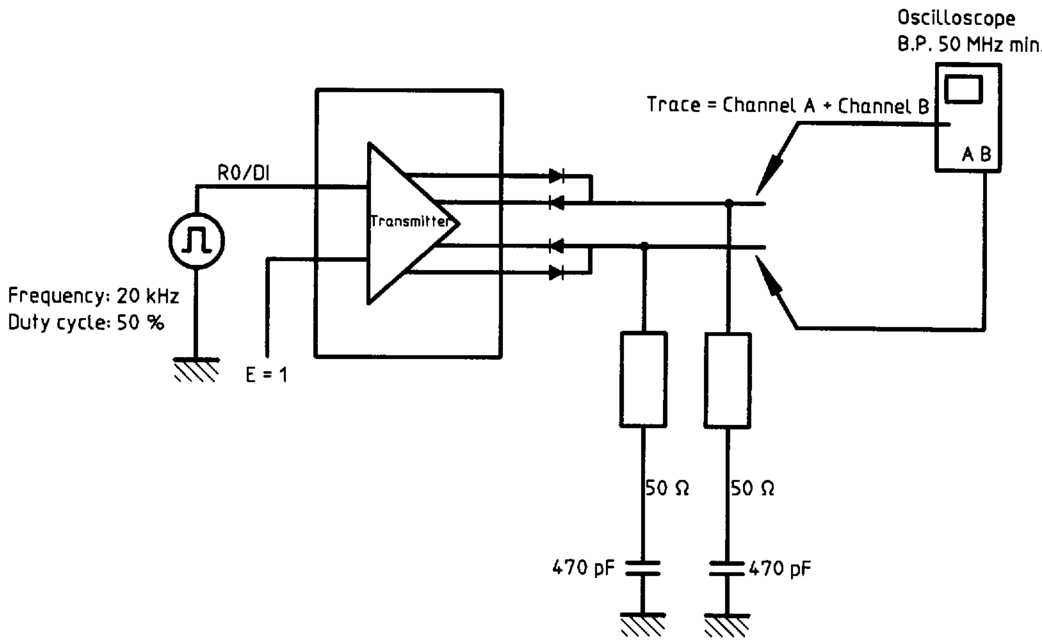


Figure 53 — Measurement of common-mode voltage variation

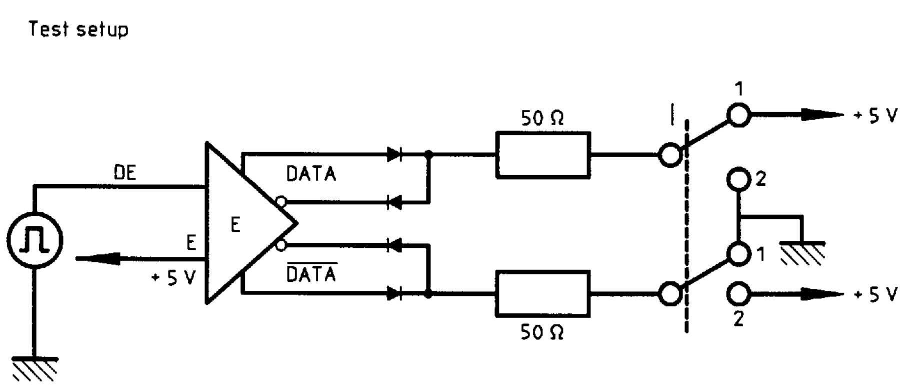


Figure 54 — Measurement of T1 and T2

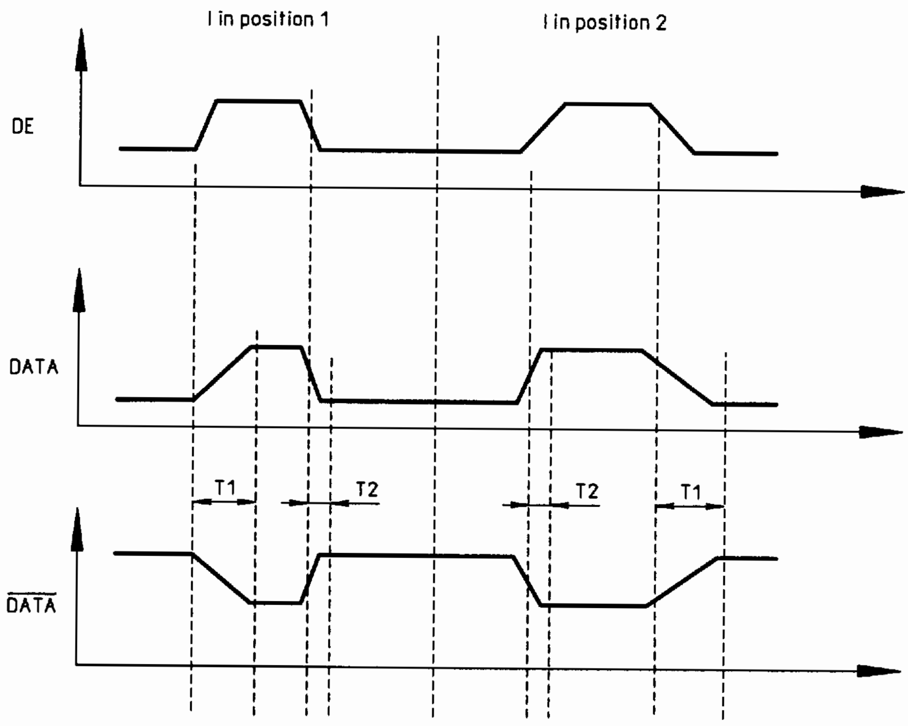


Figure 55 — Timing diagram for figure 54

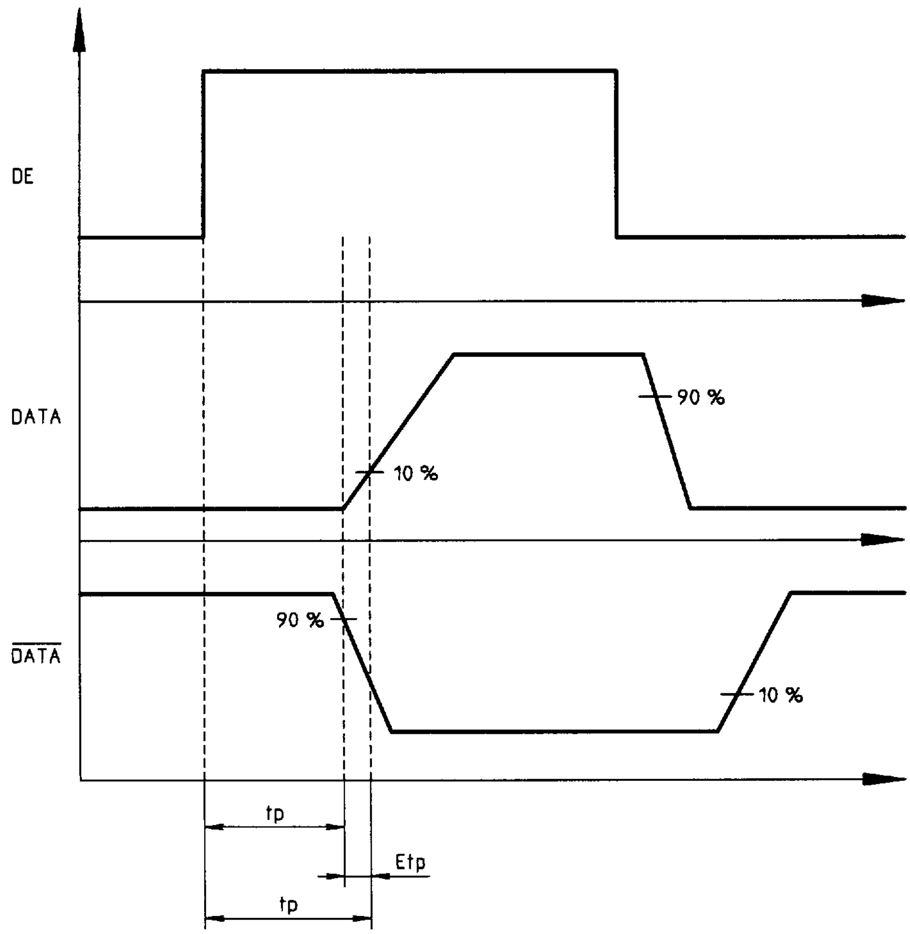
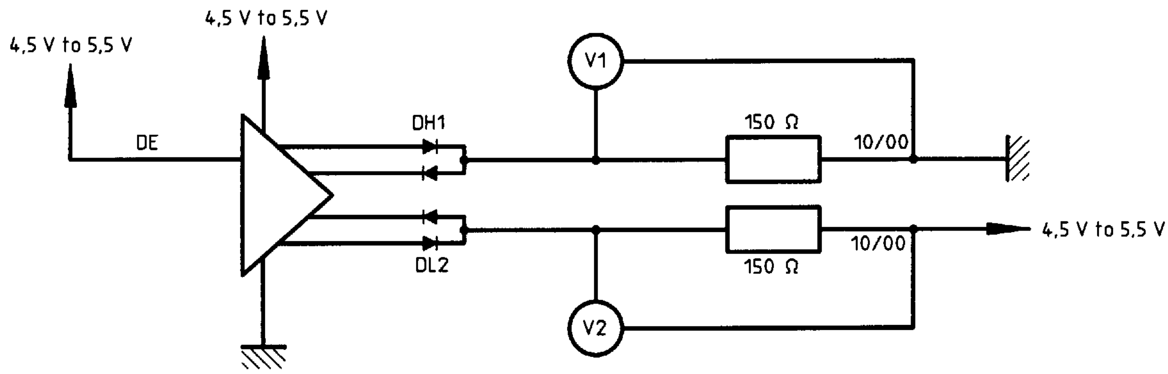
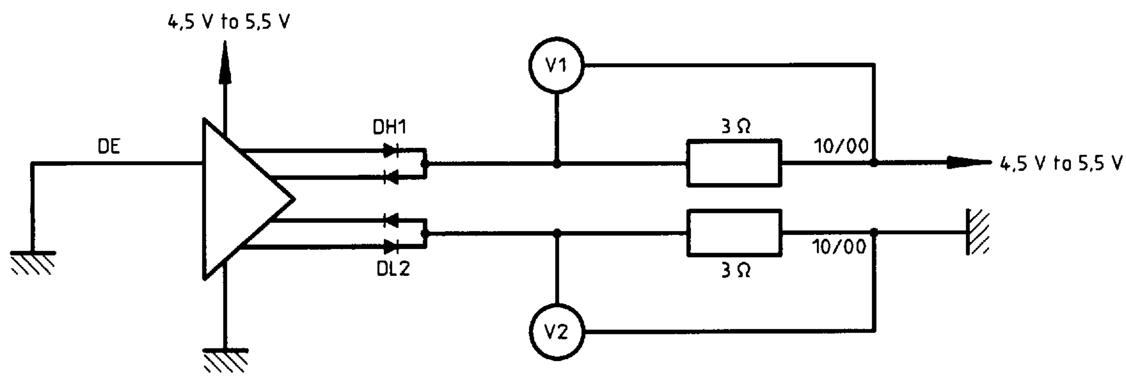


Figure 56 — Determination of propagation times



Measurement of 1 mA current



Measurement of 50 mA current

Figure 57 — Current measurement method

Annex A
(normative)

Setup example for Baud Rate Multiplier

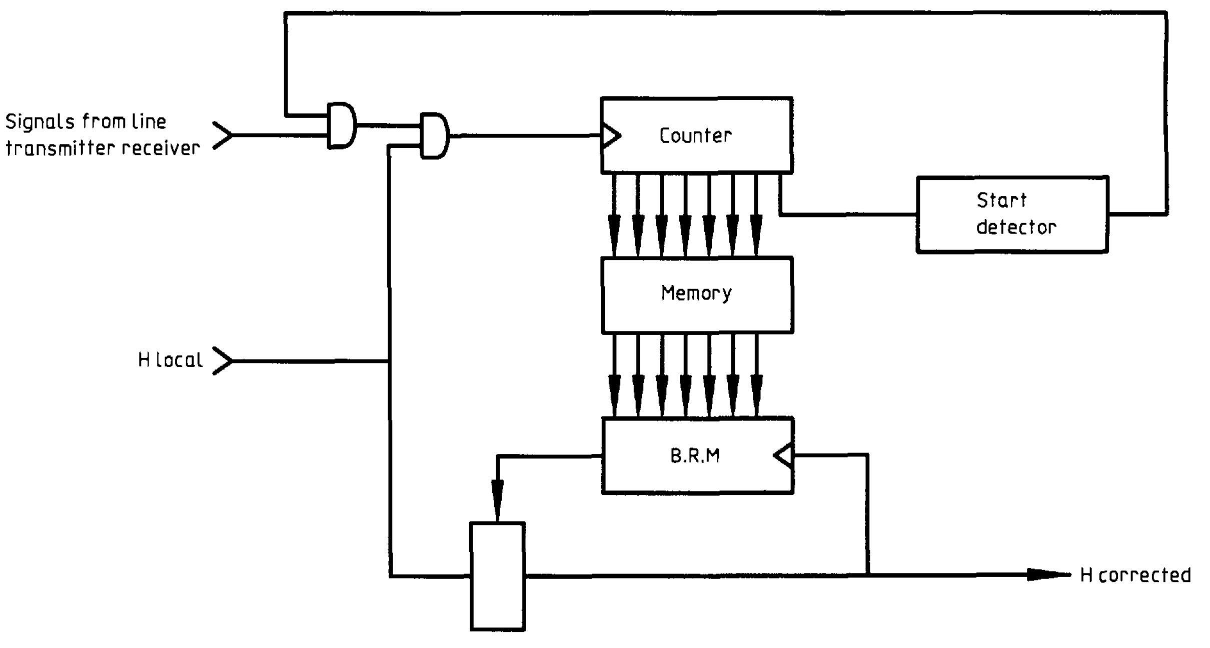


Figure A.1 — Baud Rate Multiplier

Annex B
(normative)

Setup example of realization of interface between physical layer and data link layer

B.1 Interface position/OSI model

In order to simplify the hardware of the communication interface, a new sublayer structure has been defined. This new substructure does not strictly correspond to the model used in this part of ISO 11519.

The example of the realization of the interface layer between physical and MAC layers described in this annex corresponds to the breakdown of layers in accordance with figure B.1.

The MAC sublayer contains the access mechanism to the physical layer which assures the transfer between the two LLC entities.

The physical sublayer directly connected to the MAC sublayer performs the electric encoding and decoding of bits and symbols (characters).

B.2 Definition of dialogue signals between binary and electric encoding/decoding sublayers

Figure B.2 illustrates the breakdown of layers and gives the various signals allowing interfacing of the different sublayers.

As each symbol corresponds to an integer number of binary elements and as the bit time unit is equal to two time slots (T.S.) for the Manchester-L code (see figures 30 and 31) and 1 T.S. for the Enhanced Manchester code (see figures 30 to 32), each character is coded for each time unit which corresponds to one binary element with the signals classified in figure B.2 and defined in table B.1.

B.3 Coding bit and symbol (see 7.2.3)

The PL2 physical sublayer uses the binary information bit to transmit (BT) and transmission code violation (TCV) to generate the various characters of different fields, and uses the bit received (BR), reception code violation (RCV) and collision detection (CD) signals to read the different received characters.

In case of the Enhanced Manchester code, the physical layer automatically inserts every 3 NRZ bits a Manchester-L bit from the start of the identifier fields (see 6.2) until the End Of Data field inclusive.

From the ACK field until the next SOF, the PL1 physical sublayer follows the code defined in table B.2.

Figures B.3 and B.4 describe the specific dialogue between the physical layer and the data link layer for the SOF.

The Hsyn clock signal presents the same temporal characteristics as the Hsymb. (See figures 32 and 33.)

Tables B.2 and B.3 define the correspondence between the dialogue signals of the physical layer and the data link layer.

B.4 Character synchronization

The synchronization of characters are based on the same synchronization rules regardless of the bit or the binary value of symbol.

The electrical synchronization is based on the rules described in 7.2.4.2.

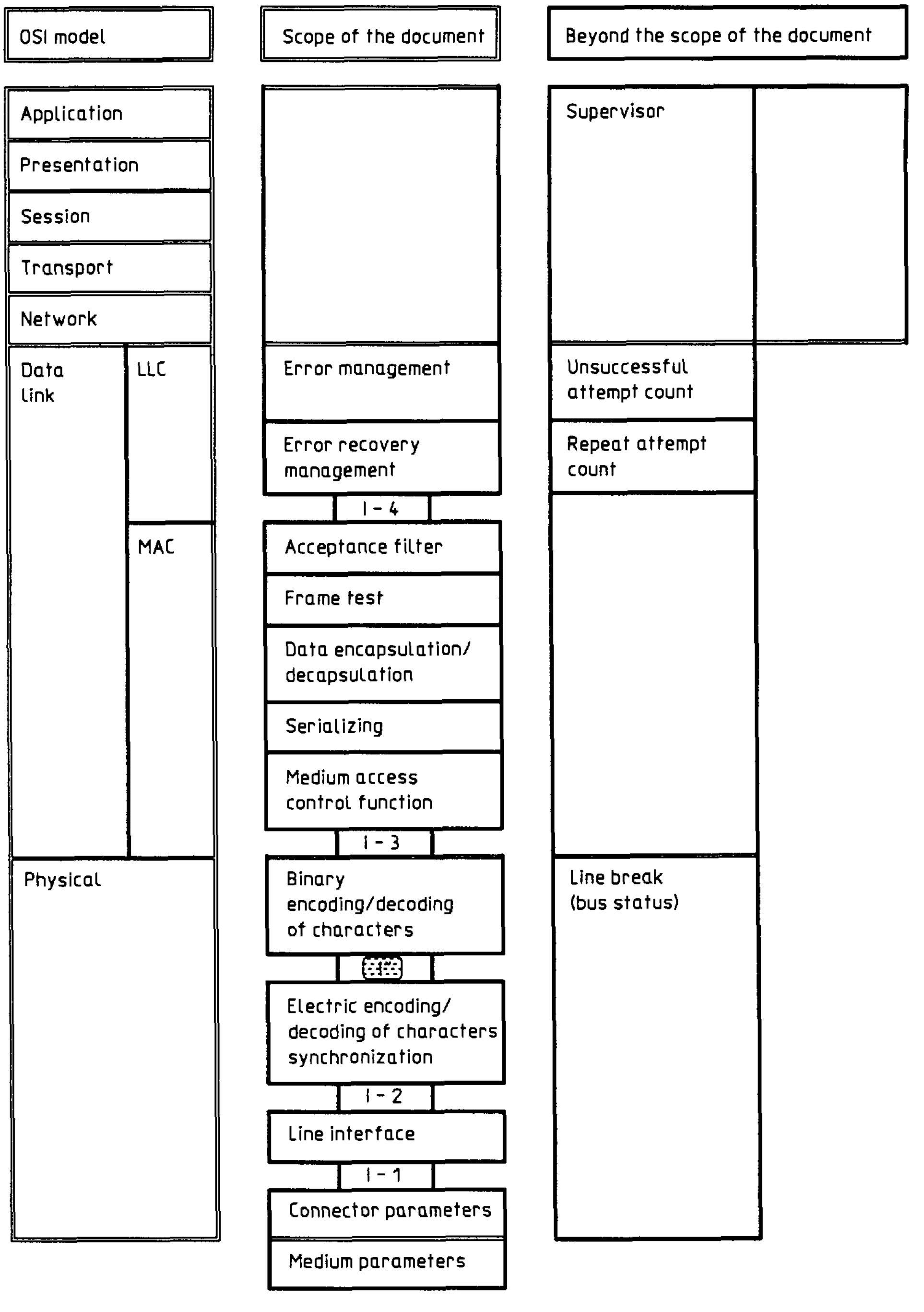


Figure B.1 — Breakdown by layers in accordance with OSI model

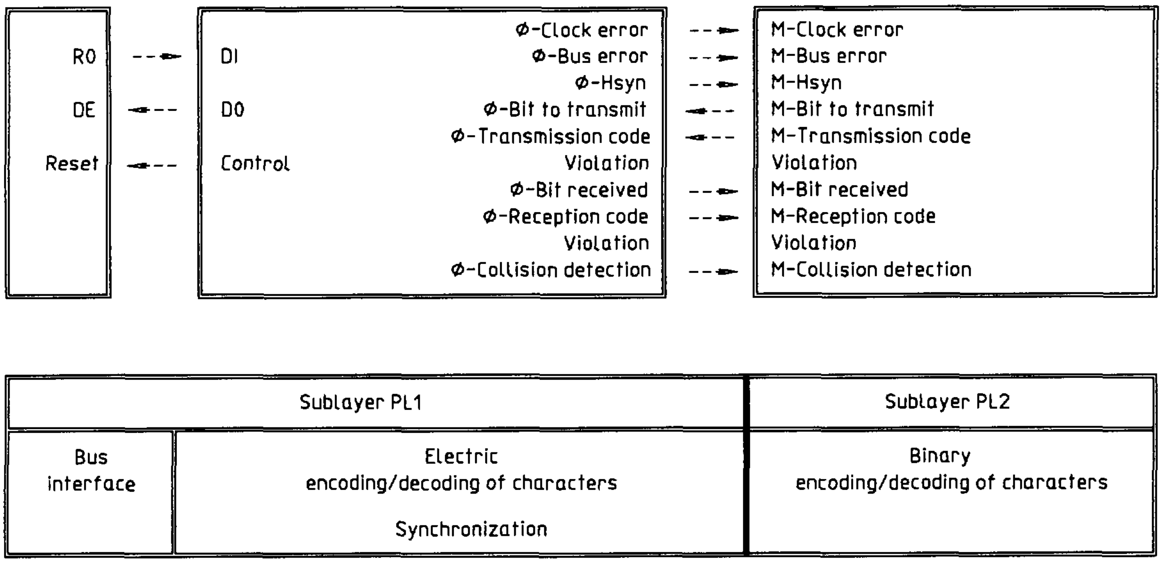
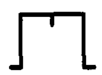


Figure B.2 — Interconnections between different sublayers

Table B.1 — Dialogue signals between sublayers PL1 and PL2

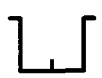
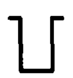
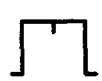
Name	Direction	Definition
Clock error	PL1 to PL2	Signal indicating a clock corrector overflow.
Bus error	PL1 to PL2	Signal indicating the presence of a dominant level of longer duration than the Start Of Frame symbol.
Hsyn	PL1 to PL2	Synchronized clock which enables PL2 to generate characters and to recognize the characters received.
Bit to transmit	PL2 to PL1	Binary value of next bit or (part of) symbol to transmit.
Transmission code	PL2 to PL1	Signal which permits disabling and enabling of electric coding of each binary value to be sent.
Bit received	PL1 to PL2	Binary value of the bit or (part of) symbol read on the bus.
Reception code violation	PL1 to PL2	Signal indicating a rule violation in the corresponding bit code (Manchester-L or Enhanced Manchester).
Collision detection	PL1 to PL2	Signal, updated at each received bit, which indicates that a collision has occurred (transmit bit different than received bit).

Table B.2 — Physical layer

Emission	BT	TCV			Signal on bus
Reception			BR	RCV	Manchester-L
	0	1	0	1	R D  Dominant bit uncoding
	0	0	0	0	R D  Dominant bit coding
	1	1	1	1	R D  Recessive bit uncoding
	1	0	1	0	R D  Recessive bit coding

BT: Bit to Transmit
TCV: Transmission Code Violation
BR: Bit Received
RCV: Reception Code Violation
R: Recessive level
D: Dominant level

Table B.3 — Data link layer

Emission	BT	TCV			Signal on bus
Reception			BR	RCV	Enhanced Manchester
	0	1	0	1	R D  Dominant bit uncoding
	0	0	0	0	R D  NRZ — Dominant bit coding R D  Manchester
	1	1	1	1	R D  Recessive bit uncoding
	1	0	1	0	R D  NRZ — Recessive bit coding R D  Manchester

BT: Bit to Transmit
TCV: Transmission Code Violation
BR: Bit Received
RCV: Reception Code Violation
R: Recessive level
D: Dominant level

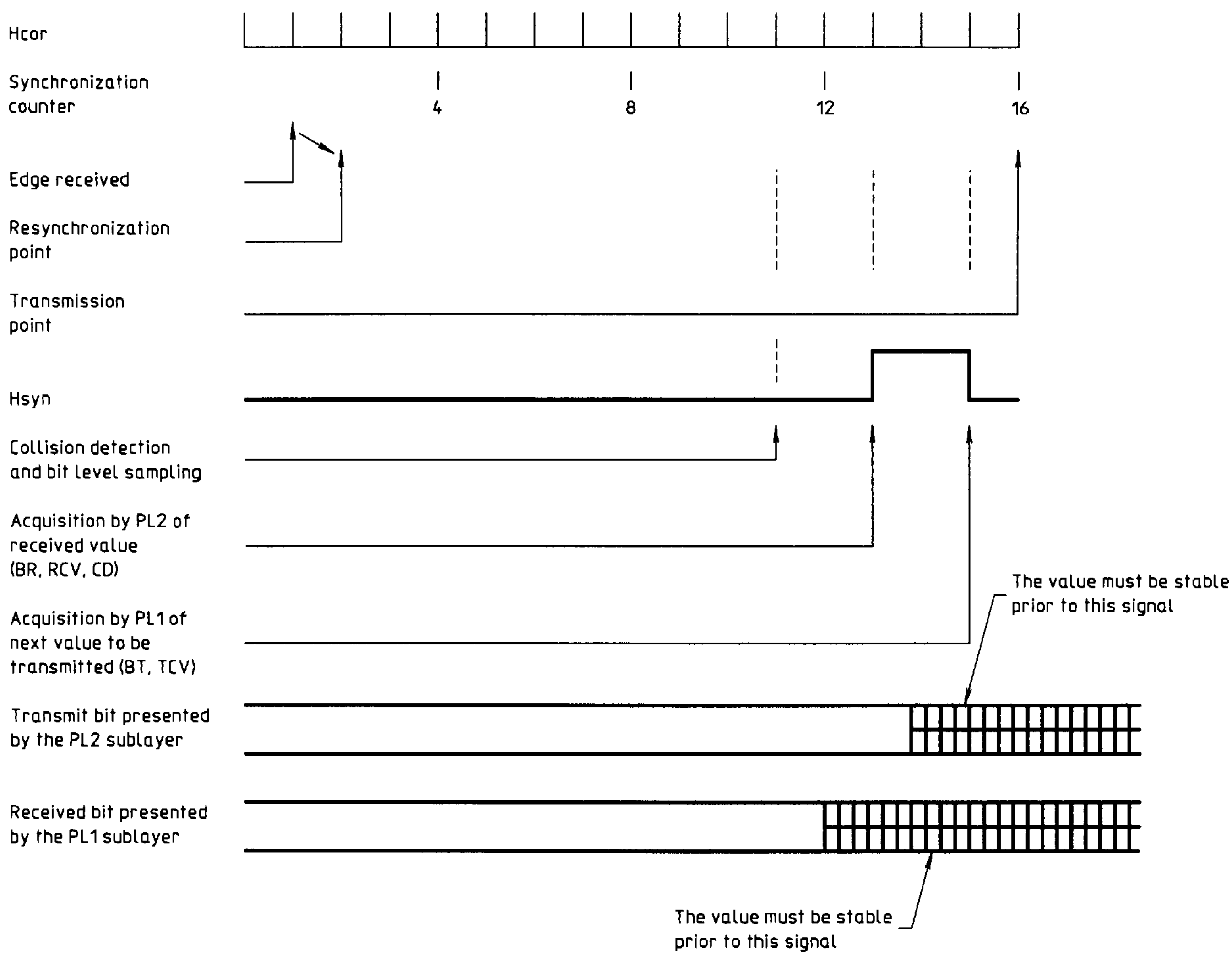


Figure B.3 — Bit synchronization: Bit NRZ

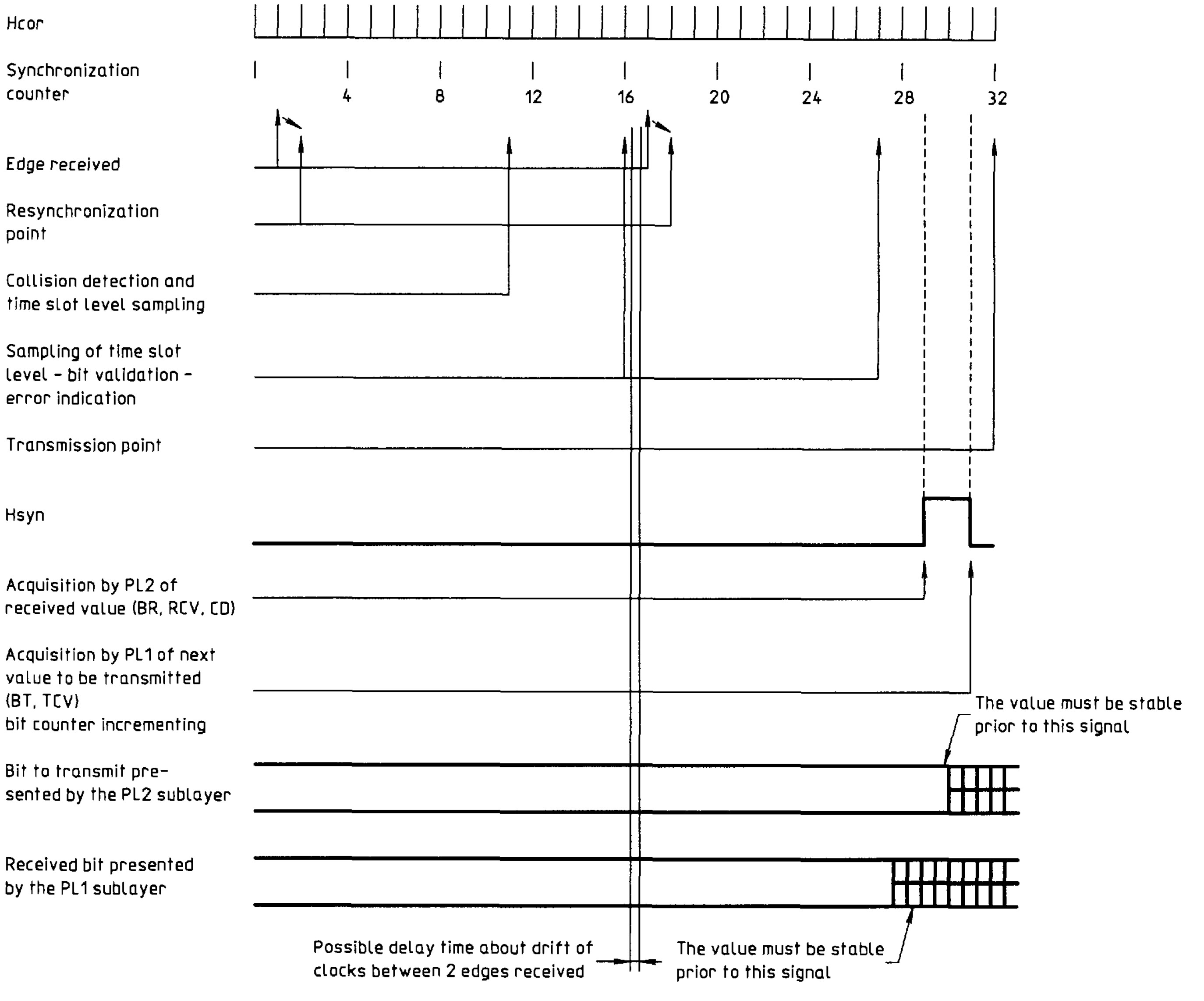


Figure B.4 — Bit synchronization: Bit Manchester

B.5 Signals exchange rules between PL1 and PL2 sublayers

The rules governing the exchanges between PL1 and PL2 sublayers are based on the use of the Hsyn clock:

— The acquisition of the signals from the PL2 sublayer to the PL1 sublayer must be made on the rising edge of Hsyn.

— At the interface, these signals must be present at least half a corrected clock period (Hcor) before the synchronized clock (Hsyn) appears.

— The acquisition of the signals from the PL1 sublayer to the PL2 sublayer must be made on the falling edge of Hsyn.

— At the interface, these signals must be present at least half a corrected clock period (Hcor) before the synchronized clock (Hsyn) appears. See figure B.5.

B.6 Exceptions to this signal exchange rules

Two signals can be acquired or generated asynchronously:

— the clock error signal,

— the bus error signal.

The time duration must be at least superior to the synchronized clock period (H syn).

B.7 Timing diagram for various dialogue signals between PL1 and PL2 sublayers

See figure B.6.

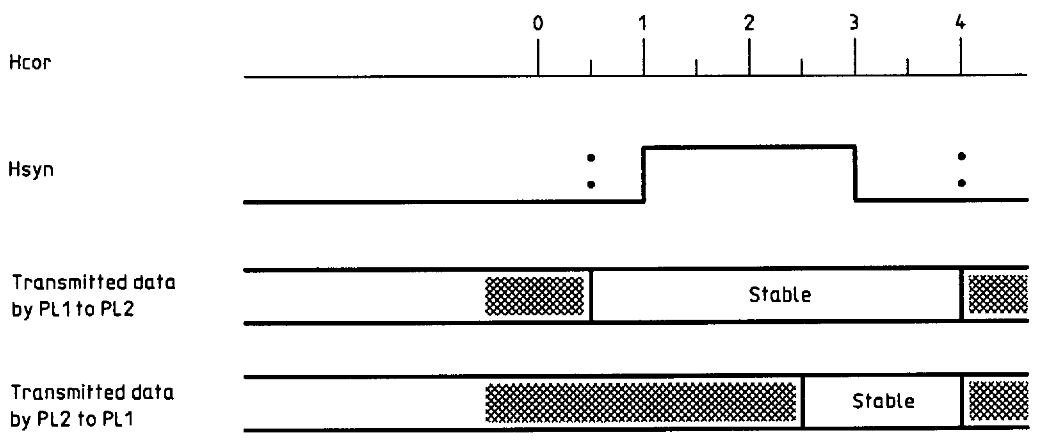


Figure B.5 — Information exchange rule between PL1 and PL2 sublayer

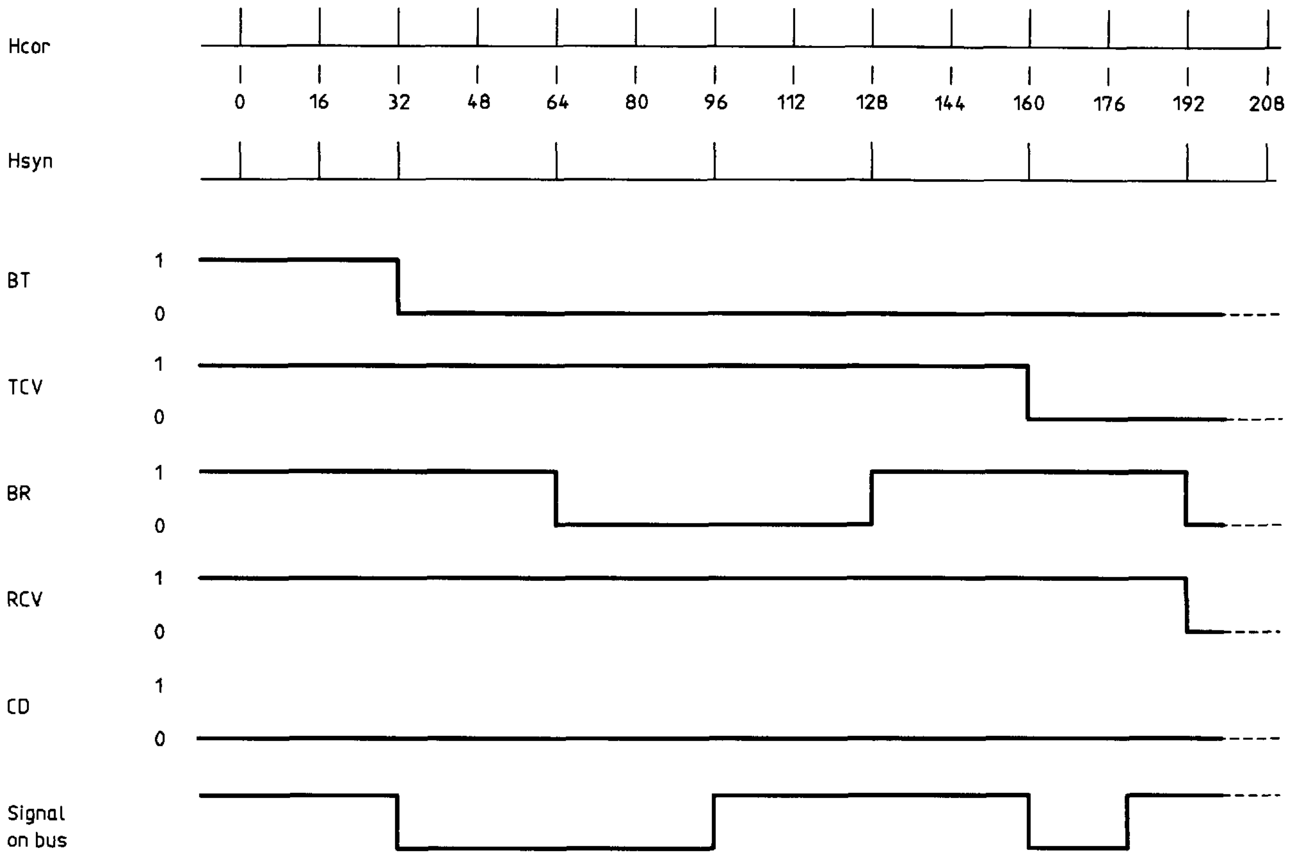


Figure B.6 — Dialogue for generation of SOF in Enhanced Manchester

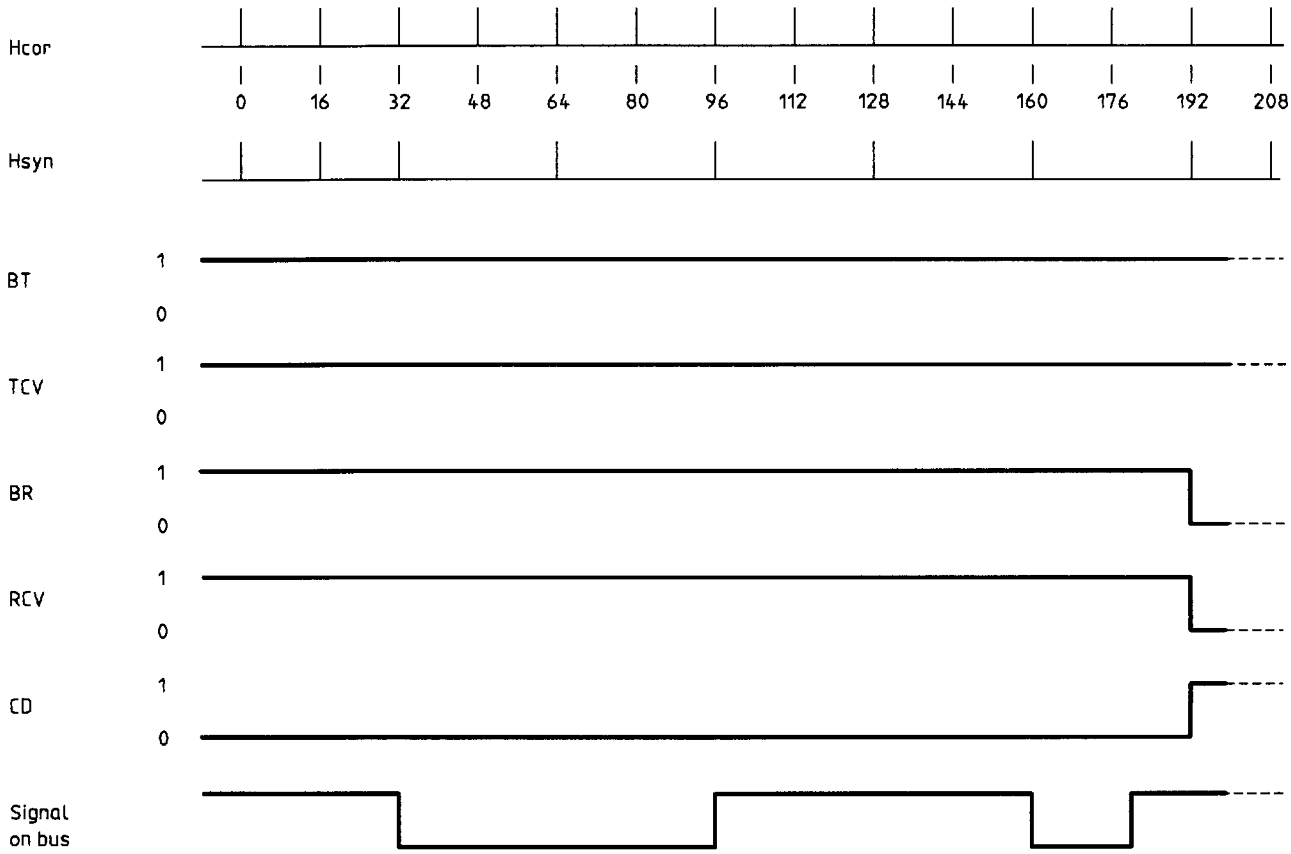


Figure B.7 — Dialogue for signalling reception of SOF in Enhanced Manchester

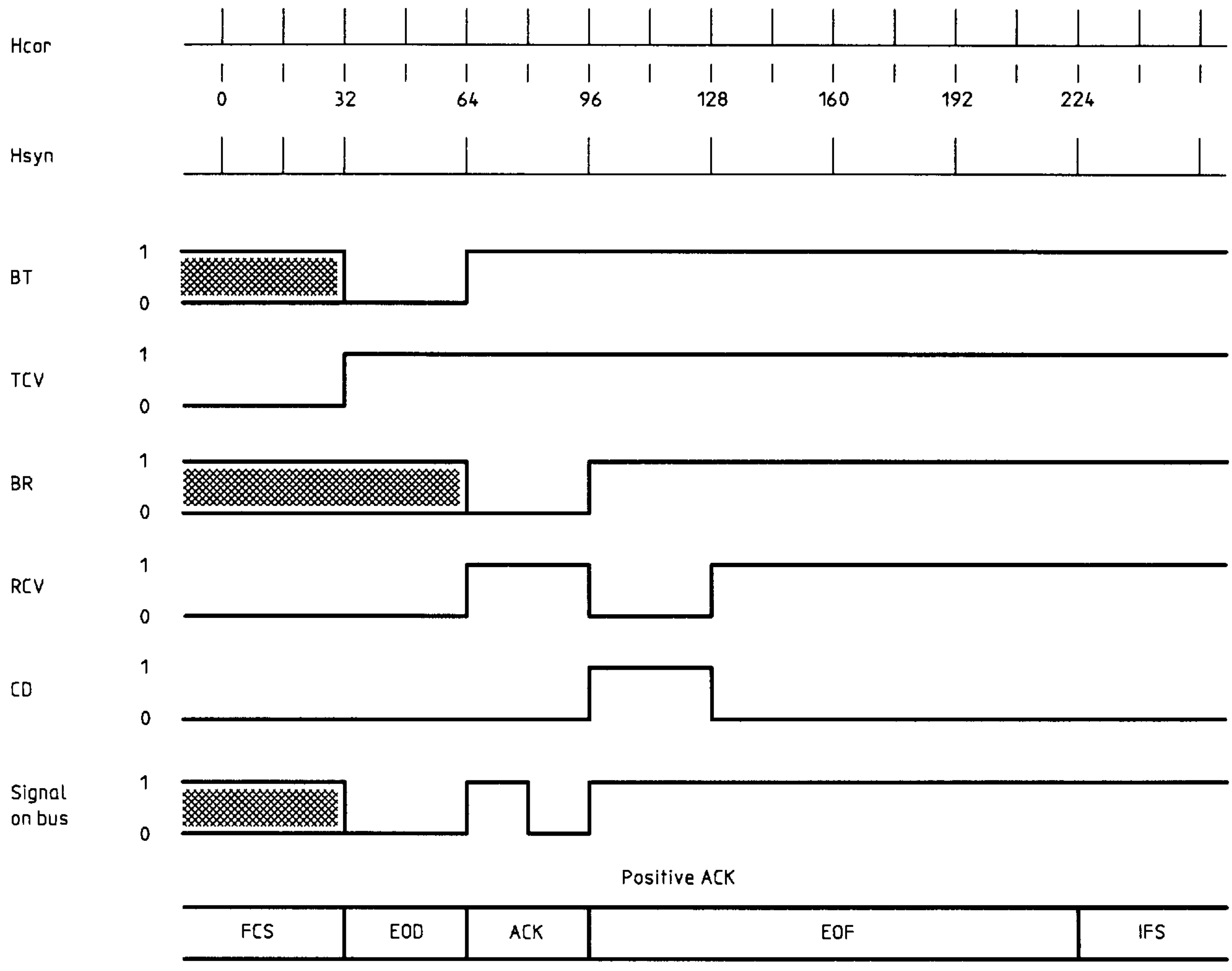


Figure B.8 — Dialogue for generation of End Of Frame with positive acknowledgement reception in Enhanced Manchester

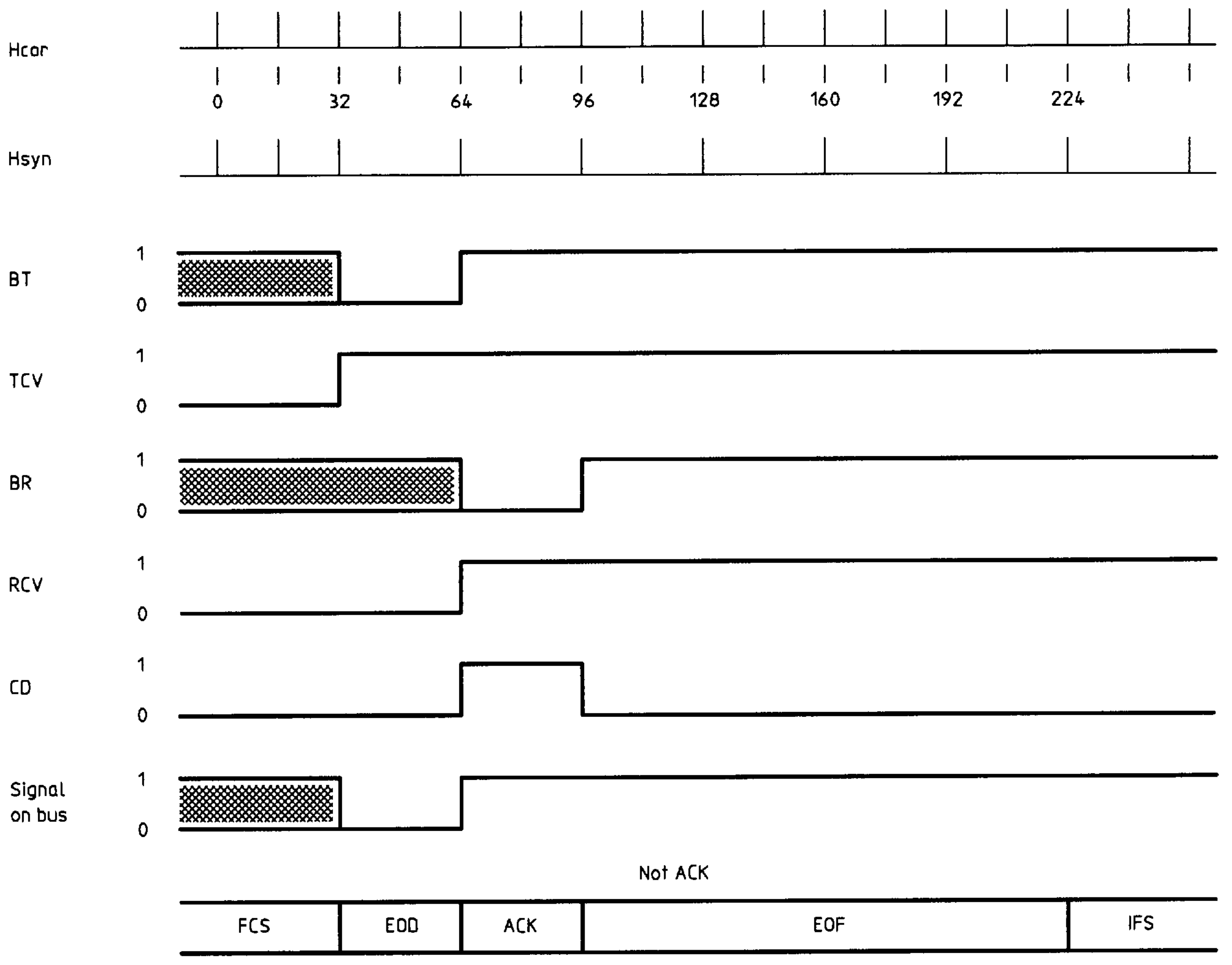


Figure B.9 — Dialogue for generation of End Of Frame without positive acknowledgement reception in Enhanced Manchester

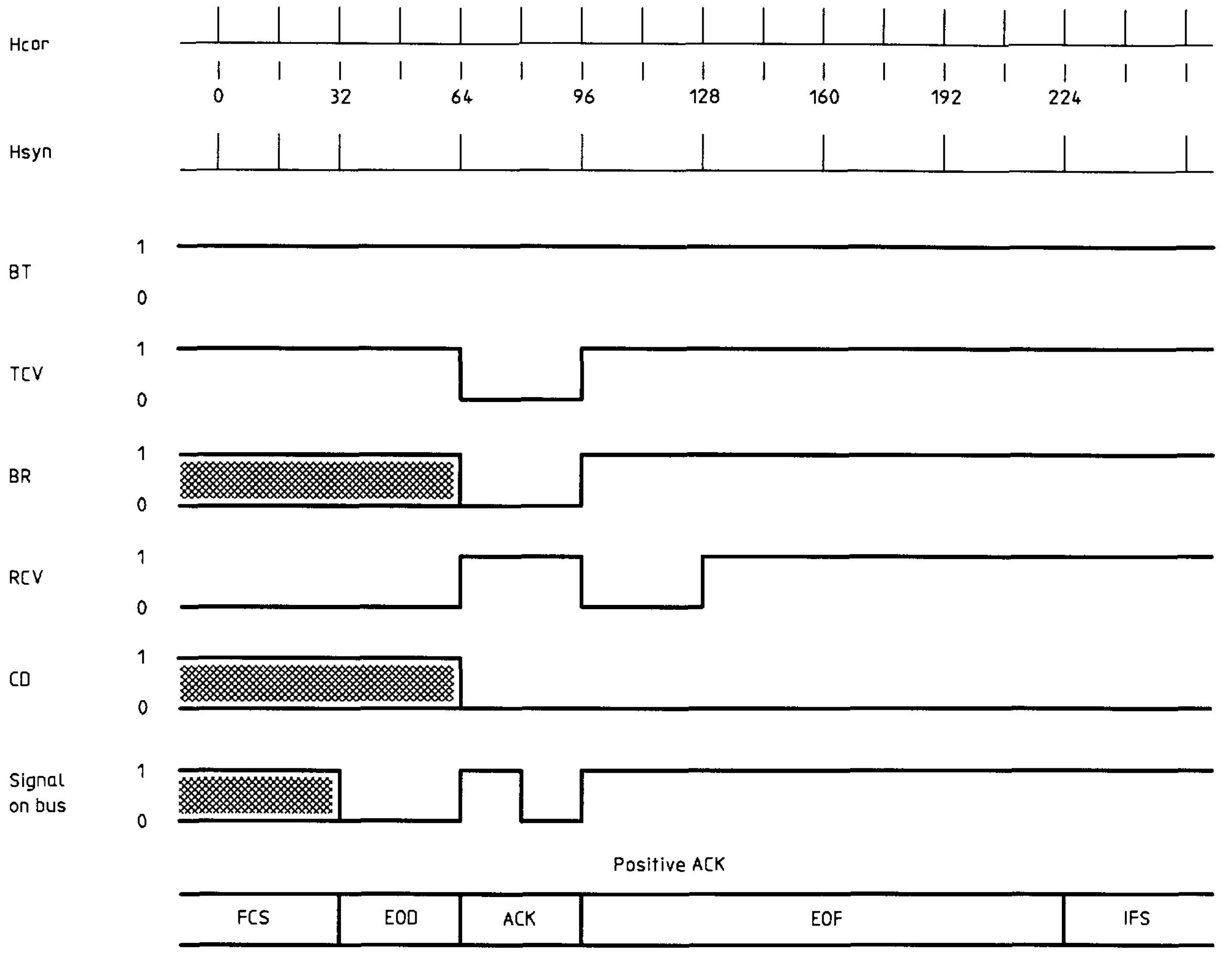


Figure B.10 — Dialogue for signalling reception of End Of Frame with positive acknowledgement reception in Enhanced Manchester

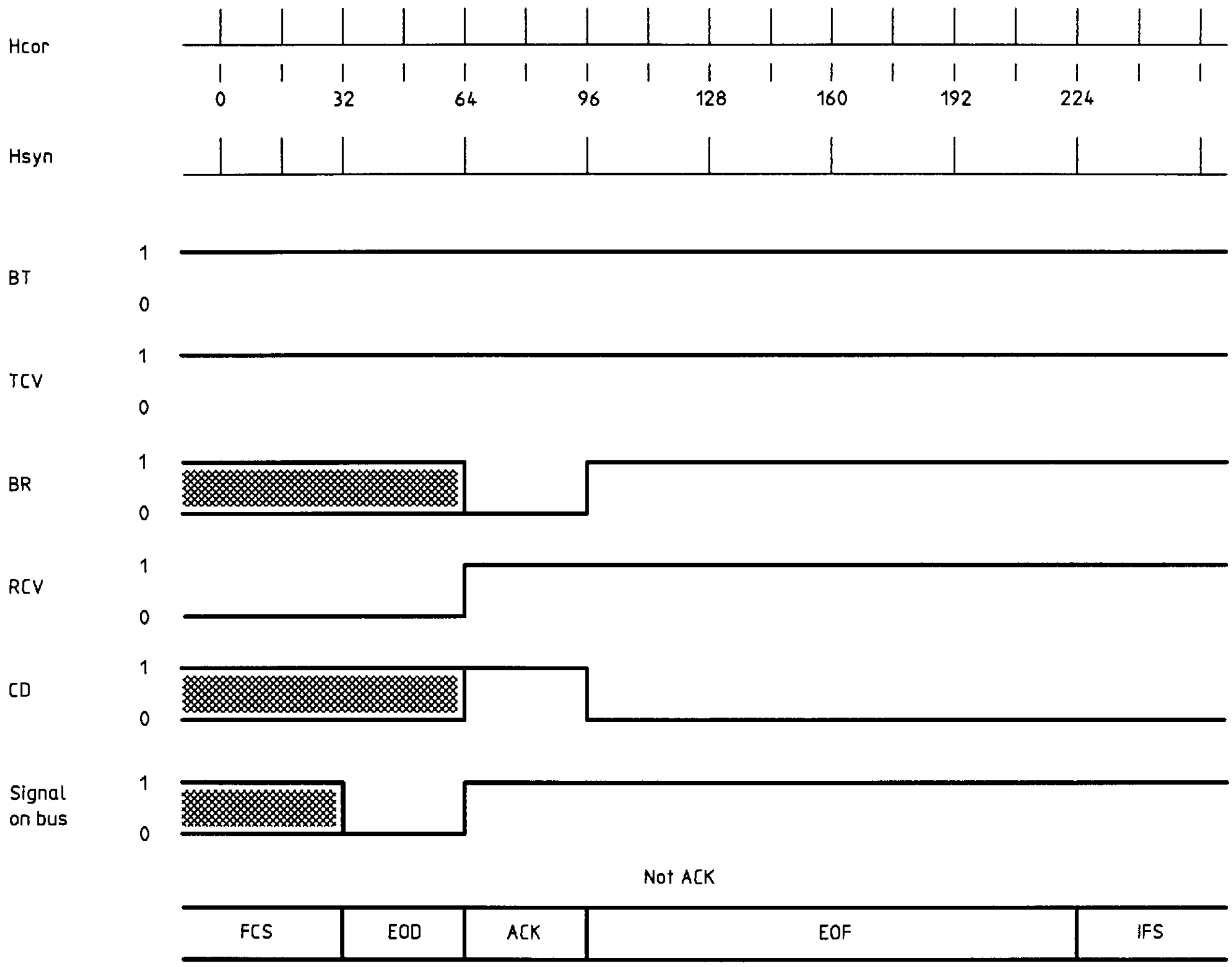


Figure B.11 — Dialogue for signalling reception of End Of Frame without positive acknowledgement reception in Enhanced Manchester

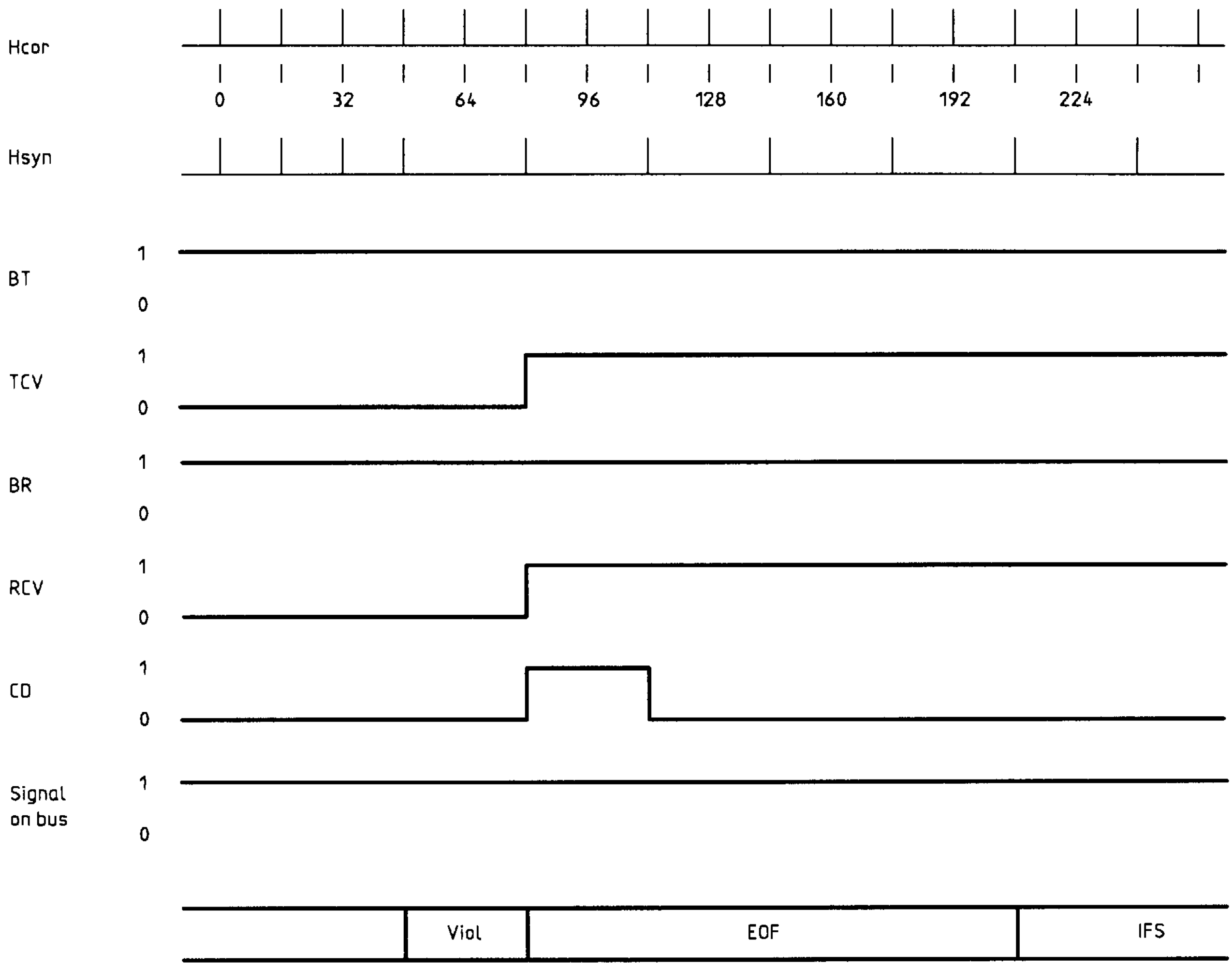


Figure B.12 — Dialogue when violation of code occurs in Enhanced Manchester

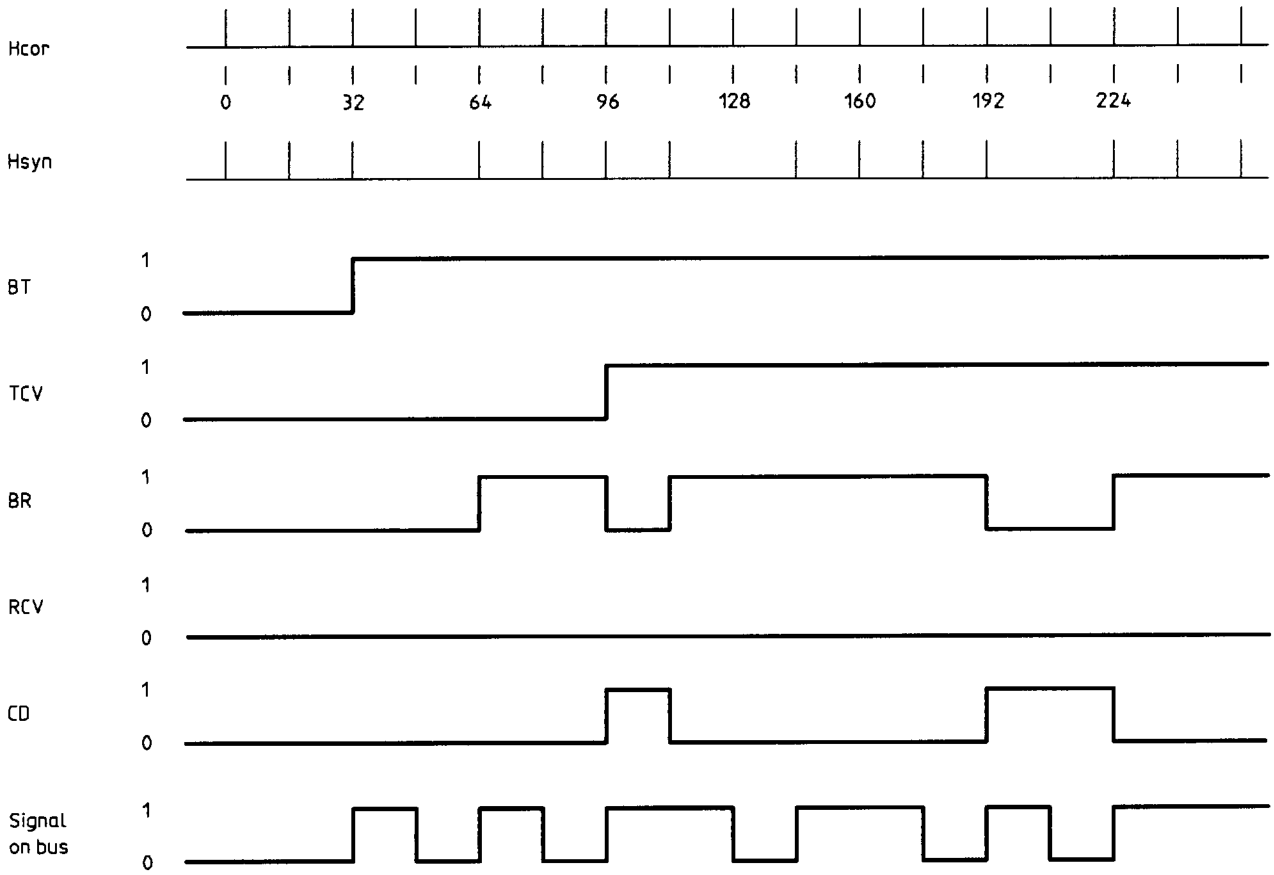


Figure B.13 — Dialogue when collision detection occurs in Enhanced Manchester

ICS 43.040.10

Descriptors: road vehicles, electronic equipment, data communication equipment, data processing, information interchange, local area networks, vehicle area networks, data transmission, data link layer, physical layer.

Price based on 104 pages