

Manual for serials and parallel interfaces



image 9030

image 9020

Small character ink-jet printing



A DOVER COMPANY



Contents

General Connection	7
■ Serial links - Introduction	9
■ Connection	10
■ Wiring diagrams	11
Voltage transmission wiring diagrams	12
■ Format and transmission speed	13
■ Electrical specifications	13
General principle of an exchange	15
■ General principle of an exchange	17
Data sending from the computer to the printer	17
Data request from the computer to the printer	17
Sending data from the printer to the computer	18
Transmission protocole	18
■ Identification (1 byte in hexadecimal)	19
■ Length (2 bytes in hexadecimal)	19
■ Data	19
■ Checksum	19
■ Management of the dialog and faults	20
In Reception	20
In Transmission	20
Fault	20
List of identification	21
■ Sendings	23
■ Requests	24
■ List of the commands	25



Contents

Protocols - Sending 27

■ Sending concerning the printer	29
Stop/Start of the jet or the printer	29
Acknowledgement of faults	30
Command of the languages of the month to be printed	30
Sending request of the print acknowledgement by the computer and sending of the print acknowledgement by the printer	32
Sending the request for non doubled printing	33
■ Sending concerning the messages	34
Sending a message for printing	34
Sending a partial message for printing	35
Sending a message selected according to its number	36
Sending a message to the library	36
Sending cancellation of a message	38
Print sending	39
■ Sending of the variables	40
Initialization of the autodating	40
Sending the autodating parameters	40
Sending of external variables	43
Sending an autodating table	44
Counters initialization	45

Protocols - Request 47

■ Request concerning the printer	49
Request printer status	49
Request of faults	51
Request for the history of the last thirty faults	54
Request for status of the tacho pulse number/fly time offset output	56
Request for software version	57
Request for functions	58
Request for types of character generators available in the printer	60
■ Requests concerning messages	61
Request for general parameters by default	61
Request for a message by number	62
Request for a free message number	63
Request for the number of the active message	64



Contents

■ Request concerning variables	65
Request for autdating parameters	65
Request for the autdating	69
Request for an autdating table	70
Request for the current value of the autdating elements	72
Request for the languages of the month to be printed	75
Request for the current value of the counters	76

Protocols – Precisions / Examples

■ Message data details	81
Structure indicator	81
General parameters of the message	82
Variable element parameters	84
Definition of lines	90
Definition of blocks	90
Detail of the text	92
End of message	102
■ Examples	103
Sending a complete message	103
Request for a message active number	107
Sending a partial message	108

Appendix 1

■ Reminder	111
Binary, decimal, hexadecimal conversion	111
■ ASCII table of characters and symbols used for a message	112
■ Fonts list	113
Standard fonts	113
CurvePrint fonts	115
■ Algorithmes list	115
■ Performance	116
■ Result	117

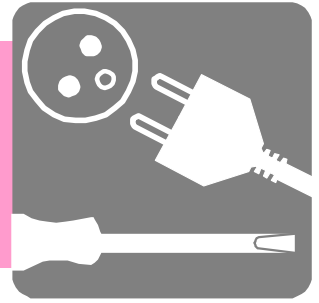


Contents

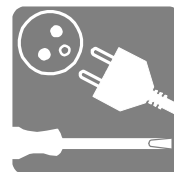
Appendix 2 119

Message selection by Parallel interface 125

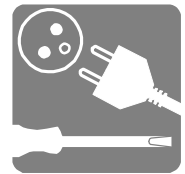
■ External interface - Introduction	127
Configuration	127
Characteristics	127
■ Connection	128
Connection of the parallel interface to a SIGEA head movement	130
■ Message selection by parallel interface	131
Operation	131
■ Signals Diagram	132
■ Message selection by message increment interface	133
Parallel Interface Mode	133
Signals Diagram	134
Message Increment Mode	135



General Connection



General - Connection



General - Connection

This manual describes the serial links on 9020 and 9030 printers and the parallel link on the 9030 printer.

The printer manual should be consulted for normal use and for programming the printer.

■ Serial links - Introduction

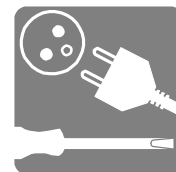
The printers have an asynchronous, serial interfaces, allowing a V24/RS232C or RS422 mode, "voltage level".

The serial links are located on the industrial interface board.

REMINDER	<i>RS422 transmission has excellent immunity to electrical and electromagnetic interference. It is therefore particularly recommended for long links or if interference is high.</i>
-----------------	--

The RS232C/V24 or RS422 standard describes the signals available during a dialog between DTE (Data Terminal Equipment) and DCE (Data Communication Equipment).

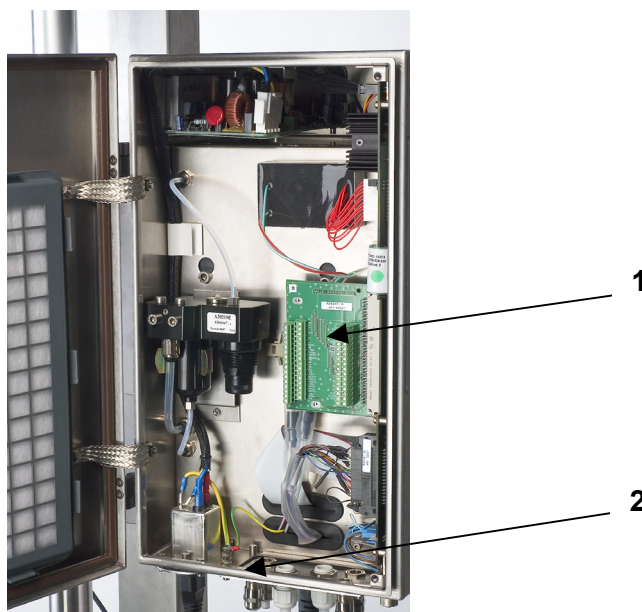
NOTE	<i>The printer is DTE.</i>
-------------	----------------------------



■ Connection

The connection of the serial link to the industrial interface card (1) is made as follows:

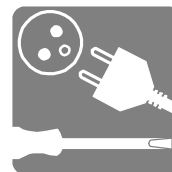
- Press  to stop the printer.
- Disconnect the printer from the mains.
- Open the left door on the printer.



- Pass the shielded link cable through the stuffing boxes (2) under the printer.
- Attach the wires to their corresponding connector terminals J4 (3) according to the type of link used.

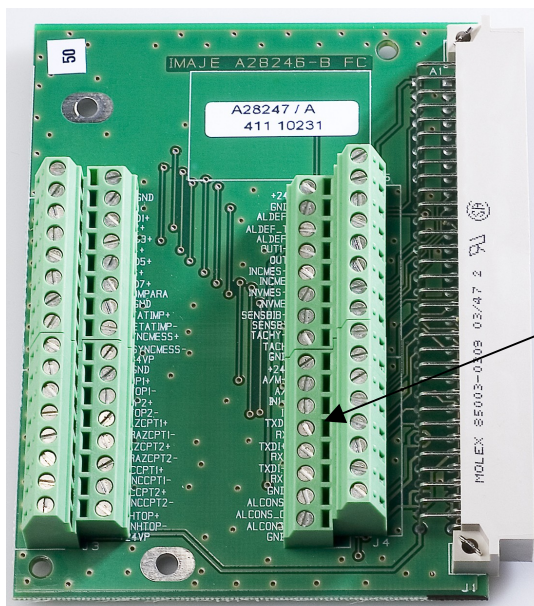
IMPORTANT

The shielding of the link wire must be connector throughout the full length of the metallic stuffing box of the printer, and the same type of connection must be made on the computer.



General - Connection

View of the industrial interface board



■ Wiring diagrams

The transmission links are wired between terminal J4 of the industrial interface card and the terminals of the industrial computer or PC.

TXDI: Data V24/RS232C

TXDI +: Data RS 422

TXDI -:
(equivalent to TD: Transmit Data)

Data transmission line

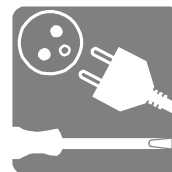
RXDI: Data V24/RS232C

RXDI +: Data RS 422

RXDI -:
(equivalent to RD: Receive Data)

Data reception line

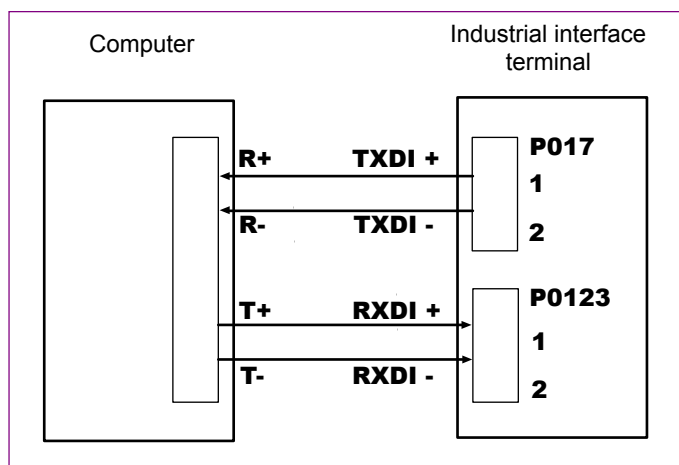
GND: (Ground) Electrical Common



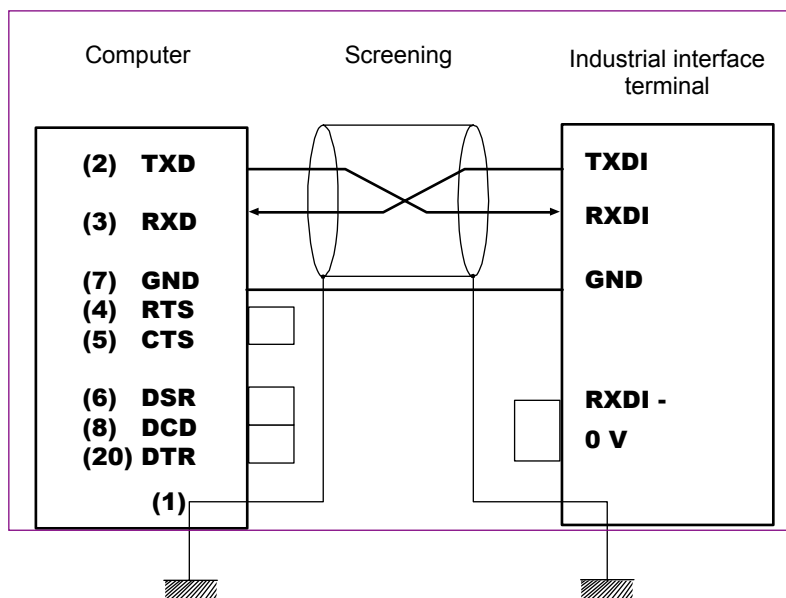
General - Connection

Voltage transmission wiring diagrams

■ RS422 Link



■ RS232/V24 Link





General - Connection

■ Format and transmission speed

The speed and format for the serial link are configured in the menu

► *Printer preparation/Initialization/Printer/Communication.*

The data are coded in hexadecimal unless otherwise indicated.

All the data coded on several bytes must be transmitted with the high order byte first. All bytes must be transmitted with the least significant bit first.

Three possible formats:

- 1 start bit, 8 bits, no parity, 1 stop bit,
- 1 start bit, 8 bits, no parity, 2 stop bits,
- 1 start bit, 8 bits, even number, 1 stop bit.

Speed transmission (in bauds):

- 38400
- 19200
- 9600
- 7200
- 4800
- 3600
- 2400
- 1200
- 600
- 300
- 150

■ Electrical specifications

The electrical specifications are governed by the V24, RS232C standard or the RS422 standard.



General principle of an exchange



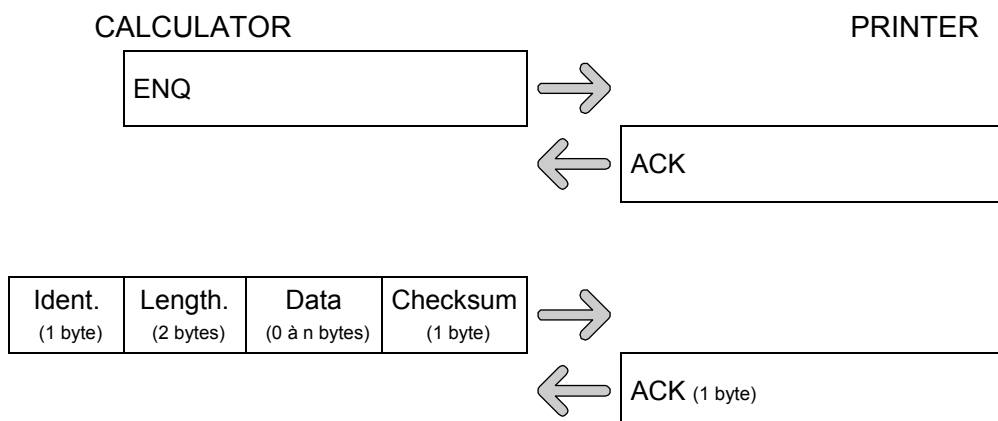
General principle of an exchange



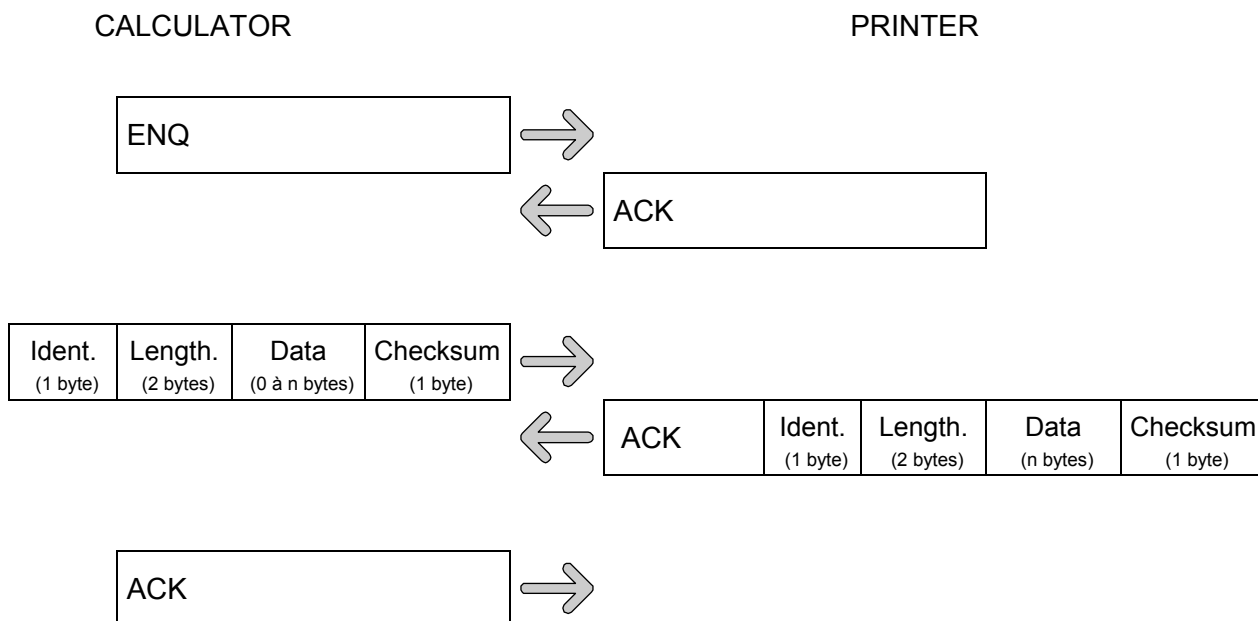
General principle of an exchange

■ General principle of an exchange

Data sending from the computer to the printer



Data request from the computer to the printer



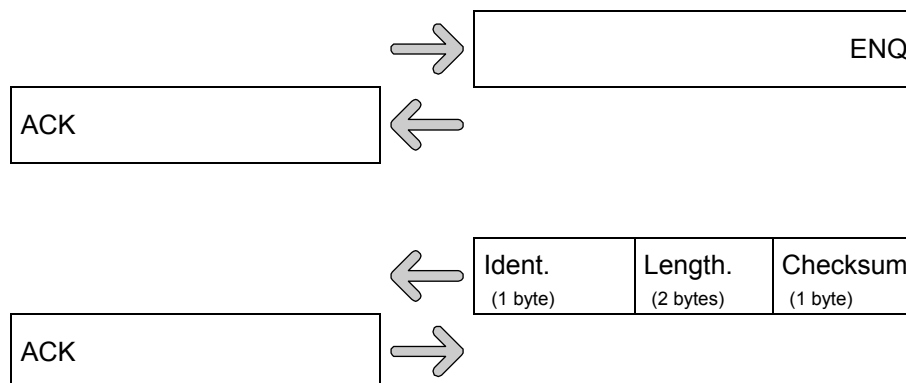


General principle of an exchange

Sending data from the printer to the computer

CALCULATOR

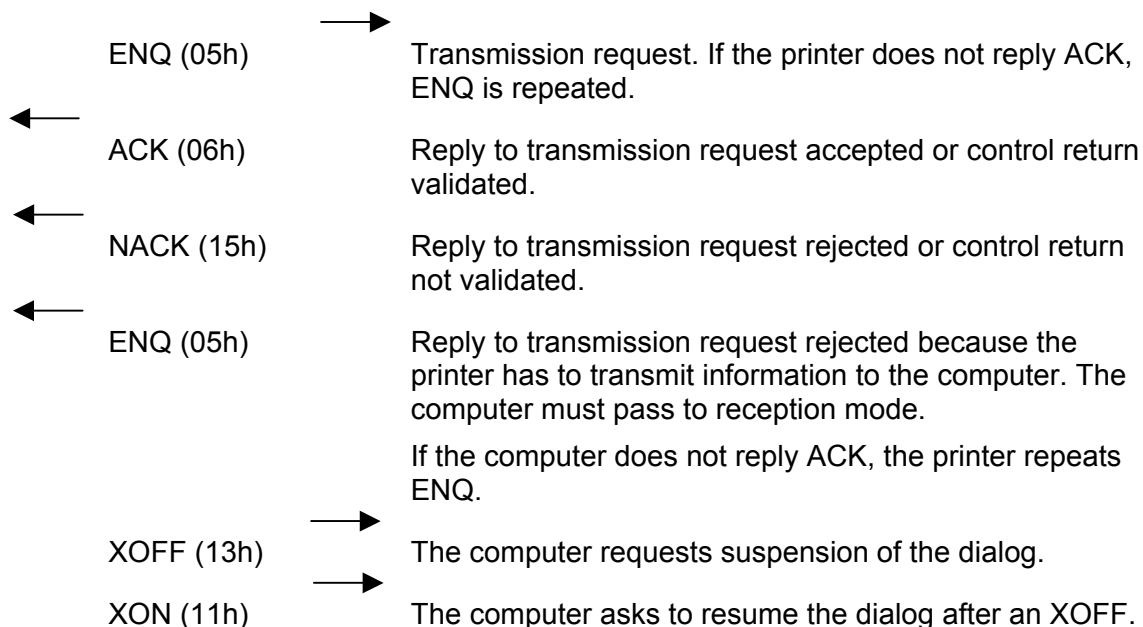
PRINTER



NOTE

This case exists only for sending the print acknowledgement.

Transmission protocole





General principle of an exchange

■ Identification (1 byte in hexadecimal)

Specific for each command (see tables following pages).

■ Length (2 bytes in hexadecimal)

The length is the number of bytes comprised between the length and the checksum byte.

The maximum value is in general 2044 bytes or 07h FCh.

In case of a sent to the printing, the maximum value is: 4092 bytes or 0Fh, FCh.

■ Data

Zero byte in case of request from the calculator to the printer.

n bytes representing the instructions required for defining a function.

■ Checksum

This represents an "OR exclusive" of all preceding bytes (identification, length and data bytes).



General principle of an exchange

■ Management of the dialog and faults

In Reception

The printer can receive orders from the computer at any time.

In Transmission

If the computer cannot receive it transmits an XOFF (13h) code, the printer suspends transmission and awaits the XON (11H) signal to resume. If the printer receives the XON signal after more than two seconds it cancels the transmission, returns a NACK and the "time out external communication" fault appears.

IMPORTANT *The maximum answering time for the printer, without taking the line transfer into account, is 5 ms.*

Fault

■ "TIME OUT external communication" fault

During initialization of the dialog:

When the printer sends an ACK or an ENQ, if the computer does not reply within two seconds the printer stops the dialog, sends a NACK and indicates "TIME OUT external communication" fault.

During the dialog:

- If the computer sending time between two bytes exceeds two seconds, the printer stops the dialog, returns a NACK and indicates "TIME OUT external communication" fault.
- If the time between the computer XOFF and XON sending exceeds two seconds, the printer stops the dialog, returns a NACK and indicates "TIME OUT external communication" fault.
- If at the end of the printer sending the computer does not reply within two seconds the printer stops the dialog, returns a NACK and indicates "TIME OUT external communication" fault.

■ "V24 DATA" fault

If the printer answers NACK at the end of the dialog, the control (checksum) is incorrect, and the message must be resumed from the beginning with ENQ. If after three trials resumption is impossible or if command identification is unknown the printer indicates "Incorrect external data received" fault.



List of identification



List of identification



List of identification

■ Sendings

TITLE OF THE COMMANDS	IDENT. COMMAND	DENT. ANSWER
Stop/Start of the jet or the printer	C6h	
Acknowledgement of faults	E6h	
Sending a message to the library	9Fh	C5h
Sending a message for printing	E3h	
Sending a partial message for printing	E5h	
Sending the request for non doubled printing	E9h	
Sending a message selected according to its number	98h	
Print sending	94h	
Sending cancellation of a message	C7h	C5h
Sending an autodating table	DFh	
Counters initialization	97h	
Sending of external variables	99h	
Initialization of the autodating	C8h	
Sending the autodating parameters	ECh	
Languages of the month to be printed	C4h	
Sending request of the print acknowledgement by the computer	D8h	
Sending of the print acknowledgement by the printer	CEh	
Configuration of "ETAT" output	B8h	



List of identification

■ Requests

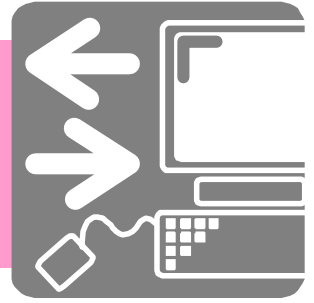
TITLE OF THE COMMANDS	IDENT. COMMAND	DENT. ANSWER
Request printer status	A6h	E4h
Request of faults	DAh	D2h
Request for the history of the last thirty faults	CFh	D0h
Request for status of the tacho pulse number/fly time offset output	B7h	B8h
Request for software version	B1h	B2h
Request for functions	ADh	B0h
Request for types of character generators available in the printer	A4h	A5h
Request for general parameters by default	CBh	CDh
Request for a message by number	92h	9Fh
Request for a free message number	A2h	A3h
Request for the number of the active message	DBh	91h
Request for autodating parameters	BBh	BCh
Request for the autodating	D6h	9Ch
Request for an autodating table	DEh	DFh
Request for the current value of the autodating elements	A9h	ABh
Request for the languages of the month to be printed	E0h	C4h
Request for the current value of the counters	95h	96h



List of identification

■ List of the commands

IDENT.	TITLE OF THE COMMANDS
92h	Request for a message by number
94h	Print sending
95h	Request for the current value of the counters
97h	Counters initialization
98h	Sending a message selected according to its number
99h	Sending of external variables
9Fh	Sending a message to the library
A2h	Request for a free message number
A4h	Request for types of character generators available in the printer
A6h	Request printer status
A9h	Request for the current value of the autodating elements
ADh	Request for functions
B1h	Request for functions
B7h	Request for status of the tachometer pulse number/fly time offset output
B8h	Configuration of ETAT output
BBh	Request for autodating parameters
C4h	Languages of the month to be printed
C6h	Stop/Start of the jet or the printer
C7h	Sending cancellation of a message
C8h	Initialization of the autodating
CBh	Request for general parameters by default
CEh	Sending of the print acknowledgement by the printer
CFh	Request for the history of the last thirty faults
D6h	Request for the autodating
D8h	Sending request of the print acknowledgement by the computer
DAh	Request of faults
DBh	Request for the number of the active message
DEh	Request for an autodating table
DFh	Sending an autodating table
E0h	Request for the languages of the month to be printed
E3h	Sending a message for printing
E5h	Sending a partial message for printing
E6h	Acknowledgement of faults
E9h	Sending the request for non doubled printing
ECh	Sending the autodating parameters



Protocols - Sending



Protocols - Sending

Protocols - Sending



This section and following deal with all the commands, transmissions or requests, for dialoging with the printer.

CAUTION For the sake of clarity, the transmission protocol elements ENQ, ACK, etc. are not indicated - see the section "General Dialog Principles".

■ Sending concerning the printer

Stop/Start of the jet or the printer

CALCULATOR		PRINTER	
			
Identification (1 byte)		C6h	
Length (2 bytes)		00h, 01h	
Data: - state of the jet and of the machine (1 byte)		xxh	
Checksum (1 byte)		xxh	

■ Detail of the data byte

- 00h Jet stop
- 01h Jet start
- 08h Machine stop
- 0Ch Machine restart during stop phase
- 0Dh Machine restart at end of stop phase or of maintenance function or in stand by state with power on.



Acknowledgement of faults

CALCULATOR

PRINTER



Identification (1 byte)	E6h
Length (2 bytes)	00h, 00h
Checksum (1 byte)	xxh

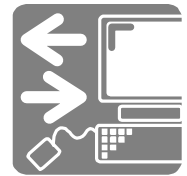
Command of the languages of the month to be printed

CALCULATOR

PRINTER



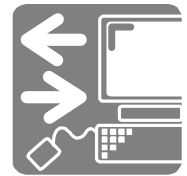
Identification (1 byte)	C4h
Length (2 bytes)	00h, 02h
Data: - number of the timer language 1 (1 byte) - number of the timer language 2 (1 byte)	xxh xxh
Checksum (1 byte)	xxh



■ Detail of the data bytes (for the language 1 and 2)

Number	Language
00h	French
01h	English
02h	German
03h	Italian
04h	Spanish
05h	Norwegian
06h	Swedish
07h	Dutch
08h	Danish
09h	Japanese (Kanji)
0Ah	Portuguese
0Bh	Brasilian
0Ch	Hungarian
0Dh	Polish
0Eh	Turkish
0Fh	Czech
10h	Arab
11h	Chinese
12h	Korean
13h	Hebrew
14h	Russian
15h	Thai
16h	Greek
17h	Finnish
18h	Croatian
19h	Slovenian
1Ah	Vietnamese
1Bh	Iranian
1Ch	Bulgarian
1Dh	Indonesian
1Eh	Taiwanese
1Fh	Hegirien

Protocols - Sending



Sending request of the print acknowledgement by the computer and sending of the print acknowledgement by the printer

CALCULATOR

PRINTER



Identification (1 byte)	D8h
Length (2 bytes)	00h, 00h
Checksum (1 byte)	D8h

Sending is to be made once prior to the start of production, and remains effective until the printer comes to a stop. The printer sends a print acknowledgement at the end of the printing.

CALCULATOR

PRINTER



CEh	Identification (1 byte)
00h, 00h	Length (2 byte)
CEh	Checksum (1 byte)

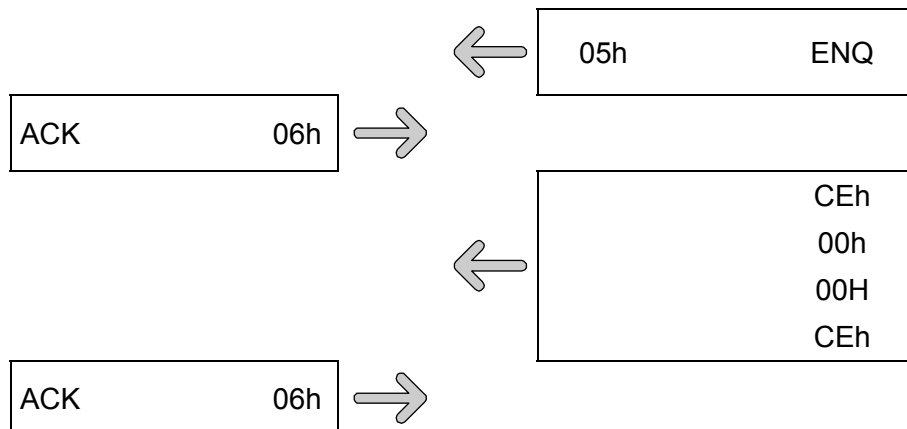
NOTE

In case of a conflicting dialog at this moment, the printer has the priority.

Example:

CALCULATOR

PRINTER



Protocols - Sending



Sending the request for non doubled printing

CALCULATOR

PRINTER



Identification (1 byte)	E9h
Length (2 bytes)	00h, 01h
Data: - type (1 byte)	xxh
Checksum (1 byte)	xxh

This sending is made once prior to the start of production. Non doubled printing can be disabled without stopping the machine.

■ Data details

Type:

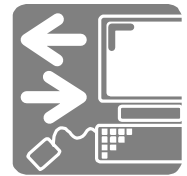
00 unvalidate non doubled printing

01 validate non doubled printing

NOTE

The role of this function is to prevent a message being printed twice in succession, without any modification.

Protocols - Sending



■ Sending concerning the messages

Sending a message for printing

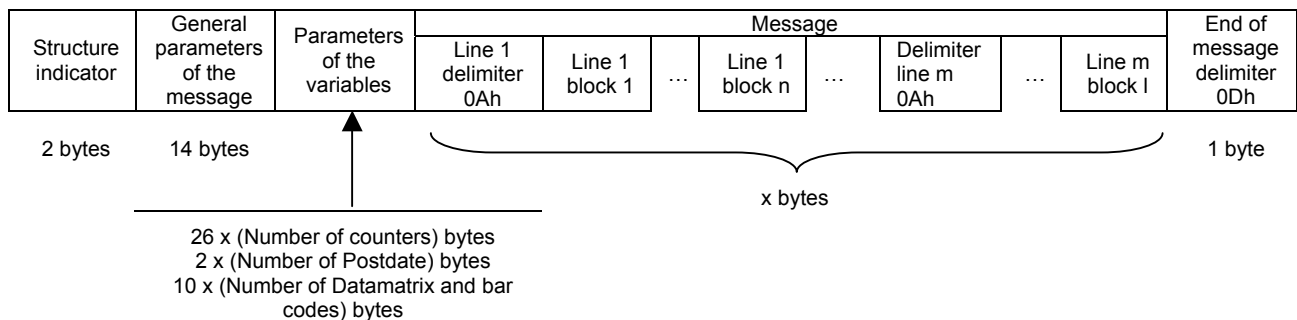
CALCULATOR

PRINTER



Identification (1 byte)	E3h
Length (2 bytes)	xxh, xxh
Data - structure indicator - general parameters - variable element parameters - message to print	xxh - - - xxh
Checksum (1 byte)	xxh

■ Data details



See "Protocols - Precisions/Examples" section

IMPORTANT

The size of the sending, from identifier to control, can be up to 4096 bytes.





Sending a partial message for printing

This sending allows amendment of the active message content on printing without altering its structure.

CALCULATOR

PRINTER



Identification (1 byte)	E5h
Length (2 bytes)	xxh, xxh
Data: - area number to be amended (1 byte) - modification	xxh - - xxh
Checksum (1 byte)	xxh

■ Data details

Number of areas to be amended	Line "k" number (from 0 to 15)	Position of the first characters to be amended in line (from 0 to x)	Number of characters to be amended in the area (from 1 to x)	SYMBOLS	...	Line "l" number (from 0 to 15)	...
1 byte	1 byte	2 bytes	2 bytes	P bytes		1 byte	

An area is a succession of continuous characters of the same block.

For example, there are two possibilities to modify the "I" and the "E" of IMAJE:

I M A J E 2 zones will be modified (Nbre of zones = 02h)

Zone 1 Zone 2

or

I M A J E 1 seul zone sera modifiée (Nbre de zones = 01h)

Zone 1

The line structure is not modified.

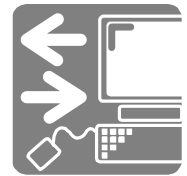
The first line corresponds to the number "0", the first byte after the line delimiter 0Ah corresponds to position "0".

See section "Protocols - Precisions/Examples" for the composition of a message.

IMPORTANT

The size of the sending, from the identifier to control, can be up to 2 kilobytes.

Protocols - Sending



Sending a message selected according to its number

This sending consists in selecting for printing a message contained in the printer library.

CALCULATOR

PRINTER



Identification (1 byte)	98h
Length (2 bytes)	00h, 02h
Data: (2 bytes) - message number (001 to 999)	xxh xxh
Checksum (1 byte)	xxh

Sending a message to the library

CALCULATOR

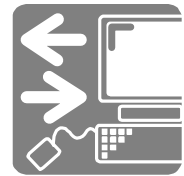
PRINTER



Identification (1 byte)	9Fh
Length (2 bytes)	xxh, xxh
Data: - message number (001 to 999) - message heading, (8 characters ASCII) - message parameters and text (n bytes) - type of write (1 byte)	xxh, xxh
Checksum (1 byte)	xxh



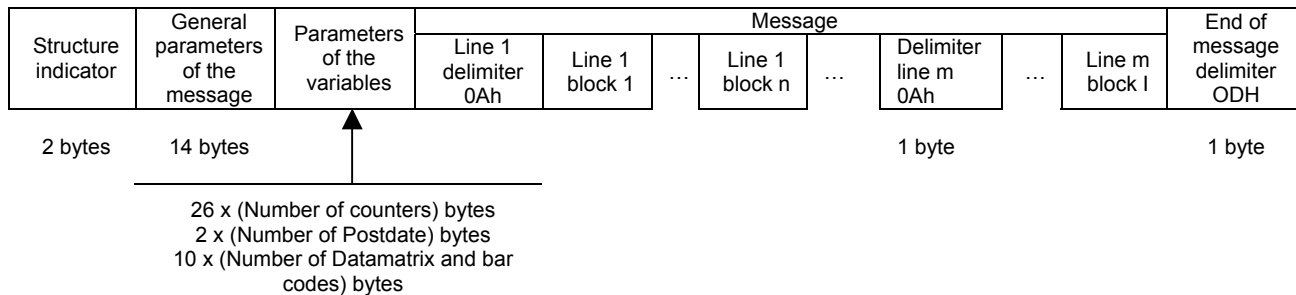
C5h	Identification (1 byte)
00h, 01h	Longueur (2 octets)
xxh	Data (1 byte): - report of memory recording of the message
xxh	Checksum (1 byte)



■ Data detail

Sending:

- Message number in library.
- Message heading: the title chosen for the message, 8 numeric.
- Message parameters and text.



All the details are to be found in the section "Protocols - Precisions/Examples".

- Type of write:
 - 00 creation of a message
 - 01 replacement of an existing message

NOTE

If the Type of Write byte is missing in the request, the printer creates or replaces the message depending on the number received.

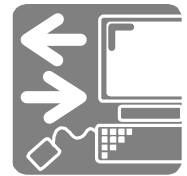
Reply: Report of memory recording of the message.

- 1 byte:
 - 00 write with message replacement
 - 01 write with message creation
 - 02 message not recorded (library full)
 - 03 message not recorded (no text in message received)
 - 04 message not recorded (text contains elements of an invalid function in the printer)
 - 05 message not recorded (library invalid)
 - 06 reserved for cancellation of message
 - 07 reserved for cancellation of message
 - 08 message not written in library (heading of message not valid)

NOTE

Before storing a message in memory it is necessary to check the free message numbers and the validity of the functions used.

Protocols - Sending



Sending cancellation of a message

CALCULATOR

PRINTER



Identification (1 byte)	C7h
Length (2 bytes)	00h, 02h
Number of the message to be cancelled (2 bytes)	xxh, xxh
Checksum (1 byte)	xxh



C5h	Identification (1 byte)
00h, 01h	Length (2 bytes)
xxh	Data (1 byte). - report of the cancellation
xxh	Checksum (1 byte)

■ Details of the reply data

- 00 à 05 Reserved for message writing
- 06 Message cancelled
- 07 Message not cancelled (library invalid or message active)

IMPORTANT *An active message cannot be cancelled when printing.*



Protocols - Sending

Print sending

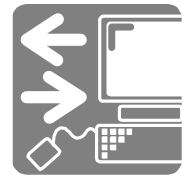
This command starts up the active message printing.

CALCULATOR

PRINTER



Identification (1 byte)	94h
Length (2 bytes)	00h, 00h
Checksum (1 byte)	94h



■ Sending of the variables

Initialization of the autodating

CALCULATOR

PRINTER



Identification (1 byte)	C8h
Length (2 bytes)	00h, 07h
Data: (7 bytes) - seconds in "bcd": 00 to 59 - minutes in "bcd": 00 to 59 - hours in "bcd": 00 to 23 - day in "bcd": 01 to 31 - month in "bcd": 01 to 12 - year in "bcd": 00 to 99 - 12/24 H mode in ASCII (A/P/SPACE)	xxd xxd xxd xxd xxd xxd xxh
Checksum (1 byte)	xxh

CAUTION

The data, seconds, minutes, hours, day, month, year are to be coded in BCD and the mode in ASCII characters.

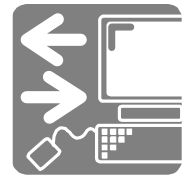
Sending the autodating parameters

CALCULATOR

PRINTER



Identification (1 byte)	ECh
Length (2 bytes)	xxh, xxh
Data: - type of parameter to be modified (2 bytes) - parameters to be modified according to the type previously chosen (from 1 to 112 bytes)	xxh, xxh xxh, ..., ..., xxh
Checksum (1 byte)	xxh



■ Data details

The first two bytes allow choice of amendment parameters and the following bytes are the amended parameters.

- TYPE 0000: postdate 1 modulo (2 bytes) 0000 to 9999

For details of these 2 bytes see § "Request for autodating parameters"

- TYPE 0001: shift code 1 parameters (5 bytes)

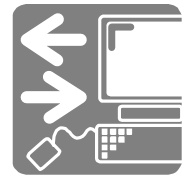
Starting time	00h to 17h	(1 byte)
Starting minute	00h to 3Bh	(1 byte)
Mode 12/24 H in ASCII	(A/P/SPACE)	(1 byte)
Number of hours of the interval	00h to 17h	(1 byte)
Number of minutes of the interval	00h to 3Bh	(1 byte)

- TYPE 0002: shift code 2 parameters (8 bytes)

Starting day	00h to 1Fh	(1 byte)
Starting month	00h to 0Ch	(1 byte)
Starting hour	00h to 17h	(1 byte)
Starting minute	00h to 3Bh	(1 byte)
Mode 12/24 H in ASCII	(A/P/SPACE)	(1 byte)
Number of days of the interval	00h to 1Fh	(1 byte)
Number of hours of the interval	00h to 17h	(1 byte)
Number of minutes of the interval	00h to 3Bh	(1 byte)

- TYPE 0003: first weekday for coding the day of the week (1 byte)

00	Monday
01	Tuesday
02	Wednesday
03	Thursday
04	Friday
05	Saturday
06	Sunday



□ TYPE 0004: shift code 3 parameters (112 bytes)

Table number for Monday	(01 to 07)	(1 byte)
Table number for Tuesday	(01 to 07)	(1 byte)
Table number for Wednesday	(01 to 07)	(1 byte)
Table number for Thursday	(01 to 07)	(1 byte)
Table number for Friday	(01 to 07)	(1 byte)
Table number for Saturday	(01 to 07)	(1 byte)
Table number for Sunday	(01 to 07)	(1 byte)

□ Tables 1 to 7 (15 bytes x 7)

Starting hour	(00h to 17h)
Starting minute	(00h to 3Bh)
Mode 12/24H in ASCII	(A/P/Space)
Number of hours of the interval	(00h to 17h)
Number of minutes of the interval	(00h to 3Bh)
5 bytes reserved	00h, 00h, 20h, 00h, 00h
5 bytes reserved	00h, 00h, 20h, 00h, 00h

□ TYPE 0005 (10 bytes)

postdate modulo 2	0000 to 9999	(2 bytes)
postdate modulo 3	0000 to 9999	(2 bytes)
postdate modulo 4	0000 to 9999	(2 bytes)
postdate modulo 5	0000 to 9999	(2 bytes)
postdate modulo 6	0000 to 9999	(2 bytes)

For details of these 2 bytes see § "Request for autodatating parameters"

NOTE	<i>Only one type of parameter is to be amended per sending.</i>
-------------	---



Sending of external variables

CALCULATOR

PRINTER



Identification (1 byte)	99h
Length (2 bytes)	xxh, xxh
Data:	
- start delimiter of external variable 1	12h
- external variable text 1 (n bytes)	xxh
- end delimiter of external variable 1	12h
...	...
- start delimiter of external variable 10	12h
- external variable text 10 (n bytes)	xxh
- end delimiter of external variable 10	12h
Checksum (1 byte)	xxh

■ Data details

This command cannot be used unless the zones to be modified, separated by the 12h delimiter, are already in the active message.

If a zone should not be modified, it should be sent with text containing no characters.

10 variables maximum per message.



Sending an autodating table

CALCULATOR

PRINTER



Identification (1 byte)	DFh
Length (2 bytes)	xxh, xxh
Data: - type of table (1 byte) - table (n bytes)	xxh xxh, ...
Checksum (1 byte)	xxh

■ Data details

□ Type of table:

00h	hours table
01h	minutes table
02h	day of the week table
03h	day of the year table
04h	day of the month table
05h	weeks table
06h	month of the year table
07h	year table
08h	shift code 2 table
0Ah	shift code 3 table
0Ch	Hégirien Calendar

□ Table:

hours table	24 x 3 characters ASCII
minutes table	60 x 3 characters ASCII
day of the week table	7 x 3 characters ASCII
day of the year table	366 x 3 characters ASCII
day of the month table	31 x 3 characters ASCII
week table	53 x 3 characters ASCII
month of the year table	12 x 3 characters ASCII
year table	10 x 3 characters ASCII
shift code 2 table	366 x 3 characters ASCII
shift code 3 table	(24 x 3 x 7) x 3 characters ASCII
Hegirien Calendar	2 + (24 x 4) bytes



Counters initialization

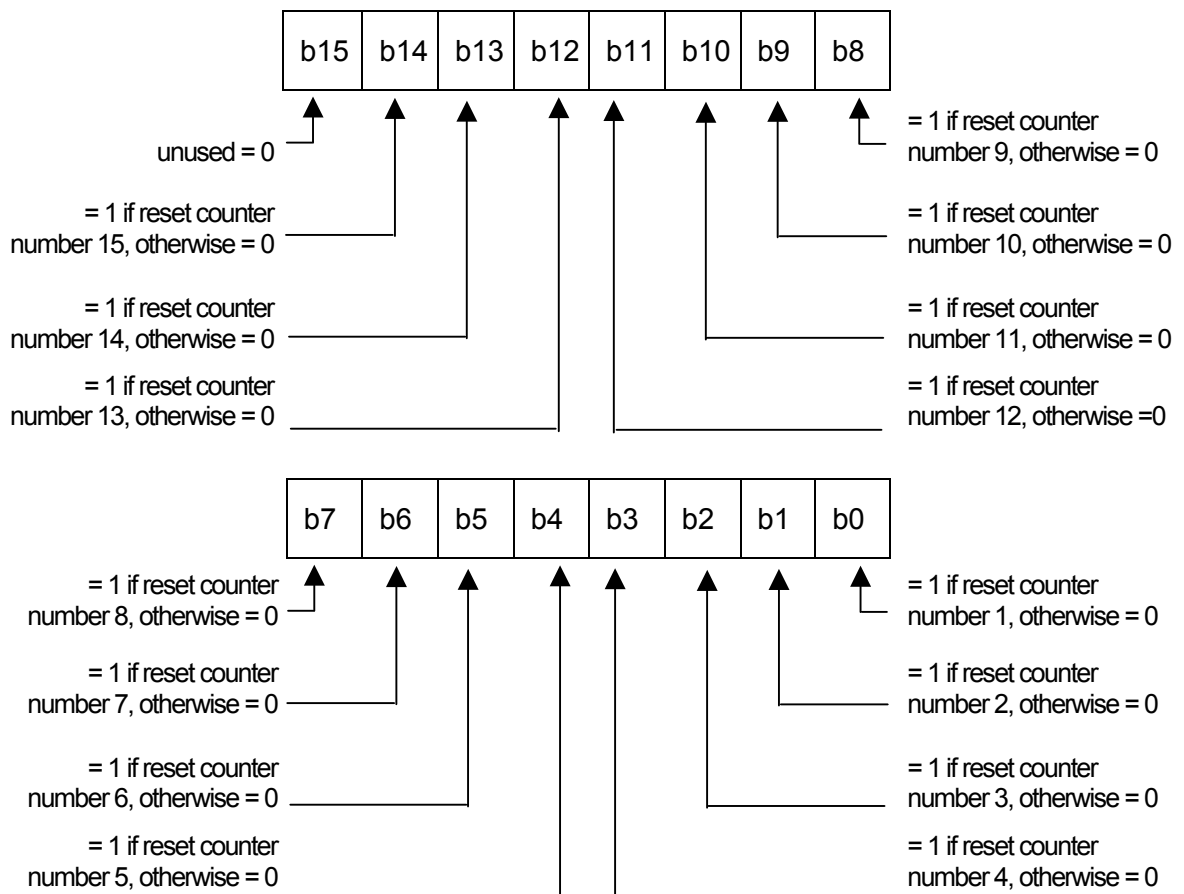
CALCULATOR

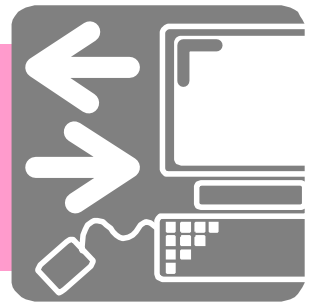
PRINTER



Identification (1 byte)	97h
Length (2 bytes)	00h, 02h
Data (2 bytes)	xxh, xxh
Checksum (1 byte)	xxh

Data details

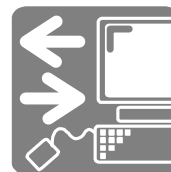




Protocols - Request



Protocols - Request



Protocols - Request

■ Request concerning the printer

Request printer status

CALCULATOR

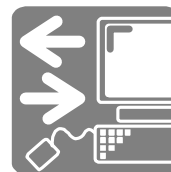
PRINTER



Identification (1 byte)	A6h
Length (2 bytes)	00h, 00h
Checksum (1 byte)	A6h



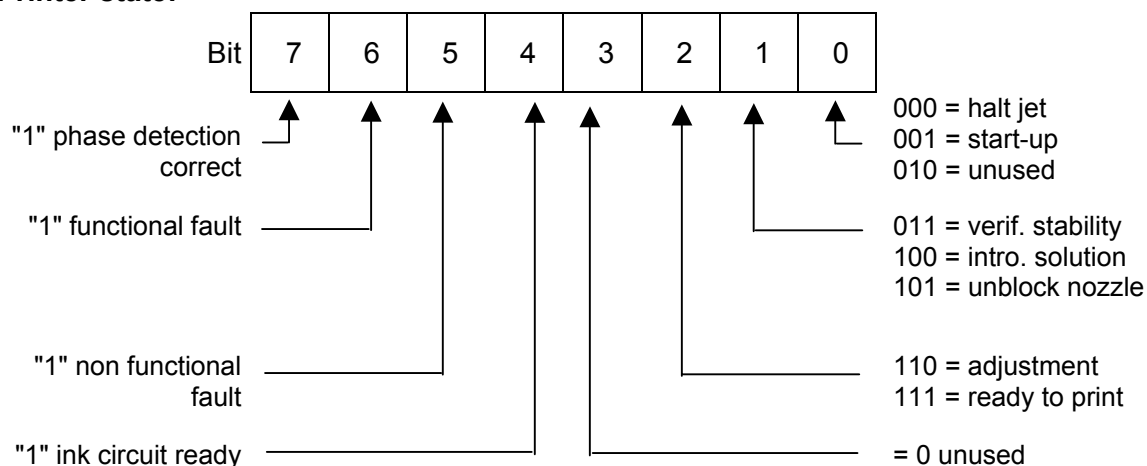
E4h	Identification (1 byte)
00h, 0Fh	Length (2 bytes)
xxh 2Bh xxh, xxh 2Bh ou 2dh xxh, xxh 2Bd ou 2Dh xxh xxh xxh xxh, xxh xxh,xxh	Data bytes (15 bytes): - printer state (1 byte) - reference pressure sign (1 byte) - reference pressure (2 bytes) - accumulator pressure sign (1 byte) + (2Bh) or - (2Dh) - accumulator pressure (2 bytes) - buffer pressure sign (1 byte) + (2Bh) or - (2Dh) - buffer pressure (2 bytes) - temperature (1 byte) - jet 1 speed (2 bytes) - jet 2 speed (2 bytes)
xxh	Checksum (1 byte)



Protocols - Request

■ Data details

Printer state:



Pressure reference sign (1 byte)

The pressure reference is always positive. The value is 2Bh.

Reference pressure (2 bytes)

Value expressed in centibars.

Accumulator pressure sign (1 byte)

For a positive pressure, the value is 2Bh.
For a negative pressure, the value is 2Dh.

Accumulator pressure (2 bytes)

Value expressed in centibars.

Buffer pressure sign (1 byte)

For a positive pressure, the value is 2Bh.
For a negative pressure, the value is 2Dh.

Buffer pressure (2 bytes)

Value expressed in centibars.

Temperature (1 byte)

Value expressed in Celsius degrees.

Jet 1 speed (2 bytes)

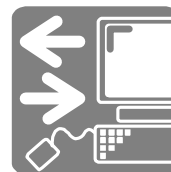
Value expressed in centimeters per second.

Jet 2 speed (2 bytes)

Value expressed in centimeters per second.

NOTE

If the printer has a monojet print head, the speed of Jet 2 equals 0 (1st and 2nd bytes at 00h).



Protocols - Request

Request of faults

CALCULATOR

PRINTER



Identification (1 byte)	DAh
Length (2 bytes)	00h, 00h
Checksum (1 byte)	DAh



D2h	Identification (1 byte)
xxh, xxh	Length (2 bytes)
xxh xxh xxh	Data (n bytes). - number of faults (1 byte) - fault code 1 (1 byte) - fault code "X" (1 byte)
xxh	Checksum (1 byte)

■ Data details

Number of faults

This value represents the number of faults appearing on the printer.



Protocols - Request

Faults code (from 1 to X)

A hexadecimal code corresponds to each fault.

N°	A/F	Message displayed
02	F	CRC EEPROM ink circuit incorrect
03	F	CRC EEPROM CPU incorrect
04	F	CRC Flash memory error (print)
05	F	EEPROM ink memory error
08	D	Algo flash memory CRC error
10	A	External communication timeout
11	F	Internal communication timeout
14	F	Incorrect external data received
16	F	Totaliser end value reached
17	F	CRC (RAM) incorrect for algorithms
18	F	Font used in message not available
19	F	CRC (RAM) incorrect for fonts
21	A	Duplicate print non authorized
22	F	CRC (RAM) message memory lost
25	F	FPGA impossible to program
26	F	CRC message library incorrect
27	A	Timeout speed measurement (20s)
28	F	No signal for speed measurement
30	F	CRC (RAM) Ink Incorrect
31	F	CRC (RAM) Print Incorrect
32	F	Incorrect date / time composition
33	A	Incompatible software versions
34	F	Incorrect or no compressed air
35	F	Cooling fan failure
38	A	Read/write error on Ink Mod EEPROM
39	A	Read/write error on CPU EEPROM
40	A	Print speed above specifications
45	A	Message contains non valid option
46	F	CRC Flash memory incorrect (Font)
48	F	Algorithm not available
50	A	LOW INK level. Prepare new cartridge
51	A	LOW MAKE-UP. Prepare new cartridge
52	F	EMPTY INK cartridge
53	F	EMPTY MAKE-UP cartridge
54	F	Full waste cartridge
55	A	Prevent: Change air filters
56	A	Prevent: Change ink circuit
57	A	Prevent: Change pressu filter
60	A	Ink valve malfunction
61	A	Make-up valve malfunction
62	A	Buffer valve malfunction



Protocols - Request

63	A	Accumulator valve malfunction
64	A	Condenser valve malfunction
70	A	Ink concentration out of specifications
71	F	Ambient pressure measurement failed
72	F	Motor synchronization incorrect
73	F	Ink transfer timeout
74	F	Air inflation timeout
75	A	Air pocket management incorrect
76	A	Jet refreshment timeout
77	A	Pump performance incorrect
78	F	Printer empty. No more ink available
79	F	Condenser overflow
81	F	Drop detection jet 1 (left)
82	F	Drop detection jet 2 (right)
83	F	Ink concentration out of range
84	F	Head cover absent
85	F	Recovery not detected. Jet out of gutter
86	F	High voltage supply failure
87	F	High voltage insulation failure
88	F	Jetspeed measurement timeout
90	A	Temperature out of range
91	F	Temperature inside console > 60°C – 140°F
92	F	Pressure servo control

NOTE *An “A” (Alarm) signal indicates a fault that does not inhibit printing.*

An “F” (Fault) signal indicates a fault that inhibits printing.



Protocols - Request

Request for the history of the last thirty faults

CALCULATOR

PRINTER



Identification (1 byte)	CFh
Length (2 bytes)	00h, 01h
Data (1 byte): - Fault position	xxh
Checksum (1 byte)	xxh



D0h	Identification (1 byte)
00h, 74h	Length (2 bytes)
xxh xxh, ... xxh, ... xxh, ...	Data: - fault number (see § "Request for machine faults") (1 byte) - time and date of notification (days, month, year, hours, minutes) 5 bytes - "Printer status" on notification of the fault (11 bytes) - 9 last recordings of "Printer status" prior to fault notification (11 bytes)
xxh	Checksum (1 byte)



Protocols - Request

■ Data details

"Printer status":

- ☐ Reference pressure
- ☐ Accumulator pressure
- ☐ Buffer pressure
- ☐ Jet 1 speed
- ☐ Jet 2 speed
- ☐ Temperature

Pressures are expressed in two bytes. Bit 15 is the sign bit (b15 to 1: -, b15 to 0: +).

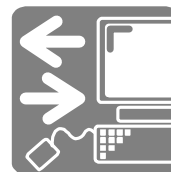
■ Detail of the request data

The faults are recorded in an L.I.F.O. file array. The request data are the fault position in this array:

- ☐ first in the table : 00
- ☐ following : 01
- ☐ preceding : 02

■ Detail of the reply data

The printer gives information on the last six recording of the machine status (recording every minute), with the number, time and date of the fault.



Protocols - Request

Request for status of the tachometer pulse number/fly time offset output

CALCULATOR

PRINTER



Identification (1 byte)	B7h
Length (2 bytes)	00h, 03h
Data (3 bytes): - current version A12 (3 ASCII characters)	41h,31h,32h
Checksum (1 byte)	B7h



B8h	Identification (1 byte)
00h, 06h	Length (2 bytes)
	Data: - Measurement type (2 bytes) - Count value (4 bytes)
xxh	Checksum (1 byte)

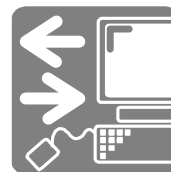
■ Data details

Measurement type:

- 00 tachometer pulse number output
- 01 inactive output
- 02 fly time offset output

Count value:

- If measurement type = 00 tachometer pulse number (300 to 999999999)
- If measurement type = 02 compensation time (0 to 2000 µs)



Protocols - Request

Request for software version

CALCULATOR

PRINTER



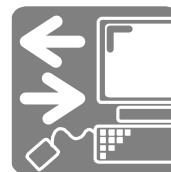
Identification (1 byte)	B1h
Length (2 bytes)	00h, 01h
Checksum (1 byte)	B0h



B2h	Identification (1 byte)
00h, 29h	Length (2 bytes)
xxh xxh xxh xxh xxh xxh xxh xxh, ... xxh, ... xxh, ... xxh, ... xxh, ... xxh, ...	Data: - ink processor version: 1 ASCII character, A to Z index: 2 ASCII characters (00 to 99) type of application: 1 byte (00 = standard) - print processor version: 1 ASCII character, A to Z index: 2 ASCII characters, 00 to 99 type of application: 1 byte (00 = standard) - Reserved, always = 1 - CPU revision (6 bytes) - IMP revision (6 bytes) - Boot CPU version (4 bytes) - Boot IMP version (4 bytes) - Boot CPU revision (6 bytes) - Boot IMP revision (6 bytes)
xxh	Checksum (1 byte)

NOTE

If the length is "0", the response will be S7 type. See corresponding manual.



Protocols - Request

Request for functions

CALCULATOR

PRINTER



Identification (1 byte)	ADh
Length (2 bytes)	00h, 00h
Checksum (1 byte)	ADh



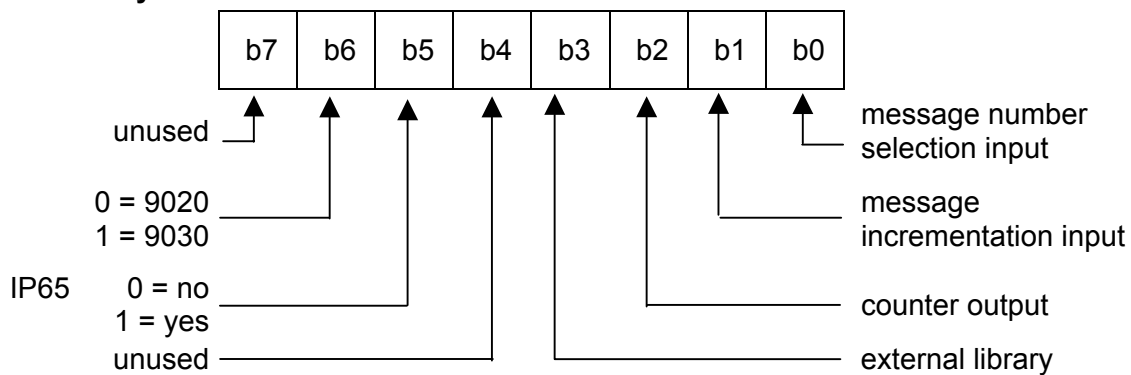
B0h	Identification (1 byte)
00h, 08h	Length (2 bytes)
xxh, ...	Data (8 bytes)
xxh	Checksum (1 byte)

■ Data detail:

Function present if the corresponding bit is at 1.

The first five bytes are not used.

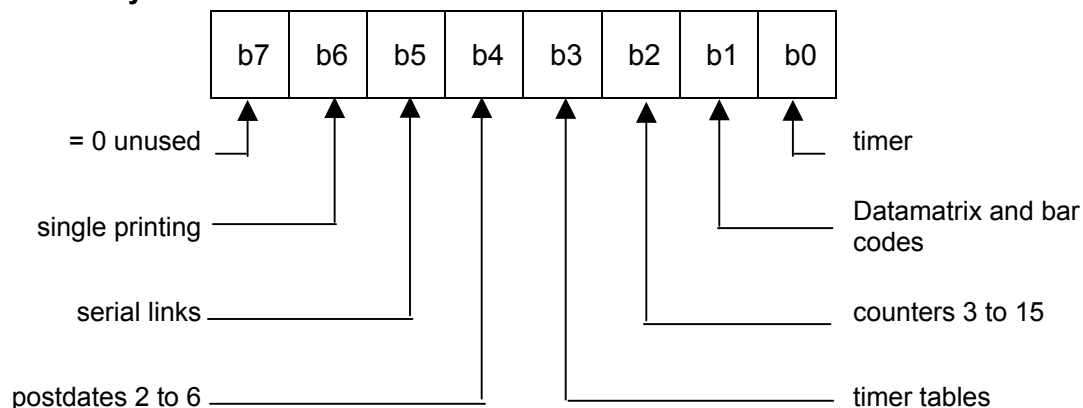
□ 6th byte:





Protocols - Request

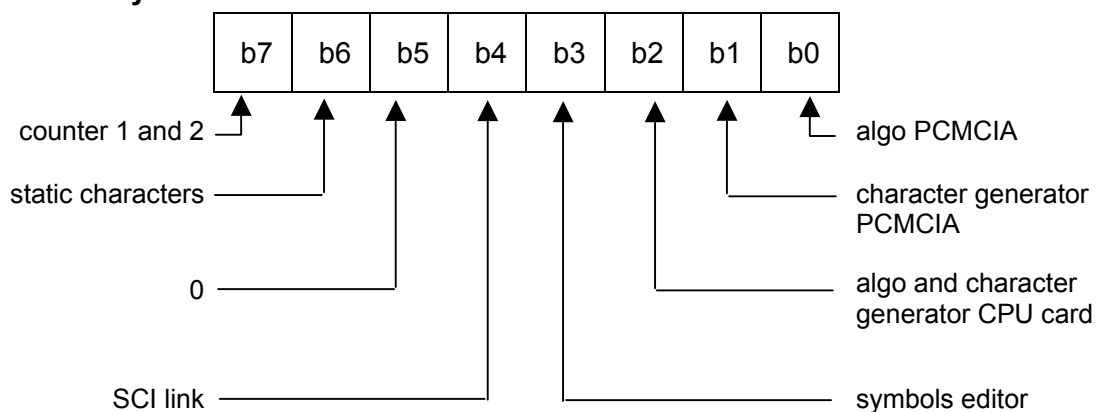
□ 7th byte:

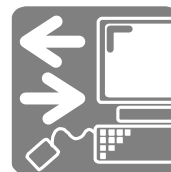


Counters 3 to 15: this function allows utilization of 15 counters, inputs of increment counters 1 and 2, zeroing of counters 1 and 2 (only for 9030).

Postdates 2 to 6: This function allows utilization of 6 postdates.

□ 8th byte:





Protocols - Request

Request for types of character generators available in the printer

CALCULATOR

PRINTER



Identification (1 byte)	A4h
Length (2 bytes)	00h, 03h
Data (3 bytes): - type (3 bytes)	00h, 00h, 00h
Checksum (1 byte)	A7h



A5h	Identification (1 byte)
00h, xxh	Length (2 bytes)
xxh, xxh xxh, ... xxh, ... xxh, ...	Data: - number of generators (2 bytes) - description of the first generator (22 bytes) ... - description of the end generator (22 bytes)
xxh	Checksum (1 byte)

■ Data detail

Description of a generator:

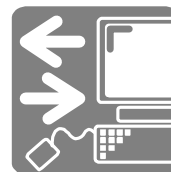
- ☐ title of the generator (16 ASCII characters)
- ☐ reference number of the generator (1 byte)
- ☐ number of drops per frame (1 byte)
- ☐ number of frames per character (2 bytes)
- ☐ number of characters (1 byte)
- ☐ number of byte of the full frame (1 byte)

See APPENDIX for the list of character generators.

NOTE

If there is no character generator stored in the printer the length of the reply is 00h, 00h.





Protocols - Request

■ Requests concerning messages

Request for general parameters by default

CALCULATOR

PRINTER

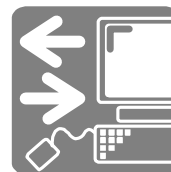


Identification (1 byte)	CBh
Length (2 bytes)	00h, 03h
Data (3 bytes): - current version A12 (3 ASCII characters)	41h, 31h, 32h
Checksum (1 byte)	CBh



CDh	Identification (1 byte)
xxh, xxh	Length (2 bytes)
ODh	Data: - structure indicator (2 bytes) - general parameters (14 bytes) - parameter counters 1 and 2 (52 bytes) - postdate 1 (2 bytes) - parameter counters 3 to 15 (338 bytes) - postdate 2 to 6 (10 bytes) - Datamatrix and bar codes 1 to 3 (30 bytes) - end of message indicator (1 byte)
xxh	Checksum (1 byte)

See § "Protocols – Precisions/Examples" for details.



Protocols - Request

Request for a message by number

CALCULATOR

PRINTER



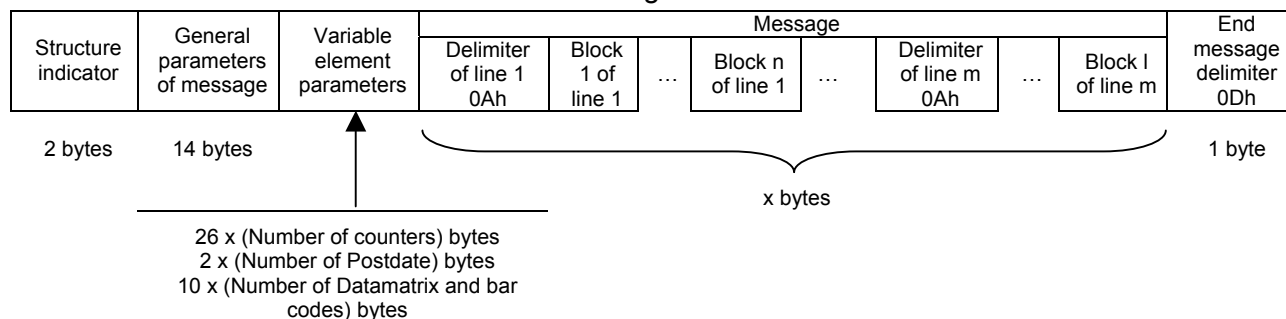
Identification (1 byte)	92h
Length (2 bytes)	00h, 02h
Data (2 bytes): - message number 001 to 999	xxh, xxh
Checksum (1 byte)	xxh



9Fh	Identification (1 byte)
xxh, xxh	Length (2 bytes)
xxh,xxh xxh,... xxh,...	Data: - message number (2 bytes) 001 to 999 - message heading (8 bytes) 8 ASCII characters - message parameters and text (n bytes)
xxh	Checksum (1 byte)

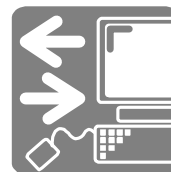
■ Data details

. Parameters and text of the message



See "Protocols – Precisions/Examples" section for details.





Protocols - Request

Request for a free message number

CALCULATOR

PRINTER



Identification (1 byte)	A2h
Length (2 bytes)	00h, 00h
Checksum (1 byte)	A2h



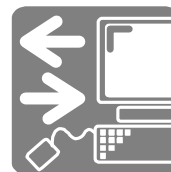
A3h	Identification (1 byte)
00h, 02h	Length (2 bytes)
xxh, xxh	Data (2 bytes): - message number available (2 bytes) 001 to 999
xxh	Checksum (1 byte)

■ Data details

Message number available: the printer always gives the lowest number available.

NOTE

*This request must only be used when storing a message in the library in order not to overwrite the existing messages.
If the reply is 000 there are no more numbers available or the library is invalid*



Protocols - Request

Request for the number of the active message

CALCULATOR

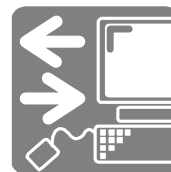
PRINTER



Identification (1 byte)	DBh
Length (2 bytes)	00h, 00h
Checksum (1 byte)	DBh



91h	Identification (1 byte)
00h, 0Ah	Length (2 bytes)
xxh, xxh	Data: - message number (2 bytes) 001 to 000 - message heading 8 bytes ASCII
xxh	Checksum (1 byte)



Protocols - Request

■ Request concerning variables

Request for autdating parameters

CALCULATOR

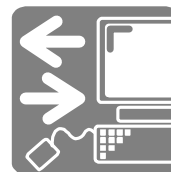
PRINTER



Identification (1 byte)	BBh
Length (2 bytes)	00h, 03h
Data (3 bytes): - current version A12 (3 ASCII characters)	41h,31h,32h
Checksum (1 byte)	BBh



BCh	Identification (1 byte)
00h, 8Fh	Length (2 bytes)
xxh - - - xxh	Data (143 bytes): - postdate 1 modulo - first day of the week for coding the week-day - shift code 1 parameters - shift code 2 parameters - language 1 for months of the year - shift code 3: - shift code 3 parameters - table 1 to 7 - postdate 2 modulo - postdate 3 modulo - postdate 4 modulo - postdate 5 modulo - postdate 6 modulo - language 2 for months of the year - date change time shift
xxh	Checksum (1 byte)



Protocols - Request

■ Data details

Postdate 1 modulo:

From 0000 to 9999 (2 bytes)

First day of the week for coding the week-day (1 byte):

00	Monday
01	Tuesday
02	Wednesday
03	Thursday
04	Friday
05	Saturday
06	Sunday

Shift code 1 parameters (5 bytes):

Starting time	00h to 17h	(1 byte)
Starting minute	00h to 3Bh	(1 byte)
Mode 12/24 H in ASCII	(A/P/SPACE)	(1 byte in ASCII)
Number of hours of the interval	00h to 17h	(1 byte)
Number of minutes of the interval	00h to 3Bh	(1 byte)

Shift code 2 parameters (8 bytes):

Starting day	00h to 1Fh	(1 byte)
Starting month	00h to 0Ch	(1 byte)
Starting hour	00h to 17h	(1 byte)
Starting minute	00h to 3Bh	(1 byte)
Mode 12/24 H in ASCII	(A/P/SPACE)	(1 byte in ASCII)
Number of days of the interval	00h to 1Fh	(1 byte)
Number of hours of the interval	00h to 17h	(1 byte)
Number of minutes of the interval	00h to 3Bh	(1 byte)



Protocols - Request

Language 1 for months of the year (1 byte):

Number	Language
00h	French
01h	English
02h	German
03h	Italian
04h	Spanish
05h	Norwegian
06h	Swedish
07h	Dutch
08h	Danish
09h	Japanese (Kanji)
0Ah	Portuguese
0Bh	Brasilian
0Ch	Hungarian
0Dh	Polish
0Eh	Turkish
0Fh	Czech
10h	Arab
11h	Chinese
12h	Korean
13h	Hebrew
14h	Russian
15h	Thai
16h	Greek
17h	Finnish
18h	Croatian
19h	Slovenian
1Ah	Vietnamese
1Bh	Iranian
1Ch	Bulgarian
1Dh	Indonesian
1Eh	Taiwanese
1Fh	Hegirien



Protocols - Request

Shift code 3 (112 bytes):

□ Shift code 3 parameters (7 bytes)

Table number for Monday	(01 to 07)	(1 byte)
Table number for Tuesday	(01 to 07)	(1 byte)
Table number for Wednesday	(01 to 07)	(1 byte)
Table number for Thursday	(01 to 07)	(1 byte)
Table number for Friday	(01 to 07)	(1 byte)
Table number for Saturday	(01 to 07)	(1 byte)
Table number for Sunday	(01 to 07)	(1 byte)

□ Tables 1 to 7 (15 bytes * 7)

Starting hour	(00h to 17h)
Starting minute	(00h to 3Bh)
Mode 12/24H in ASCII	(A/P/Space)
Number of hours of the interval	(00h to 17h)
Number of minutes of the interval	(00h to 3Bh)
5 bytes reserved	00h, 00h, 20h, 00h, 00h
5 bytes reserved	00h, 00h, 20h, 00h, 00h

Postdate 2 to 6 modulo (10 bytes)

postdate 2 modulo 0000 to 9999	(2 bytes)
postdate 3 modulo 0000 to 9999	(2 bytes)
postdate 4 modulo 0000 to 9999	(2 bytes)
postdate 5 modulo 0000 to 9999	(2 bytes)
postdate 6 modulo 0000 to 9999	(2 bytes)

Language 2 for months of the year (1 byte):

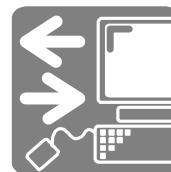
(the numbers are identical to month 1 of the year).

Date change time shift

This parameter also affects the "Day of the year" field

Shift direction (+ or -)	2bh or 20h
Hours	0 to 23 binary
Minutes	0 to 59 binary





Protocols - Request

Request for the autdating

CALCULATOR

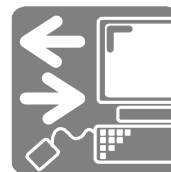
PRINTER



Identification (1 byte)	D6h
Length (2 bytes)	00h, 00h
Checksum (1 byte)	D6h



9Ch	Identification (1 byte)
00h, 16h	Length (2 bytes)
xxh, xxh	Data (25 bytes): - seconds in ASCII (2 bytes) 00 to 59 - minutes in ASCII (2 bytes) 00 to 59 - hours in ASCII (2 bytes) 00 to 23 - mode 12/24 in ASCII (2 bytes) AM/PM/SPACE - day of the month in ASCII (2 bytes) 01 to 31 - day of the year in ASCII (3 bytes) 001 to 366 - week of the year in ASCII (2 bytes) 01 to 53 - month of the year ASCII (2 bytes) 01 to12 - month of the year in ASCII (3 bytes) JAN to DEC - year in ASCII (2 bytes) 00 to 99
xxh, xxh	
xxh, xxh	
xxh, xxh	
xxh, xxh	
xxh, xxh	
xxh, xxh	
xxh, xxh	
xxh, xxh	
xxh, xxh	
xxh, xxh	
xxh, xxh	
xxh	Checksum (1 byte)



Protocols - Request

Request for an autodating table

CALCULATOR

PRINTER



Identification (1 byte)	DEh
Length (2 bytes)	00h, 01h
Data (1 byte): - type of table 00h to 0C	xxh
Checksum (1 byte)	xxh



DFh	Identification (1 byte)
xxh, xxh	Length (2 bytes)
xxh xxh,...	Data: - type of table (1 byte) - table (n bytes)
xxh	Checksum (1 byte)

■ Data details

Request: Type of table: (1 byte)

00h	Hours table
01h	Minutes table
02h	Day of the week table
03h	Day of the year table
04h	Day of the month table
05h	Weeks table
06h	Month of year table
07h	Years table
08h	Shift code 2 table
09h	Month of year table - language 1
0Ah	Shift code 3 table
0Bh	Month of year table - language 2
0Ch	Hegirien calendar



Protocols - Request

Answer: Autodating table (n bytes)

Hours table	24 x 3 ASCII characters
Minutes table	60 x 3 ASCII characters
Day of the week table	7 x 3 ASCII characters
Day of the year table	366 x 3 ASCII characters
Day of the month table	31 x 3 ASCII characters
Weeks table	53 x 3 ASCII characters
Month of year table	12 x 3 ASCII characters
Years table	10 x 3 ASCII characters
Shift code 2 table	366 x 3 ASCII characters
Month of year table - language 1	12 x 3 ASCII characters
Shift code 3 table	(24 x 3 x 7) x 3 ASCII characters
Month of year table - language 2	12 x 3 ASCII characters
Hegirien calendar	2 + (24 x 4) bytes

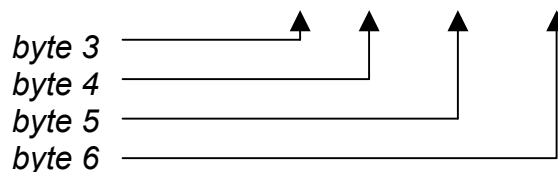
NOTE

Hegirien calendar

Bytes 1 and 2: year in Hijri calendar: century/year

Bytes 3 to 98: 24 month start dates

□ *start of month 1: Day/Month/Century/Year*

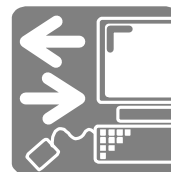


□ *start of month 2: bytes 7 to 10*

□ *start of month 3: bytes 11 to 14*

□ *etc.*

□ *start of month 24: bytes 95 to 98*



Protocols - Request

Request for the current value of the autodating elements

CALCULATOR

PRINTER



Identification (1 byte)	A9h
Length (2 bytes)	00h, 04h
Data (4 bytes): - timer type: 00 = autodating 01 = extended autodating - current version A12 (3 ASCII characters)	xxh 41h, 31 h, 32h
Checksum (1 byte)	BBh



ABh	Identification (1 byte)
00h, xxh	Length (2 bytes)
	Data: - according to the type required
xxh	Checksum (1 byte)



Protocols - Request

■ Data details

Autodating:

Seconds	from 00 to 59	(2 ASCII characters)
Minutes	from 00 to 59	(2 ASCII characters)
Hours	from 00 to 23	(2 ASCII characters)
Mode 12/24 H	AM/PM/SPACE	(2 ASCII characters)
Day of month	from 1 to 31	(2 ASCII characters)
Day of year	from 001 to 366	(2 ASCII characters)
Week of year	from 01 to 53	(2 ASCII characters)
Month of year	from 01 to 12	(2 ASCII characters)
Month of year in letters (language 1)	JAN to DEC	(2 ASCII characters)
Year on 2 digits	from 00 to 99	(2 ASCII characters)
Postday of month	from 01 to 31	(2 ASCII characters)
Postday of year	from 001 to 366	(2 ASCII characters)
Postweek of year	from 01 to 53	(2 ASCII characters)
Postmonth of year	from 01 to 12	(2 ASCII characters)
Postmonth of year in letters (language 1)	JAN to DEC	(2 ASCII characters)
Postyear	from 00 to 99	(2 ASCII characters)
Shift code 1 in letter	A to Z	(1 ASCII character)
Shift code 1 in number	from 01 to 99	(2 ASCII characters)
Shift code 1 in letter	A to Z -O -I	(1 ASCII character)
Day of week	from 1 to 7	(1 ASCII character)
Postday of year modulo programmable	from 1 to 999	(3 ASCII characters)
Delimiter ":"	3Ah	(1 ASCII character)
Delimiter "/"	2Fh	(1 ASCII character)
Delimiter "."	2Eh	(1 ASCII character)
Delimiter " "	20h	(1 ASCII character)
Month of year in letters (language 2)	JAN to DEC	(3 ASCII characters)
Postmonth of year in letters (language 2)	JAN to DEC	(3 ASCII characters)



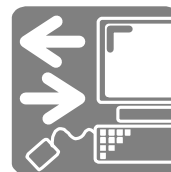
Protocols - Request

Extended autodatting:

□ For each postdate (2 to 6):

Postday of month	01 to 31	(2 ASCII characters)
Postday of year	001 to 366	(3 ASCII characters)
Postday of year modulo programmable		(3 ASCII characters)
Postweek of year	01 to 52	(2 ASCII characters)
Postmonth of year	01 to 12	(2 ASCII characters)
Postyear	00 to 99	(2 ASCII characters)
Postday of week in letters	MON to SUN	(3 ASCII characters)
Postmonth of year in letters	JAN to DEC	(3 ASCII characters)

Hours table		(3 ASCII characters)
Minutes table		(3 ASCII characters)
Month of year table		(3 ASCII characters)
Day of week table		(3 ASCII characters)
Day of month table		(3 ASCII characters)
Day of year table		(3 ASCII characters)
Week of year table		(3 ASCII characters)
Years table		(3 ASCII characters)
Shift code 2 table		(3 ASCII characters)
Shift code 3 table		(3 ASCII characters)
Delimiter ":"	3Ah	(1 ASCII character)
Delimiter "/"	2Fh	(1 ASCII character)
Delimiter "."	2Eh	(1 ASCII character)
Delimiter " "	20h	(1 ASCII character)



Protocols - Request

Request for the languages of the month to be printed

CALCULATOR

PRINTER



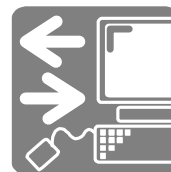
Identification (1 byte)	E0h
Length (2 bytes)	00h, 03h
Data (3 bytes): - current version A12 (3 ASCII characters)	41h, 31 h, 32h
Checksum (1 byte)	E0h



C4h	Identification (1 byte)
00h, 4Bh	Length (2 bytes)
xxh 01h xxh, ... xxh xxh, ...	Data: - number of the language 1 of the month (1 byte) - reserved and held at 01h - table of months 1 in ASCII (JAN to DEC) 36 bytes - number of the language 2 of the month (1 byte) - table of months 2 in ASCII (JAN to DEC) 36 bytes
xxh	Checksum (1 byte)

■ Data details

See command C4h "Languages of the month to be printed" of the precedent section



Protocols - Request

Request for the current value of the counters

CALCULATOR

PRINTER



Identification (1 byte)	95h
Length (2 bytes)	00h, 02h
Data (2 bytes)	xxh, xxh
Checksum (1 byte)	xxh



96h	Identification (1 byte)
xxh, xxh	Length (2 bytes)
xxh, ...	Data: - current value of the counters n x 9 ASCII characters (counter 1 → counter 15)
xxh	Check (1 octet)

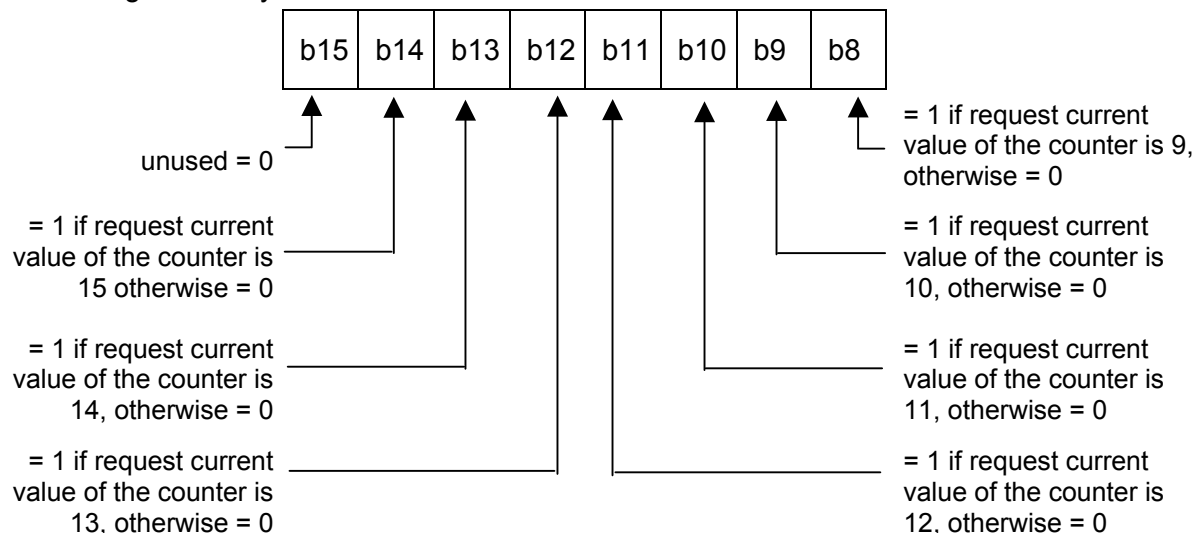


Protocols - Request

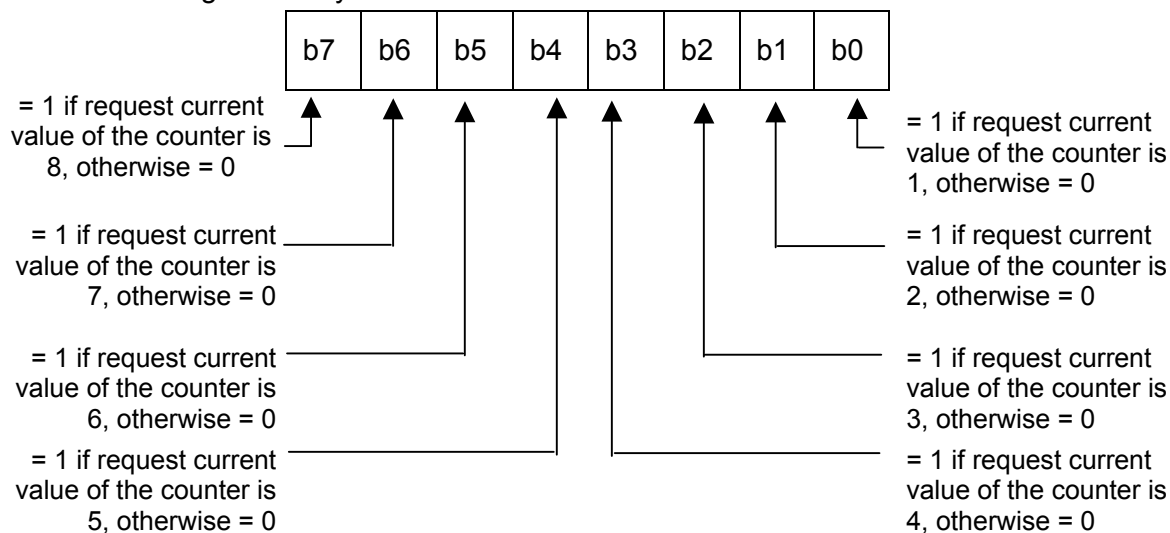
■ Data details

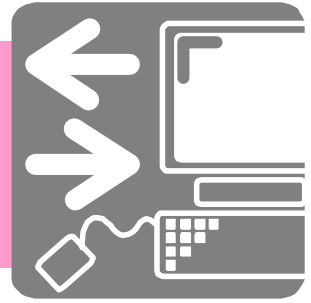
Request: counter list

□ High-order byte



□ Least significant byte





Protocols – Precisions / Examples



Protocols - Precisions/Examples

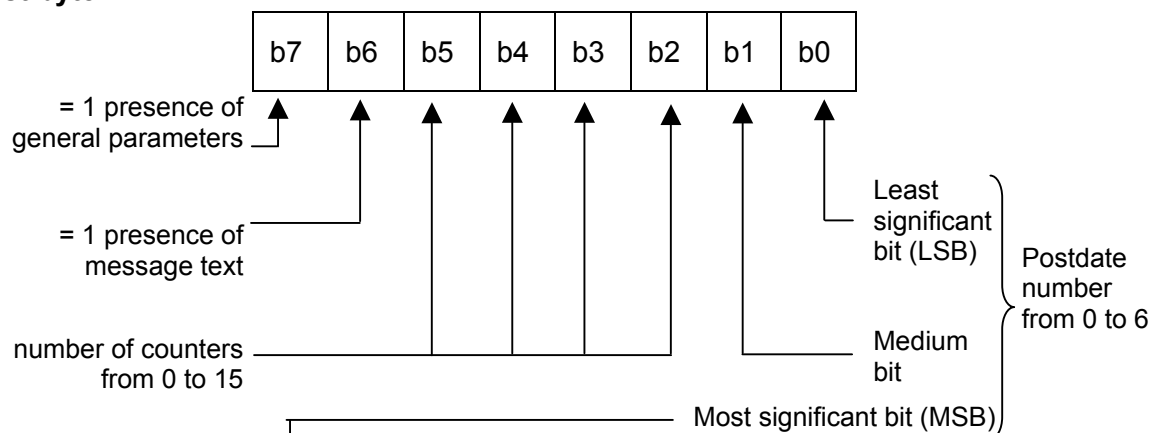


Protocols - Precisions/Examples

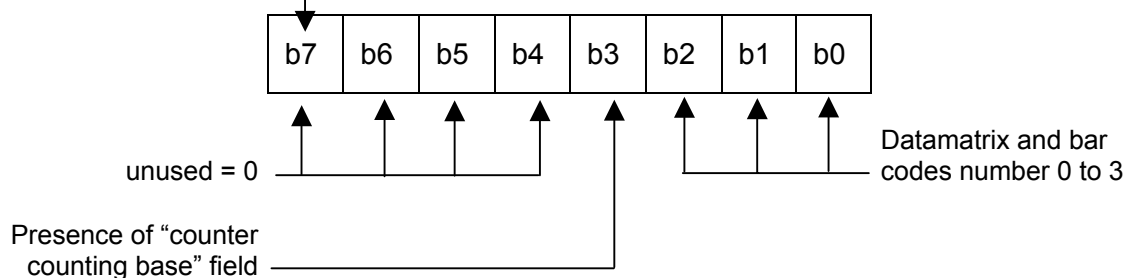
■ Message data details

Structure indicator

First byte:

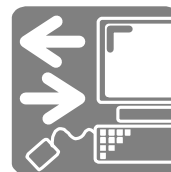


Second byte:



IMPORTANT

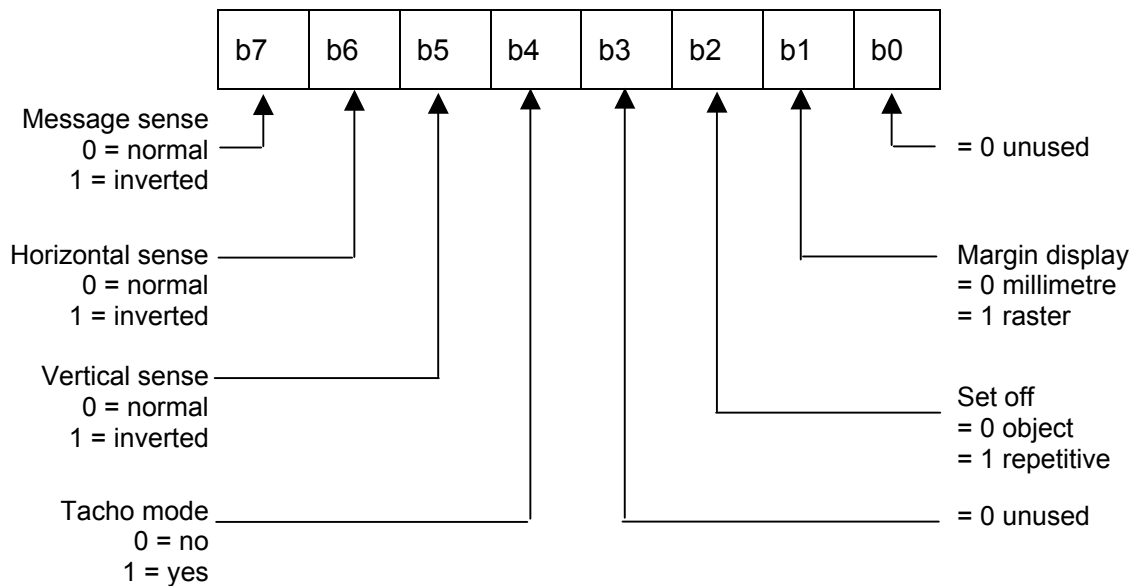
If the structure indicator is defined with variable elements (counter, Datamatrix and bar codes, postdate) the variable elements parameters must be inserted (refer to the "Variable elements parameters" section).



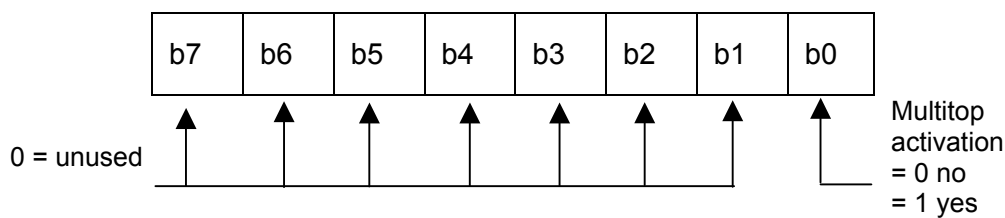
Protocols - Precisions/Examples

General parameters of the message

First byte:



Second byte:

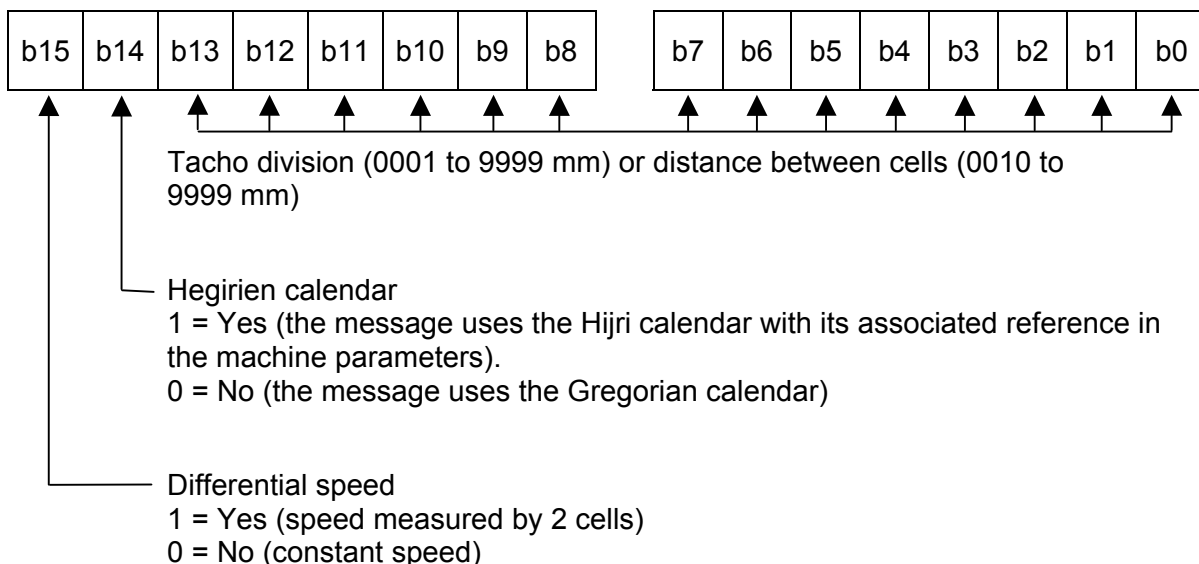


When multitop mode is enabled (= 1), bit b2 in the first byte is inhibited.



Protocols - Precisions/Examples

3rd and 4th bytes



5th and 6th bytes: forward margin (0003 to 9999 mm or 0009 to 9999 h frames)

7th and 8th bytes: return margin (0003 to 9999 mm or 0009 to 9999 h frames)

9th and 10th bytes: top object override (0002 to 9999 mm or 0009 to 9999 h frames)

11th and 12th bytes: printing speed in mm/s.

13th and 14th bytes: this parameter, coded in hexadecimal, is used to control parameters for a quick multi-line algorithm (see table in appendix) or to enable Curved mode (CurvePrint).
CurvePrint corresponds to algorithm 100.

NOTE

The “Tacho Mode” bit is “0” when the differential speed is measured.

The Tacho division is used for the “Differential speed mode” in this case.

b15 = 0 and Tacho Mode = 0: constant printing speed

b15 = 1 and Tacho Mode = 0: speed measured by 2 cells (differential speed mode)

b15 = 0 and Tacho Mode = 1: Tacho mode



Protocols - Precisions/Examples

Variable element parameters

Parameters of counter 1	26 bytes
Parameters of counter 2	26 bytes
Postdate 1	2 bytes
Parameters of counter 3	26 bytes
Parameters of counter 4	26 bytes
Parameters of counter 5	26 bytes
Parameters of counter 6	26 bytes
Parameters of counter 7	26 bytes
Parameters of counter 8	26 bytes
Parameters of counter 9	26 bytes
Parameters of counter 10	26 bytes
Parameters of counter 11	26 bytes
Parameters of counter 12	26 bytes
Parameters of counter 13	26 bytes
Parameters of counter 14	26 bytes
Parameters of counter 15	26 bytes
Postdate 2	2 bytes
Postdate 3	2 bytes
Postdate 4	2 bytes
Postdate 5	2 bytes
Postdate 6	2 bytes
Datamatrix or bar code 1	10 bytes
Datamatrix or bar code 2	10 bytes
Datamatrix or bar code 3	10 bytes
Counting base for the 15 counters: 15×2 ASCII characters	

IMPORTANT

The order of the different parameters is frozen, but their presence on the other hand is optional: it is defined by the structure indicator.

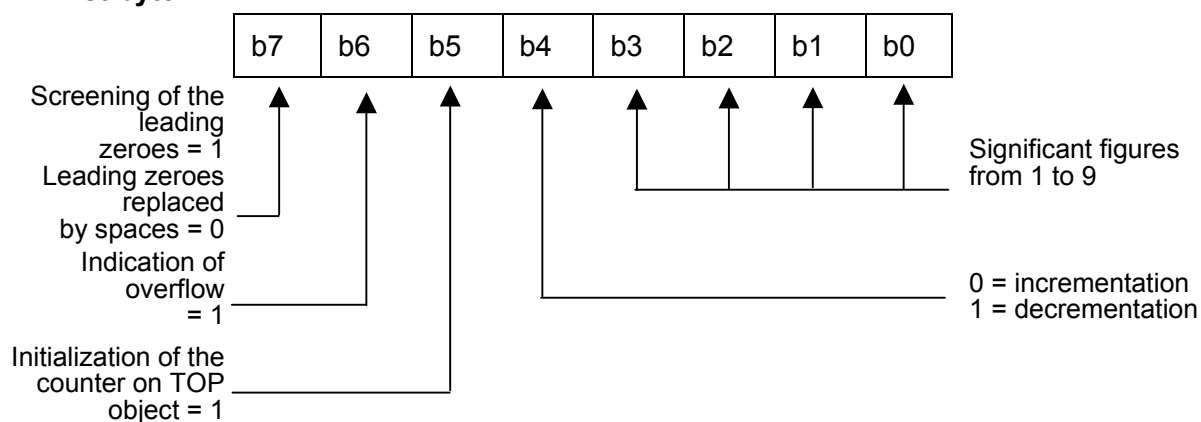


Protocols - Precisions/Examples

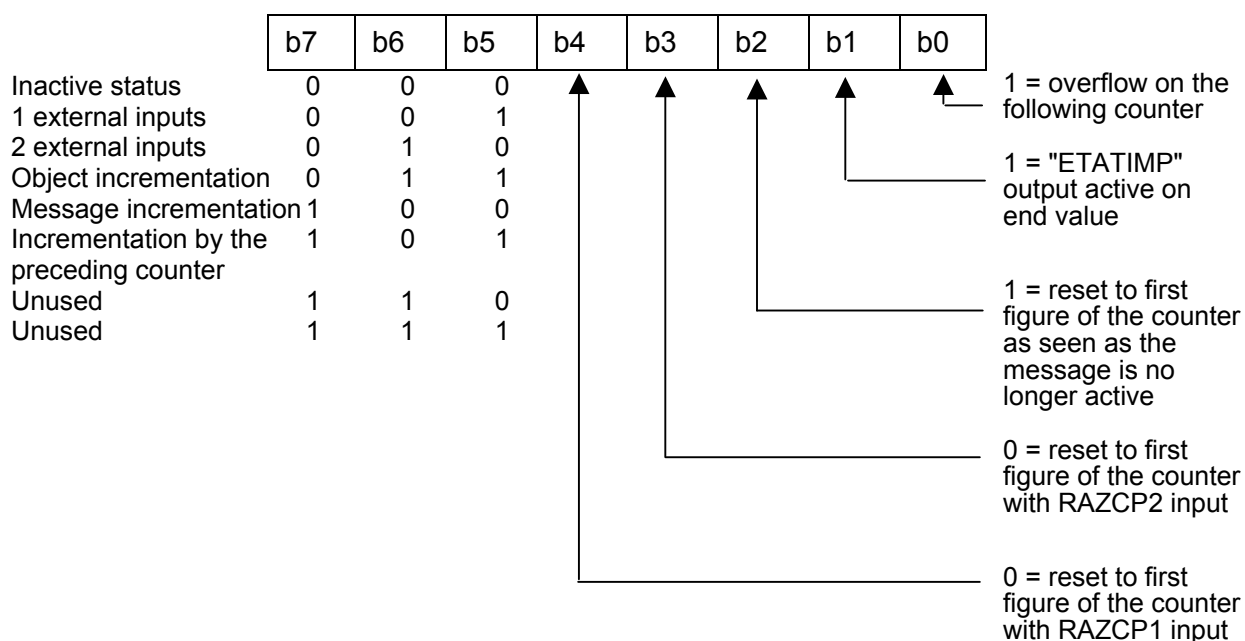
■ Parameters of a counter

IMPORTANT The following 26 bytes are configured in order to use a counter.
A configuration of 26 bytes corresponds to each counter.

First byte:

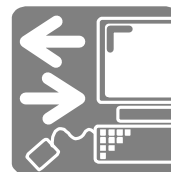


Second byte:



The following 24 bytes:

Start value in 9 ASCII characters
End value in 9 ASCII characters
Counter step in 2 ASCII characters
Incrementation divider (batch counter) in 4 bytes



Protocols - Precisions/Examples

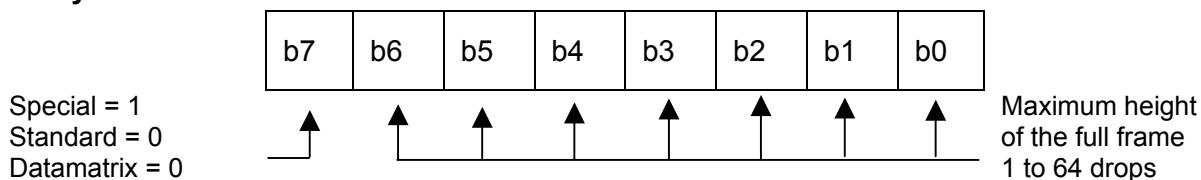
■ Postdates parameter (2 bytes)

Postdate interval coded on 2 bytes. Value from 0 to 9999 days.

Interval type:	b15	b14
Interval in number of days	0	0
Interval in number of weeks	0	1
Interval in number of months	1	0
Interval (value of 0 to 9999)	b13 to b0	

■ Datamatrix and bar codes parameters (10 bytes)

1er byte:



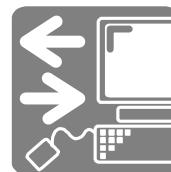
□ Standard: the bar code will be printed with the height defined by bits b0 to b6 (code example: 2/5, code 39).

□ Special: the bar code will be printed with a height defined of 24 drops or 18h.

□ Datamatrix: the height must be an even number of points, between 8 and 24. If an odd height is selected, the last drop selected will be empty.

NOTE

For standard and Datamatrix, the height must correspond to a font height present in the printer.



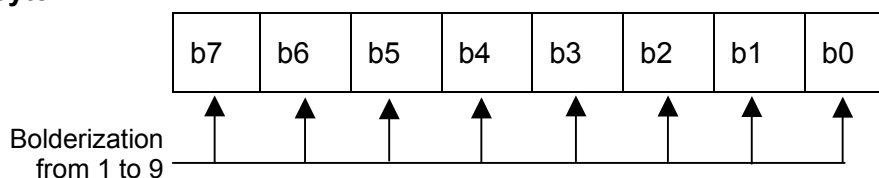
Protocols - Precisions/Examples

2nd byte: identification of the code

xx	Code
00h	2/5 interleaved
01h	Code 39
02h	EAN13
03h	EAN13 with plain text
04h	EAN8
05h	EAN8 with plain text
06h	UPCA
07h	UPCA with plain text
08h	UPCE
09h	UPCE with plain text
0Eh	EAN128
0Fh	EAN 128 with plain text
10h	Code 128
11h	Code 128 with plain text
14h	HIBC
17h	Datamatrix
18h	HIBC LIC CODE 128
19h	HIBC LIC CODE 128 with plain text
1Ah	HIBC LIC CODE 39
1Bh	HIBC LIC CODE 39 with plain text

Identification of the code allows the printer the choice of the law governing the code.
This rule also contains the raster structures used by the code.

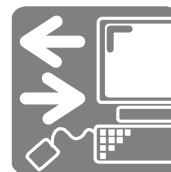
Third byte:



IMPORTANT

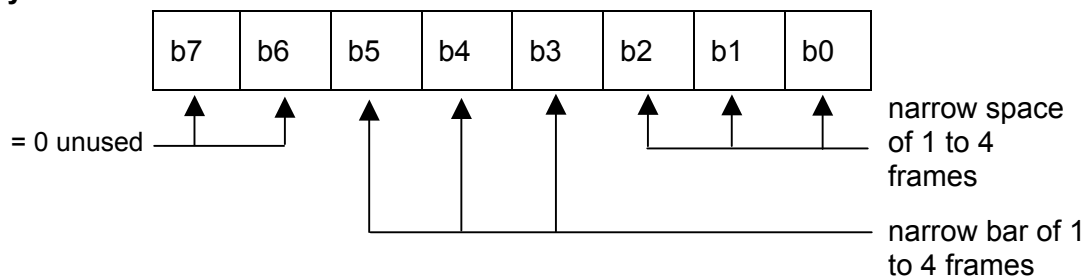
For industrial codes, this value is always equal to 1.

For Datamatrix codes, bolderization may only vary between 1 and 2. Bolderization 2 may only be used with a height of 16 drops or more.

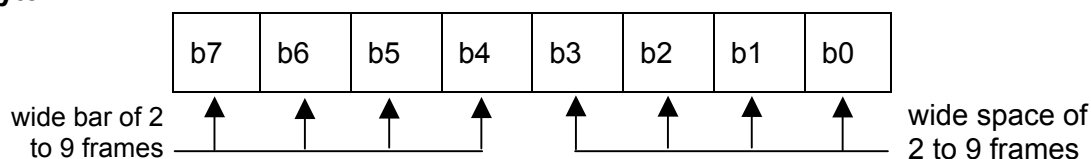


Protocols - Precisions/Examples

Fourth byte:



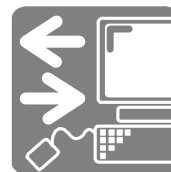
Fifth byte:



IMPORTANT

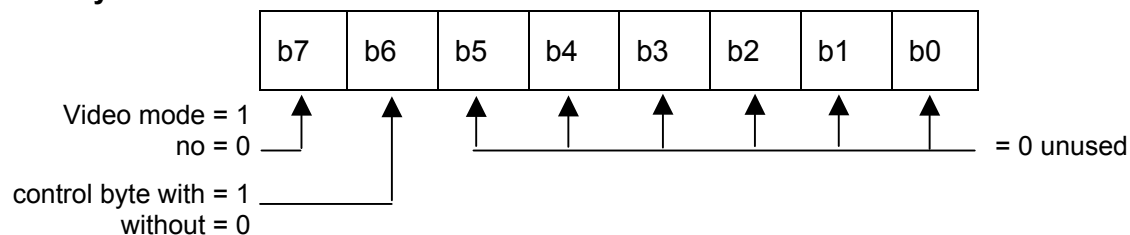
The value entered for a narrow element must always be less than the value entered for a wide element.

For distribution and Datamatrix codes, the value of narrow elements is set to 1 (09h) and the value of wide elements is set to 2 (22h).



Protocols - Precisions/Examples

Sixth byte:

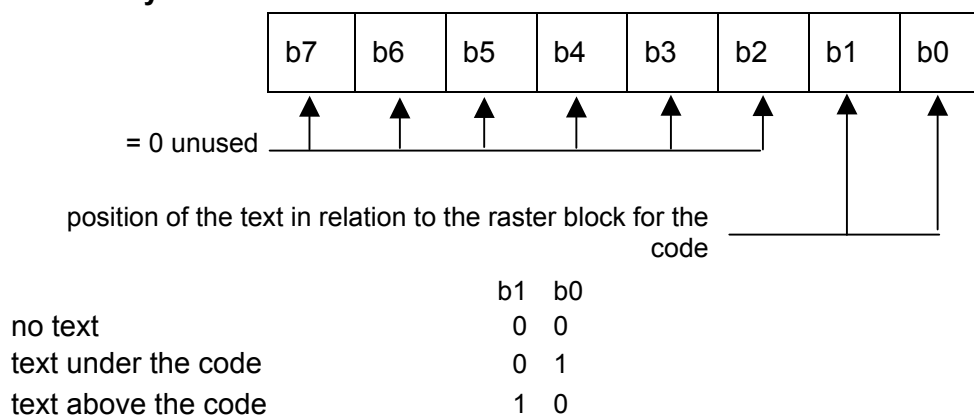


For the Datamatrix code, this byte is set to 40h.

IMPORTANT

If the control byte is used its position must be reserved in the text of the code. A control byte may never be the last character of a variable element.

Seventh byte:



The following three bytes are reserved and held at 00h.

■ Counter counting base

The presence of the “counter counting base” field adds 15×2 extra bytes to the message to represent the first and the last digit in the base of each counter.

The first and the last digit thus determine the counting base for each counter.

Examples:

First digit	Last digit	Bases	Counter type
0	9	10	Numeric
A	Z	26	Alphanumeric
0	Z	36	Alphanumeric
0	F	16	Alphanumeric

The interval between the start value and the end value must be no less than 4.
The start value entered must always be less than the end value.



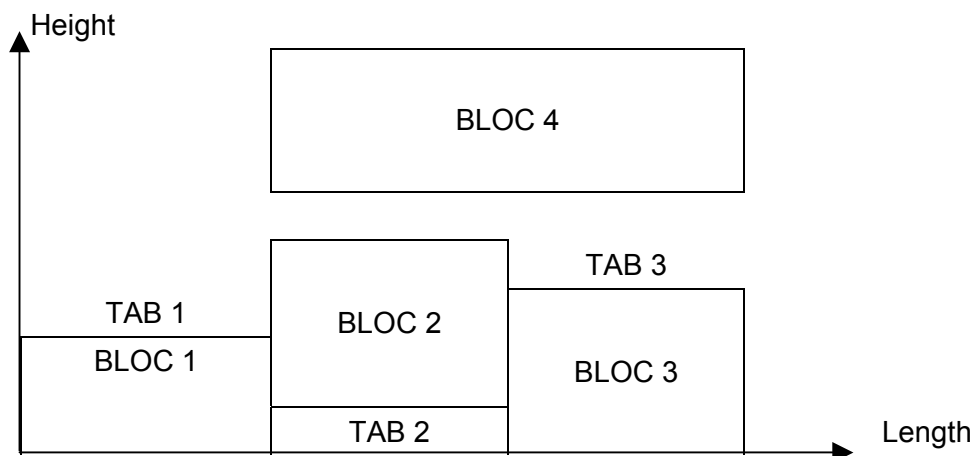


Protocols - Precisions/Examples

Definition of lines

A line is an assembly of blocks. The start line delimiter 0ah is used for each line. The end of message delimiter 0dh is used at the end of the message.

Example of message:



The message can be split up in three different ways:

- Either: line 1 consists of BLOCK1, BLOCK2, BLOCK3.
line 2 consists of TAB1 and BLOCK4.
- Or: line 1 consists of BLOCK1, BLOCK4.
line 2 consists of TAB1 and BLOCK2, BLOCK3.
- Or: line 1 consists of TAB1 and BLOCK4.
line 2 consists of TAB1, BLOCK2 and TAB3.
line 3 consists of BLOCK1, TAB2 and BLOCK3.

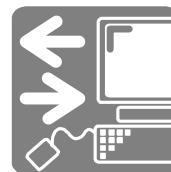
The number of variable lines is limited to 16 per message.

The lines are of variable size (e.g. 4 lines of 255 characters or 1 line of 1,020 characters)

Definition of blocks

A block is an assembly of elements made up of the same character generator, the same bolderization and the same vertical position.

2 bytes position 8xh, xxh	Characters generator 1 byte	Bolderi- zation 1 byte	Identification 1 byte 10h	TEXT	Identification 1 byte 10h	Bolderi- zation 1 byte	Characters generator (1 byte)	2 bytes position 8xh, xxh
---------------------------------	-----------------------------------	------------------------------	---------------------------------	------	---------------------------------	------------------------------	-------------------------------------	---------------------------------

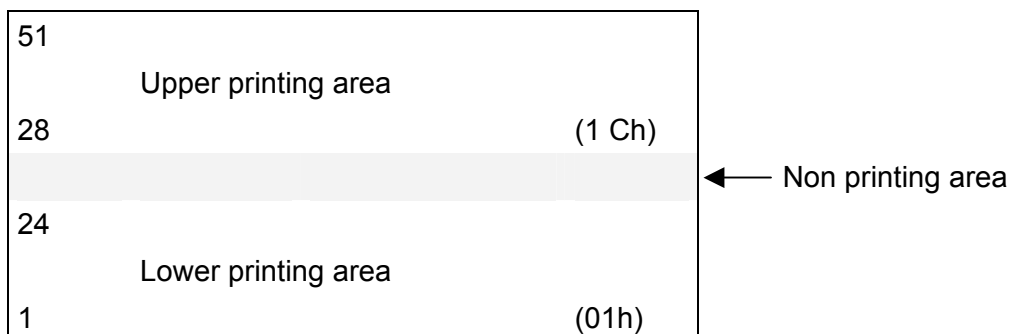


Protocols - Precisions/Examples

■ Position

These two bytes give the vertical position of the block, in number of drops, in relation to the first drop at the base of the character.

An area of 24 dots by jet is available. The first position of the lower area is 80h 01h and the first position of the upper area is 80h 1Ch



■ Character generator

Number of the character generator coded on 1 byte.

To enable CurvePrint fonts to be used:

- ☐ set algorithm 100 in the general message parameters,
- ☐ enter a Curveprint font number.

CurvePrint fonts are available on 9030 printers only.

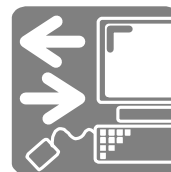
The list of character generator numbers is to be found in the APPENDIX.

■ Bolderization

This is the bolderization of the complete block from 1 to 9 in hexadecimal.

■ Identification

This is the identification of the start and end of 10h block.



Protocols - Precisions/Examples

Detail of the text

The text can be made of the following elements:

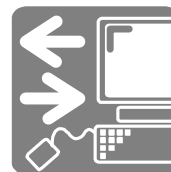
- ☐ alphanumeric characters,
- ☐ and/or counter,
- ☐ and/or autodating,
- ☐ and/or Datamatrix and bar codes,
- ☐ and/or tabulation,
- ☐ external variables,
- ☐ graphic block.

■ Alphanumeric characters

Characters coded in ASCII.

■ Counter

Delimiter 1 byte 1Ch	NUMBER OF COUNTER 01 to 15 1 byte - xxh	Delimiter 1 byte 1Ch
----------------------------	--	----------------------------



Protocols - Precisions/Examples

■ Autodating

Delimiter 1Ah	1 or more elements of autodating	Delimiter 1Ah
------------------	-------------------------------------	------------------

Autodating elements:

Autodating 1

41h	= seconds (00 to 59)	(2 hex bytes)
42h		
43h	= minutes (00 to 59)	(2 hex bytes)
44h		
45h	= hours (00 to 23)	(2 hex bytes)
46h		
47h	= mode 12/24 H (AM/PM/SPACE)	(2 hex bytes)
48h		
49h	= day of month (01 to 31)	(2 hex bytes)
4Ah		
4Bh	= day of year (001 to 366)	(3 hex bytes)
4Ch		
4Dh		
4Eh	= week of year (01 to 53)	(2 hex bytes)
4Fh		
50h	= month of year (01 to 12)	(2 hex bytes)
51h		
52h	= month of year in letters of	(3 hex bytes)
53h	language 1 (JAN to DEC)	
54h		
55h	= year (00 to 99)	(2 hex bytes)
56h		



Protocols - Precisions/Examples

Postdate 1

57h 58h	= postday of month (01 to 31)	(2 hex bytes)
59h 5Ah 5Bh	= postday of year (001 to 366)	(3 hex bytes)
5Ch 5Dh	= postweek of year (01 to 53)	(2 hex bytes)
5Eh 5Fh	= postmonth of year (01 to 12)	(2 hex bytes)
60h 61h 62h	= postmonth of year in letters of language 1 (JAN to DEC)	(3 hex bytes)
63h 64h	= postyear (00 to 99)	(2 hex bytes)
65h 66h 67h	= shift code 1 in letters (A to Z) = shift code 1 in numbers (01 to 99)	(1 hex byte) (2 hex bytes)
68h 69h 6Ah 6Bh 6Ch	= shift code 1 in letters (A to Z - O - I) = day of week in numbers 1 to 7 = postday of year modulo programmable	(1 hex byte) (1 hex byte) (3 hex bytes)
6Dh 6Eh 6Fh 70h 71h 72h 73h	= delimiter ":" = delimiter "/" = delimiter "." = delimiter " " = month of year in letters of language 2 (JAN to DEC)	(1 hex byte) (1 hex byte) (1 hex byte) (1 hex byte) (3 hex bytes)
74h 75h 76h	= postmonth of year in letters of language 2 (JAN to DEC)	(3 hex bytes)

The elements are received as bytes; these bytes are used to address a table containing the autodatating elements to be printed.

REMARQUE

The months of the year are in several languages (see C4h section "Sending language for printable month"). The default language is French.

The four delimiters (/ : .) are included in the timing elements to avoid a large number of blocks. These delimiters are fixed.



Protocols - Precisions/Examples

■ Extended autodating (see designation in Printer Manual)

Delimiter 1 byte 1Bh	One or more elements of autodating	Delimiter 1 byte 1Bh
----------------------------	---------------------------------------	----------------------------

Postdate 2

41h-42h	= postday of month (01 to 31)	(2 hex bytes)
43h-44h-45h	= postday of year (001 to 366)	(3 hex bytes)
46h-47h-48h	= postday of year modulo programmable (001 to 999)	(3 hex bytes)
49h-4Ah	= postweek of year (01 to 53)	(2 hex bytes)
4Bh-4Ch	= postmonth of year (01 to 12)	(2 hex bytes)
4Dh-4Eh	= postyear (00 to 99)	(2 hex bytes)
4Fh-50h-51h	= postday of week in letters (MON to SUN)	(3 hex bytes)
52h-53h-54h	= postmonth of year in letters (JAN to DEC)	(3 hex bytes)

Customizable tables

55h-56h-57h	= hours of the optional autodating	(3 bytes)
58h-59h-5Ah	= minute of the optional autodating	(3 bytes)
5Bh-5Ch-5Dh	= month of the optional autodating	(3 bytes)
5Eh-5Fh-60h	= day of week of the optional autodating	(3 bytes)
61h-62h-63h	= day of month of the optional autodating	(3 bytes)
64h-65h-66h	= day of year of the optional autodating	(3 bytes)
67h-68h-69h	= week of year of the optional autodating	(3 bytes)
6Ah-6Bh-6Ch	= year of the optional autodating	(3 bytes)
6Dh-6Eh-6Fh	= shiftcode 2 of the optional autodating	(3 bytes)
70h	= delimiter (:))	(1 byte)
71h	= delimiter (/)	(1 byte)
72h	= delimiter (.)	(1 byte)
73h	= delimiter ()	(1 byte)
74h-75h-76h	= shiftcode 3 of the optional autodating	(3 bytes)



Protocols - Precisions/Examples

Postdate 3

77h-78h	= postday of month (01 to 31)	(2 hex bytes)
79h-7Ah-7Bh	= postday of year (001 to 366)	(3 hex bytes)
7Ch-7Dh-7Eh	= postday of year modulo programmable	(3 hex bytes)
7Fh-80h	= postweek of year (01 to 53)	(2 hex bytes)
81h-82h	= postmonth of year (01 to 12)	(2 hex bytes)
83h-84h	= postyear (00 to 99)	(2 hex bytes)
85h-86h-87h	= postday of week in letters (MON to SUN) (customizable table of weeks of the year)	(3 hex bytes)
88h-89h-8Ah	= postmonth of year in letters (JAN to DEC) (customizable table of months of the year)	(3 hex bytes)

Postdate 4

8Bh-8Ch	= postday of month (01 to 31)	(2 hex bytes)
8Dh-8Eh-8Fh	= postday of year (001 to 366)	(3 hex bytes)
90h-91h-92h	= postday of year modulo programmable	(3 hex bytes)
93h-94h	= postweek of year (01 to 53)	(2 hex bytes)
95h-96h	= postmonth of year (01 to 12)	(2 hex bytes)
97h-98h	= postyear (00 to 99)	(2 hex bytes)
99h-9Ah-9Bh	= postday of week in letters (MON to SUN) (customizable table of weeks of the year)	(3 hex bytes)
9Ch-9Dh-9Eh	= postmonth of year in letters (JAN to DEC) (customizable table of months of the year)	(3 hex bytes)



Protocols - Precisions/Examples

Postdate 5

9Fh-A0h	= postday of month (01 to 31)	(2 hex bytes)
A1h-A2h-A3h	= postday of year (001 to 366)	(3 hex bytes)
A4h-A5h-A6h	= postday of year modulo programmable	(3 hex bytes)
A7h-A8h	= postweek of year (01 to 53)	(2 hex bytes)
A9h-AAh	= postmonth of year (01 to 12)	(2 hex bytes)
ABh-ACH	= postyear (00 to 99)	(2 hex bytes)
ADh-AEh-AFh	= postday of week in letters (MON to SUN) (customizable table of months of the year)	(3 hex bytes)
B0h-B1h-B2h	= postmonth of year in letters (JAN to DEC) (customizable table of months of the year)	(3 hex bytes)

Postdate 6

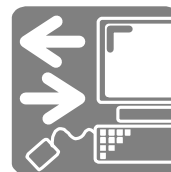
B3h-B4h	= postday of month (01 to 31)	(2 hex bytes)
B5h-B6h-B7h	= postday of year (001 to 366)	(3 hex bytes)
B8h-B9h-BAh	= postday of year modulo programmable	(3 hex bytes)
BBh-BCh	= postweek of year (01 to 53)	(2 hex bytes)
BDh-BEh	= postmonth of year (01 to 12)	(2 hex bytes)
BFh-C0h	= postyear (00 to 99)	(2 hex bytes)
C1h-C2h-C3h	= postday of week in letters (MON to SUN) (customizable table of weeks of the year)	(3 hex bytes)
C4h-C5h-C6h	= postmonth of year in letters (JAN to DEC) (customizable table of months of the year)	(3 hex bytes)
1Bh	delimiter of autodatting	(1 hex byte)

NOTE

The elements are received as bytes; these bytes are used to address a table containing the autodatting elements to be printed.

Use of tables:

- ☐ *Each parameter has an associated table of 1 to 3 characters. The number of characters printed depends on the message elements.*
 - ☐ *If an element of the table = 00, it is not printed.*
-



Protocols - Precisions/Examples

■ Datamatrix and bar codes

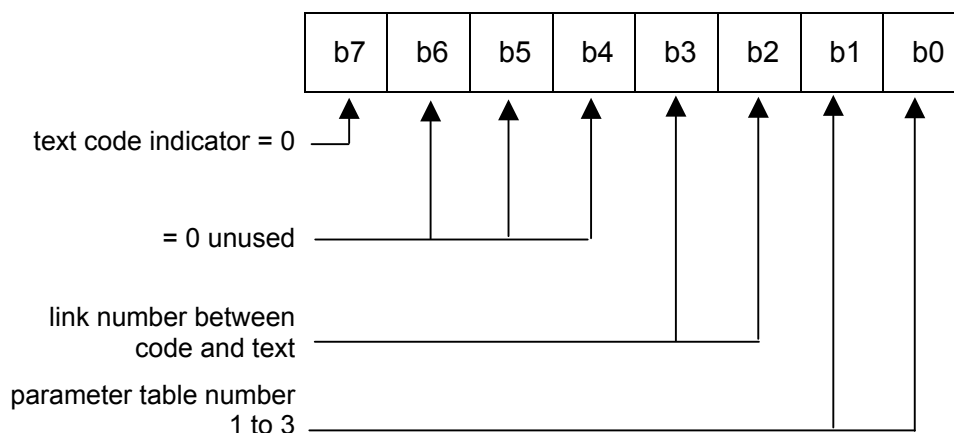
ATTENTION Bar codes must be entered in the appropriate block.

Code to be printed in cipher:

Delimiter 1Fh	definition of the code 1 byte 1 to 3	Element to be encoded	Delimiter 1Fh
------------------	--	--------------------------	------------------

Definition of the code:

A maximum of 3 codes may be printed per message. These three codes are parametered in the section "Variable Element Parameters, in order of code 1, code 2, code 3. This code number allows choice of a code together with its characteristics (height, width, etc).



- Maximum length of standard codes: 32 characters with check byte.
- Length of EAN and UPC codes (fixed).
 - 13 digits for EAN-13, ½ EAN-13, UPCA and ½ UPCA codes.
 - 8 digits for EAN-8, ½ EAN-8, UPCE and ½ UPCE codes.
- If the check byte is in the text, its location is reserved. This check byte is always a fixed character.

ATTENTION In the case of EAN-128 codes and code 128, add 20h to the characters to be encoded (except the 4 delimiters).

NOTE The entire bar code (which is seen as a unique character) is affected when the horizontal direction parameters are set.



Protocols - Precisions/Examples

Element to be encoded:

□ Symbols:

SYMBOLS in characters ASCII

□ Timer elements:

Delimiter 1Ah	AUTODATING	Delimiter 1Ah
------------------	------------	------------------

See previous paragraph "Autodating"

□ Extended timer elements:

Delimiter 1Bh	EXTENDED AUTODATING	Delimiter 1Bh
------------------	------------------------	------------------

See previous paragraph "Autodating"

□ Counters:

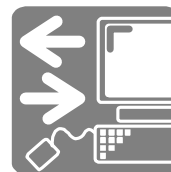
Delimiter 1Ch	Counter number from 01 to 15 1 byte xxh	Delimiter 1Ch
------------------	---	------------------

□ External variables:

Delimiter 12h	Text to be modified n bytes	Delimiter 12h
------------------	--------------------------------	------------------

□ Maximum encoding capacity of Datamatrix code

Bolderization	1 droplet cell					2 droplet cell		
Code height (dots)	9		11	16	24	16		24
Print Matrix	8 x 18	8 x 32	10 x 10	16 x 48	24 x 24	8 x 18	8 x 26	12 x 36
Number of encoded numeric characters	10	20	6	98	72	10	20	44
Number of encoded alphanumeric	6	13	3	72	52	6	11	31
Number of encoded 8-bit ASCII characters	3	8	1	47	34	3	13	20



Protocols - Precisions/Examples

Code to be printed in plain text:

Delimiter 1Fh	Definition of the code 1 byte 1 to 3	Element of code to be printed in plain text n bytes	Delimiter 1Fh
------------------	--	---	------------------

This code to be printed unencoded may only be used for distribution codes

Bit b7 of the code definition byte (text code indicator) is set to 1. The contents is identical to the encoded code printed, with an additional element to be encoded: the tabulation.

The text may contain fixed or variable elements.

The location of the control byte is reserved at the end of the text.

Spaces between characters must be replaced by tabulation blocks.

CAUTION

For EAN 128 and code 128, add 20h to the characters to be encoded (except for the 4 delimiters).

■ Tabulation

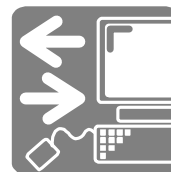
Horizontal block position is allowed by the tabulations, made with empty rasters.

Please refer to the sub-chapter "Lines definition" and the example "Sending a message for printing".

Delimiter 1Eh	Number of empty raster 1 to 255 1 byte	Delimiter 1Eh
------------------	--	------------------

NOTE

No limitation in the number of tabulations and bolderization is not taken into account..



Protocols - Precisions/Examples

■ External variables

This block is used to mark a portion of text to be modified from the serial link. To use an external variable block, it must first be defined in the current message. The text to be modified may comprise any printable elements.

If a variable element is contained in an external variable, this must be complete.

The structure of the message may not be modified.

There may be a maximum of 10 variables in a message.

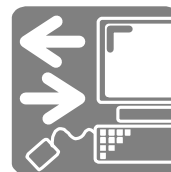
If a block is defined and is not to be modified, the text to modify zone must be zero bytes.

Delimiter 12h	Text to be modified n bytes	Delimiter 12h
------------------	--------------------------------	------------------

NOTE

When a bar code with unencoded text is printed, containing external variable zones, it is essential to separate these zones by fixed elements.

A command must be repeated in order to update the value of the bar code and its associated plain text.



Protocols - Precisions/Examples

■ Graphic block

This block may be used to insert graphics in a message to be printed.

The block may not be edited on the terminal.

The current bolderization is not taken into account when printing this block.

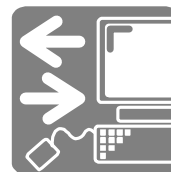
Delimiter 11h	Number of drops for the graph (height) 1 byte	Number of descriptors (width) 2 bytes	Descriptor to be printed n bytes	Number of descriptors (width) 2 bytes	Number of drops for the graph (height) 1 byte	Delimiter 11h
------------------	---	--	---	--	---	------------------

Example: protocol to send a graphic block with a height of eight points

	FFh	50h	01h	07h	
11h	●	●	●		B7
08h	●	●	●		B6
00h/08h	●		●		B5
FFh/90h/50h/F0h/01h/03h/07h/F7h	●	●	●		B4
00h/08h	●				B3
08h	●			●	B2
11h	●			●	B1
	●		●	●	B0
	90h	F0h	03h	F7h	

End of message

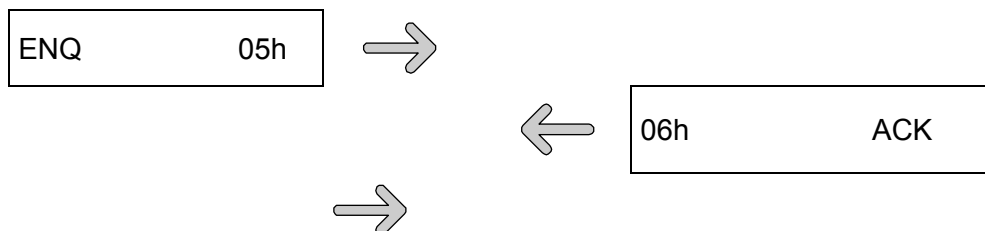
The end of message delimiter is 0Dh.



Protocols - Precisions/Examples

■ Examples

Sending a complete message

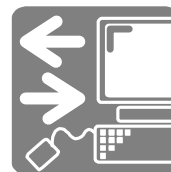


Identification		E3h
Length		00h 62h
Data:		
- <u>Structure indicator:</u> (presence of general parameters) (presence of message text)		C0h 00h
- <u>General parameters:</u>		
message sense	normal	10h
horizontal sense	normal	
vertical sense	normal	
tacho mode	yes	
set off	object	
unit	mm	
- Multitop start mode:	non	00h
- Tacho division	05	00h 05h
- Forward margin	16	00h 10h
- Return margin	3	00h 03h
- Interval	2	00h 02h
- Printing speed	256 mm/s	01h 00h
- Reserved	0	00h 00h



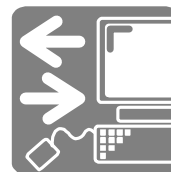
Protocols - Precisions/Examples

- <u>Text</u> :	
Identification of line 1	0Ah
First block position	80h 01h
Character generator 018 (d)	12h
Bolderization 1	01h
Text delimiter	10h
P	50h
R	52h
O	4Fh
D	44h
U	55h
I	49h
T	54h
	20h
L	4Ch
E	45h
	20h
Auto-dating delimiter	1Ah
day of month	49h 4Ah
Separator	6Eh
month	50h 51h
Separator	6Eh
years	55h 56h
Autodating delimiter	1Ah
Text delimiter	10h
Bolderization 1	01h
Characters generator 013 (d)	12h
First block position	80h 01h
Second block position	80h 01h
Characters generator 013 (d)	0Dh
Bolderization 2	02h



Protocols - Precisions/Examples

Text delimiter	10h
	20h
P	50h
O	4Fh
I	49h
D	44h
S	53h
	20h
2	32h
	20h
K	4Bh
G	47h
Text delimiter	10h
Bolderization 2	02h
Characters generator 013 (d)	0Dh
Second block position	80h 01h
Identification of second line	0Ah
First block position	80h 0Ah
Characters generator 013 (d)	0Dh
Bolderization 1	01h
Text delimiter	10h
Tabulation delimiter	1Eh
Rasters number: 240	F0h
Tabulation delimiter	1Eh



Protocols - Precisions/Examples

M	4Dh
A	41h
D	44h
E	45h
	20h
I	49h
N	4Eh
	20h
F	46h
R	52h
A	41h
N	4Eh
C	43h
E	45h
Text delimiter	10h
Bolderization 1	01h
Characters generator 013 (d)	0Dh
First block position	80h 0Ah
End message delimiter	0Dh
Checksum	B8h

Printing message is



06h

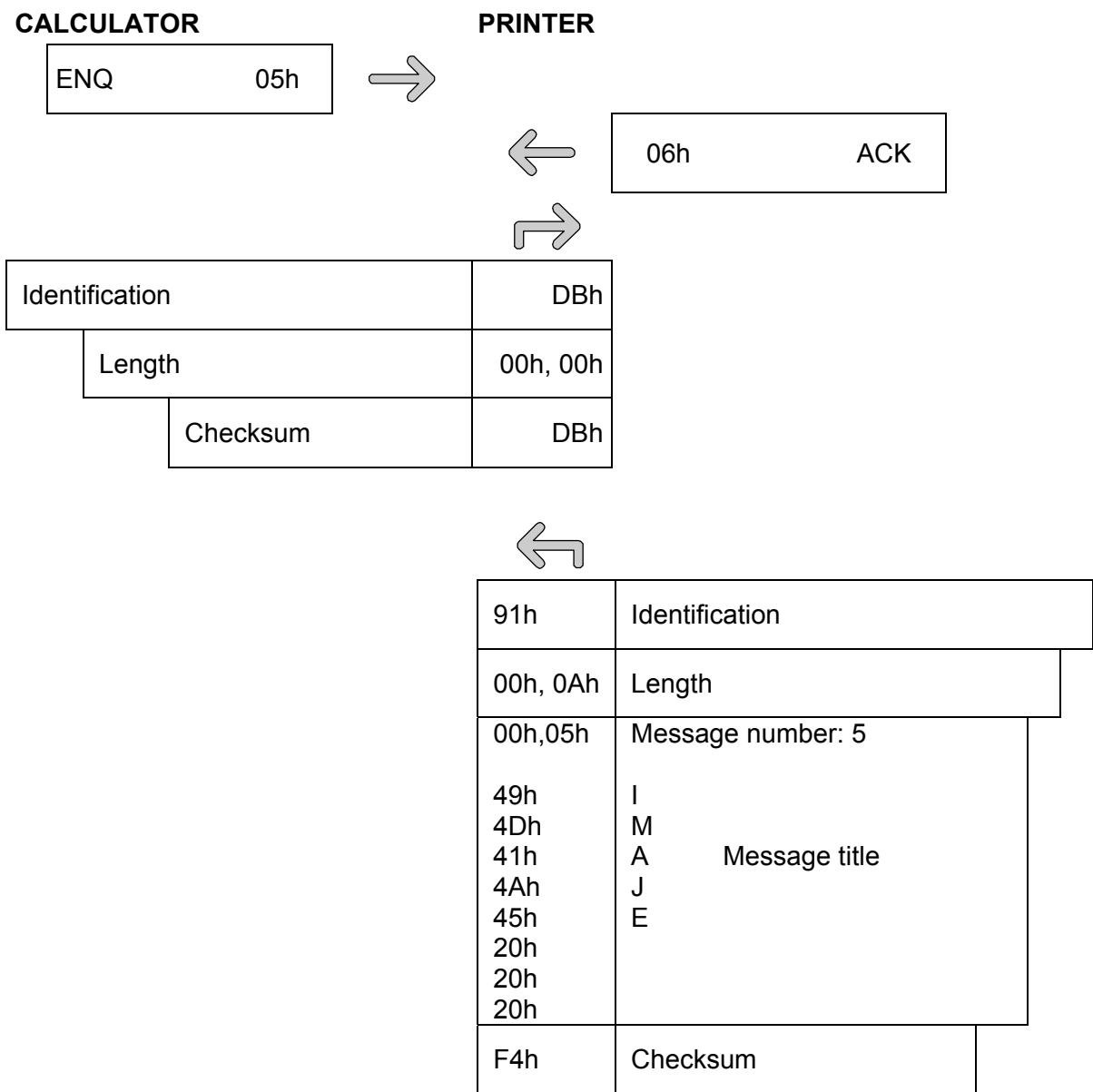
ACK

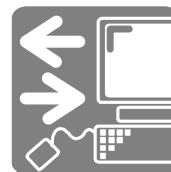
PRODUIT LE 30/09/00 MADE IN FRANCE
POIDS 2 KG

Protocols - Precisions/Examples



Request for a message active number





Protocols - Precisions/Examples

Sending a partial message

- Last printing message

PRODUIT LE 14/01/00 MADE IN FRANCE
POIDS 2 KG

- Modification of the message by sending a partial message.

Identifiant	E5h
Length	00h 1Eh
Number of area to be amended	03h
Line number	00h
Position in line	00h 05h
Number of characters to be amended	00h 07h
E	45h
M	4Dh
B	42h
A	41h
L	4Ch
L	4Ch
E	45h
Line number	00h
Position in line	00h 2Bh
Number of characters to be amended	00h 01h
3	33h
Line number	01h
Position in line	00h 10h
Number of characters to be amended	00h 06h
S	53h
U	55h
I	49h
S	53h
S	53h
E	45h
Checksum	B0h

- New printing message:

EMBALLE LE 14/01/00 MADE IN SUISSE
POIDS 3 KG



Appendix 1



Appendix 1



Appendix 1

■ Reminder

Binary, decimal, hexadecimal conversion

Decimal	Binary	Hexadecimal
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

7	6	5	4	3	2	1	0
0	0	1	1	1	0	1	0

3

A

i. e. 3Ah

A byte (8 bits) is made of two hexadecimal figures.

The symbol "h" refers to hexadecimal notation.

The ASCII standard allows association of an alphanumeric symbol with each byte.

41h corresponds, for example, to "A" in ASCII.



Appendix 1

■ ASCII table of characters and symbols used for a message



	0	1	2	3	4	5	6	7	8	9	A
0	NUL		SP	0	@	P	\	p	Ç	É	á
1			!	1	A	Q	a	q	ü	æ	í
2			"	2	B	R	b	r	é	Æ	ó
3			#	3	C	S	c	s	â	ô	ú
4			\$	4	D	T	d	t	ä	ö	ñ
5			%	5	E	U	e	u	à	ò	Ñ
6			&	6	F	V	f	v	â	û	œ
7			'	7	G	W	g	w	ç	ü	¿
8			(	8	H	X	h	x	ê	ù	§
9			)	9	I	Y	i	y	ë	œ	
A			*	:	J	Z	j	z	è	Ö	
B			+	;	K	[	k	{	ï	Ü	
C	FF		,	<	L	\	l		î	ç	
D	CR		-	=	M	]	m	}	ì	£	
E			.	>	N	^	n	~	Ä	Ø	
F			/	?	O	_	o		Å	ø	

Example:

ASCII character	I	M	A	J	E
Code hex	49h/	4Dh/	41h/	4Ah/	45h



Appendix 1

■ Fonts list

Standard fonts

Characters generators	Number	9020	9030	Size of the matrix
Latin Ind. 5	81	X	X	5 x 6
Latin Ind. 7	13	X	X	7 x 6
Latin Ind. 9	96	X	X	9 x 6
Latin Ind. 11	39	X	X	11 x 8
Latin Ind. 16	18	X	X	16 x 12
Latin Ind. 24	20	X	X	24 x 21
Latin Ind. 48	168		X	48 x 37
Latin Ind. 48C	169		X	48 x 39
Chimney 5	21	X	X	5 x 6
Chimney 7	23	X	X	7 x 8
Chimney 11	85	X	X	7 x 11
Chimney 16	87	X	X	11 x 18
Arabic 7	199	X	X	7 x 6
Arabic 7	198	X	X	7 x 6
Arabic 16	189	X	X	16 x 12
Arabic 16	188	X	X	16 x 12
Arabic 24	184	X	X	24 x 20
Cyrillic 7	35	X	X	7 x 7
Cyrillic 16	37	X	X	16 x 13
Cyrillic 24	38	X	X	24 x 18
Greek 7	88	X	X	7 x 6
Greek 16	90	X	X	16 x 12
Greek 24	91	X	X	24 x 21
Hebrew 5	97	X	X	5 x 6
Hebrew 7	183	X	X	7 x 6
Hebrew 7	182	X	X	7 x 6
Hebrew 16	181	X	X	16 x 12
Hebrew 16	180	X	X	16 x 12
Hebrew 24	95	X	X	24 x 20
Japanese 7	187	X	X	7 x 19
Japanese 11	186	X	X	11 x 27
Japanese 24	185	X	X	24 x 50
Korean 7	192	X	X	7 x 8
Korean 9	191	X	X	9 x 7
Korean 11	190	X	X	11 x 8
Thai mono	49	X	X	7 x 6
Thai bij	48	X	X	7 x 6
Chinese 11	195	X	X	11 x 11
Chinese 16	194	X	X	16 x 16
Chinese 24	193	X	X	24 x 22
py01_12	40	X	X	
py02_12	41	X	X	
py03_12	42	X	X	
py04_12	43	X	X	
py05_12	44	X	X	
py06_12	45	X	X	
py07_12	50	X	X	
py08_12	51	X	X	
py09_12	52	X	X	



Appendix 1

Characters generators	Number	9020	9030	Size of the matrix
py10_12	53	X	X	
py11_12	54	X	X	
py12_12	55	X	X	
py13_12	56	X	X	
py14_12	57	X	X	
py15_12	58	X	X	
py16_12	59	X	X	
py17_12	60	X	X	
py18_12	61	X	X	
py19_12	62	X	X	
py20_12	63	X	X	
py21_12	64	X	X	
py22_12	65	X	X	
py23_12	66	X	X	
py24_12	67	X	X	
py25_12	68	X	X	
py26_12	69	X	X	
py27_12	70	X	X	
py28_12	71	X	X	
py29_12	72	X	X	
py30_12	73	X	X	
py31_12	74	X	X	
py01_16	76	X	X	
py02_16	78	X	X	
py03_16	79	X	X	
py04_16	80	X	X	
py05_16	82	X	X	
py06_16	83	X	X	
py07_16	84	X	X	
py08_16	86	X	X	
py09_16	89	X	X	
py10_16	93	X	X	
py12_16	99	X	X	
py13_16	100	X	X	
py14_16	101	X	X	
py15_16	102	X	X	
py16_16	103	X	X	
py17_16	104	X	X	
py18_16	105	X	X	
py19_16	106	X	X	
py20_16	107	X	X	
py21_16	108	X	X	
py22_16	109	X	X	
py23_16	110	X	X	
py24_16	111	X	X	
py25_16	112	X	X	
py26_16	113	X	X	
py27_16	114	X	X	
py28_16	115	X	X	
py29_16	116	X	X	
py30_16	117	X	X	
py31_16	118	X	X	
py11_16	119	X	X	



Appendix 1

CurvePrint fonts

Characters generators	Number	9020	9030	Size of the matrix
CurvePrint 7	128 / G		X	7 x 6
CurvePrint 1 x 8	121 / G		X	8 x 6
CurvePrint 2 x 8	120 / G		X	8 x 6
CurvePrint 11	125 / G		X	11 x 8
CurvePrint 16	124 / G		X	16 x 12
CurvePrint 7	144 / M		X	7 x 6
CurvePrint 1 x 8	137 / M		X	8 x 6
CurvePrint 2 x 8	136 / M		X	8 x 6
CurvePrint 11	141 / M		X	11 x 8
CurvePrint 16	140 / M		X	16 x 12
CurvePrint 7	143		X	7 x 6

■ Algorithmes list

N°	G Head	N°	M Head
001	G10_05_01_28_05	006	M08_05_01_45_06
002	G10_07_01_28_07	007	M08_07_01_45_08
003	G10_16_01_28_17	008	M08_07_01_45_26
004	G10_07_01_28_18	009	M08_16_01_45_53
005	G10_16_01_28_40	010	M08_16_01_45_19
011	G10_11_01_28_12	015	M08_11_01_45_16
012	G10_24_01_28_60	016	M08_24_01_45_78
013	G30_07_01_28_23	017	M20_07_01_45_32
014	G30_16_01_28_57	018	M20_11_01_45_48
019	G10_09_01_28_22	022	M08_11_01_45_37
020	G10_11_01_28_26	033	M08_07_01_45_07
999	G SP EAN	035	M08_3x5_1_45_20
030	G10_3x5_1_28_16	036	M08_3x7_1_45_26
031	G10_07_01_28_06	037	M08_4x5_1_45_26
032	G10_05_01_28_04	038	M08_24_01_45_31
040	G10_3x7_1_28_22	042	M08_05_01_45_05
041	G10_4x5_1_28_24	044	M08_07_01_28_08
043	G10_24_01_28_28	045	M08_05_01_28_05
047	G10_2x7_1_28_15	049	M08_2x5_1_28_13
048	G10_2x5_1_28_11	996	M SP EAN_24
050	G10_4x7_1_28_37 *		

* Single-jet only



Appendix 1

■ Performance

The performance of the serial link of the printer is measured over a sample of ten messages:

- Message 01: 1 line of 20 static characters
- Message 02: 2 line of 20 static characters
- Message 03: 3 line of 20 static characters
- Message 04: 4 line of 20 static characters
- Message 05: 5 line of 20 static characters
- Message 06: 6 line of 20 static characters
- Message 07: 1 line with a 9-digit counter
+ 1 line with 3 timer elements
- Message 08: 1 line with a 9-digit counter
+ 1 line with 3 timer elements
+ 1 bar code 2/5 of 10 static characters
- Message 09: 1 line with a 9-digit counter
+ 1 line with 3 timer elements
+ 1 bar code 2/5 of 10 static characters
+ 1 line with a EAN 13 code
- Message 10: 1 line with a EAN 13 code

The serial link configuration is as follows:

- 19 200 bauds,
- 1 start bit, 8 bits, 1 stop bit without parity.

Time to send a message from the computer to the printer:

$$T = \frac{\text{Nbr of bytes transmitted} \times \text{Nbr of bits per bytes transmitted}}{\text{Transmission speed}} \text{ in seconds}$$



Appendix 1

■ Result

	WITH EXTENDED LIBRARY OPTION		
	Message sent to library	Message sent to print	Message number selection
Message 01	47 ms	39 ms	15 ms
Message 02	64 ms	56 ms	16 ms
Message 03	81 ms	73 ms	16 ms
Message 04	91 ms	89 ms	17 ms
Message 05	115 ms	106 ms	18 ms
Message 06	132 ms	123 ms	19 ms
Message 07	67 ms	58 ms	16 ms
Message 08	87 ms	77 ms	17 ms
Message 09	97 ms	89 ms	19 ms
Message 10	75 ms	67 ms	16 ms

□ Message sent to library:

The time covers reception, library writing, and processing of the data prior to authorization of the TOP object.

NOTE

If the message sent to the library by serial link is the active message, the next print will be that of the new message.

□ Message sent to print:

The time covers reception and processing of the data prior to authorization of the TOP object. The message is stored in the library.

□ Selection of message number:

The time covers the selection of a message number, library reading, and processing of the data prior to authorization of the TOP object.



Appendix 2



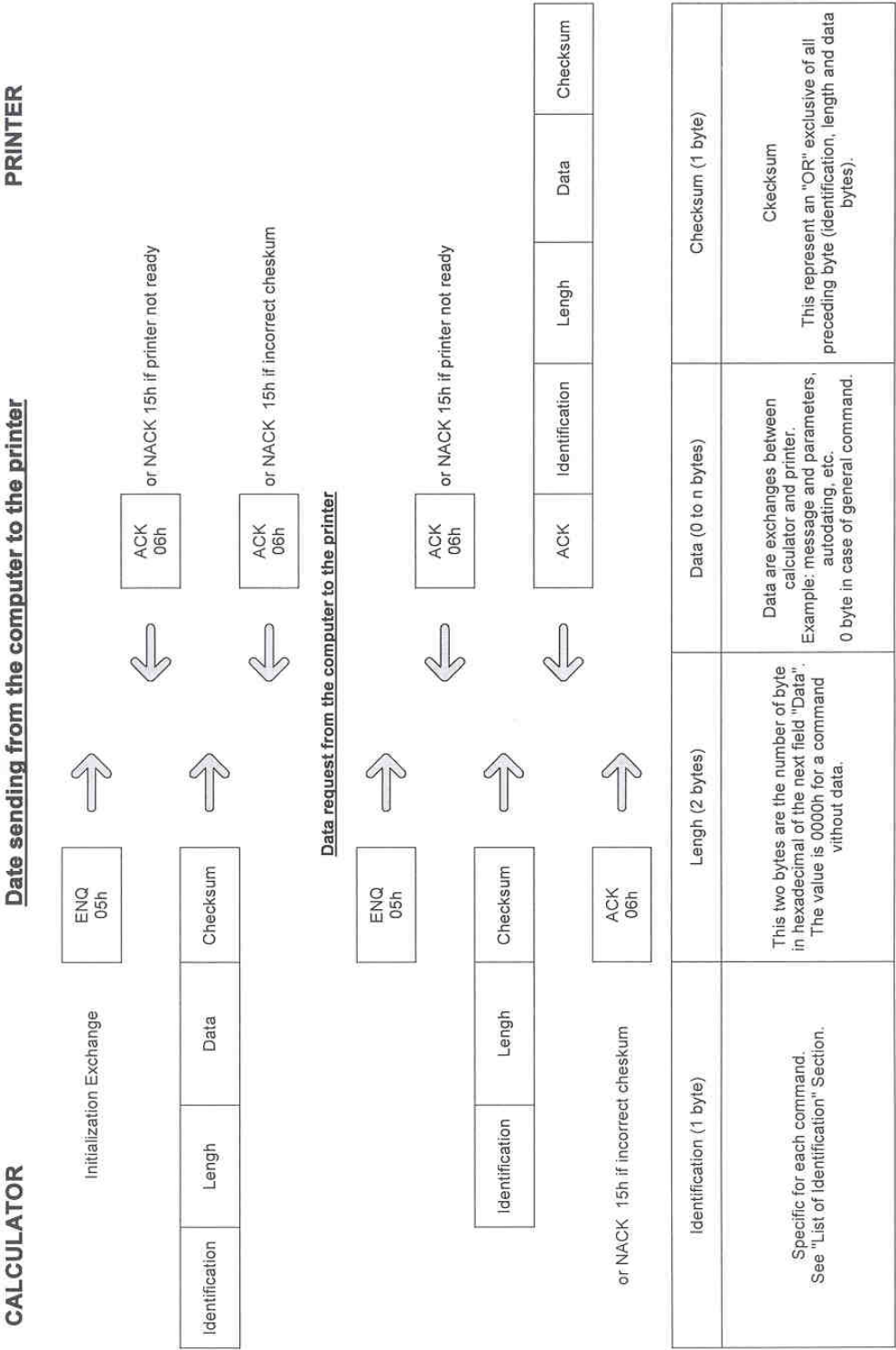
Appendix 2



A-1000000 COMPANY

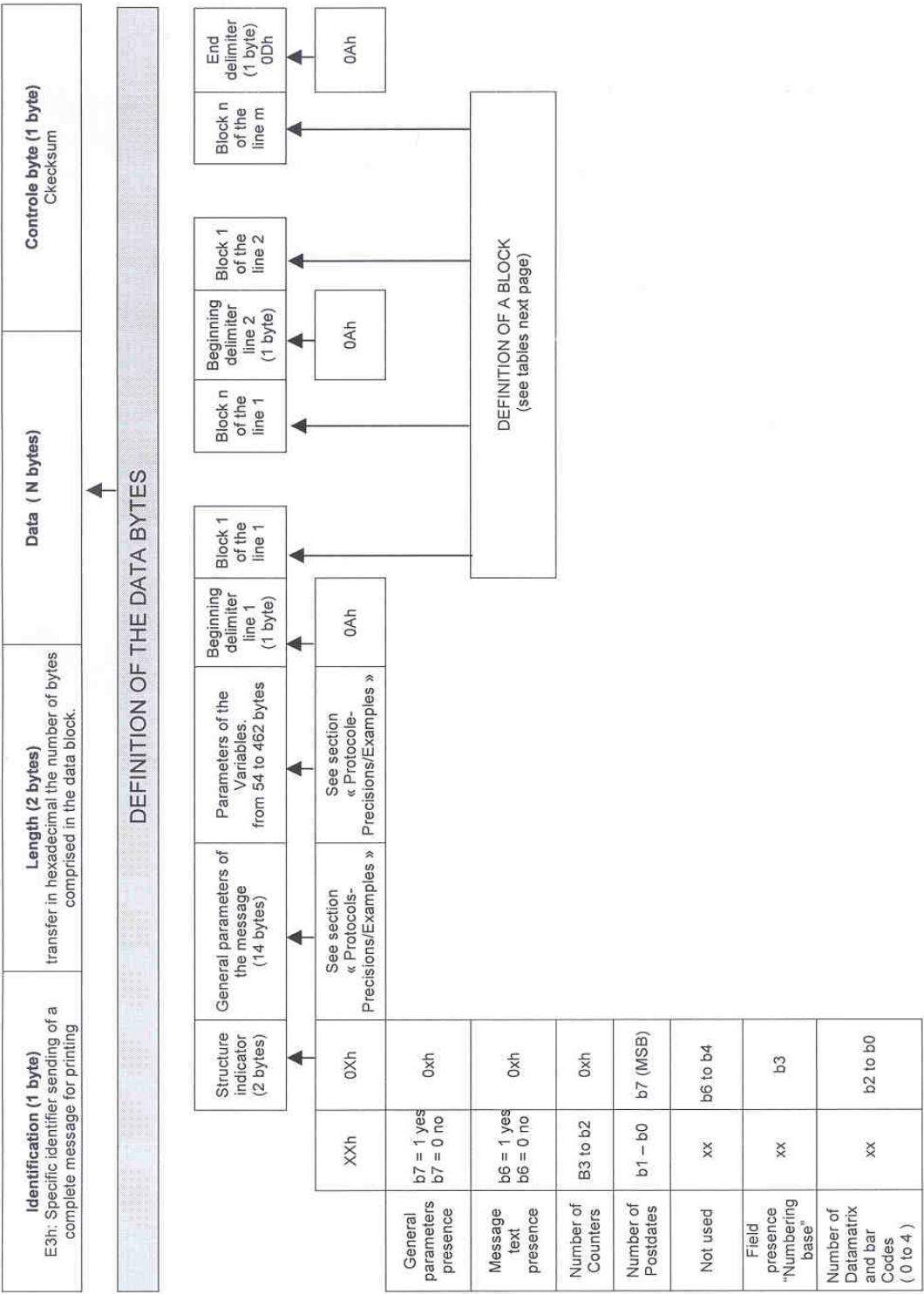


■ General principle of an exchange



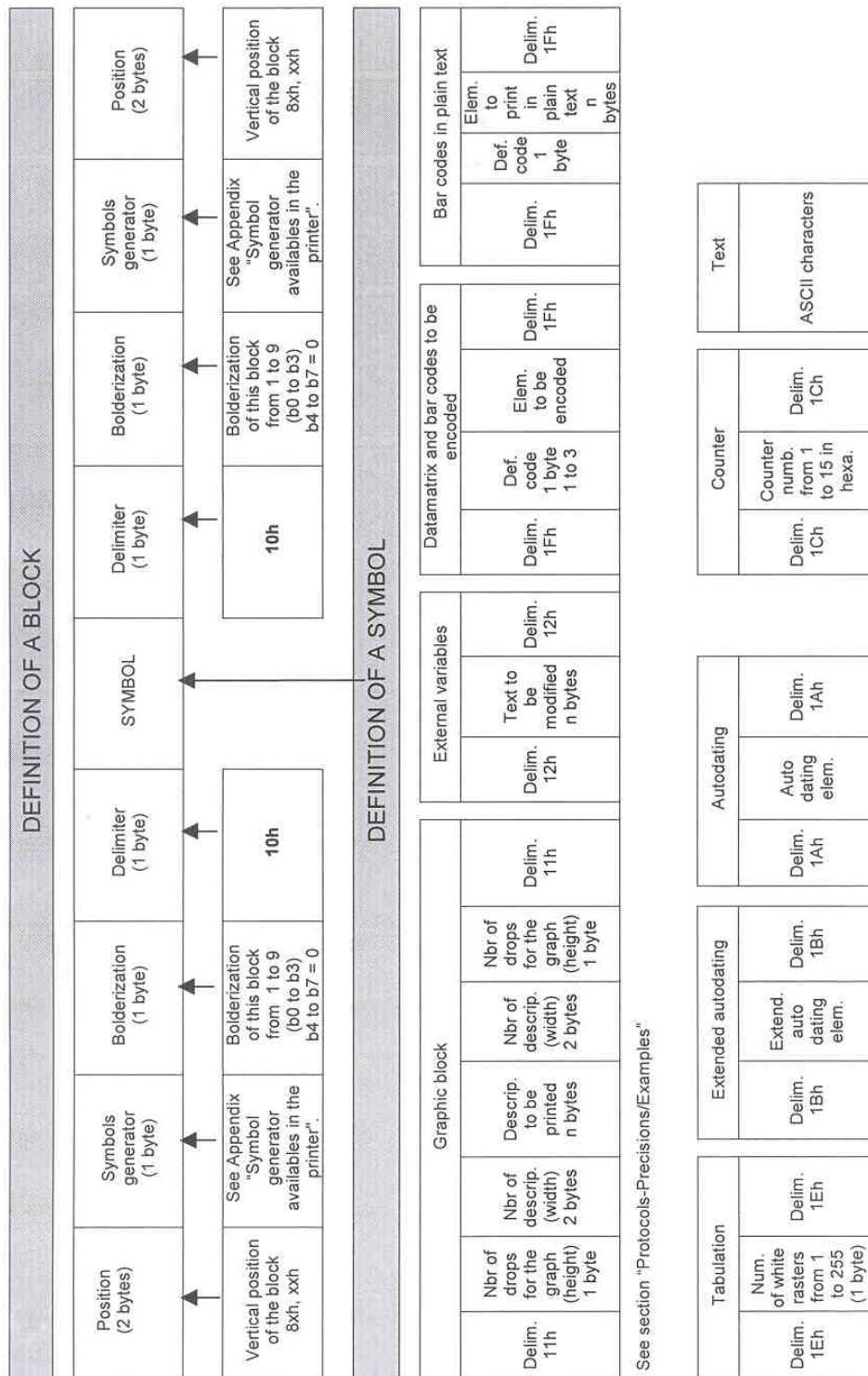


■ Sending of a complete message





Appendix 2





Message selection by Parallel interface



Message selection by Parallel interface



Message selection by Parallel interface

■ External interface - Introduction

Parallel interface is available with 9030 printer.

It allows selection of messages contained in the printer library to be selected for printing by PARALLEL INTERFACE or by MESSAGE INCREMENT.

Both types of message selection are connected to the printer with the Industrial Interface Card.

Configuration

The type of message selection used is to be configured in the

► *Printer Preparation/Initialization/Message/Miscellaneous/Operation:*

std message: message selection by Operator Interface.

select n mes: message selection by PARALLEL INTERFACE.

library: message selection by MESSAGE INCREMENT.

S7: operation with Series 7 printer message structure.

Characteristics

■ Inputs

All input circuits are carried out with the help of the high speed photocoupler.

Voltage used: from 15 to 35 volts.

Input current regulated in the operational voltage field.

The D0 to D7 inputs have a common connection "COM".

The inputs are active at high level.

■ Outputs

Two outputs are used:

□ "SYNCMESS" message synchronization output,

□ "ETATIMP" counter status output.

These outputs are carried out with the help of open collector rapid photocoupler.

Maximum output current: 16 mA.

Maximum operational voltage: 24 V.

The outputs are active at low level.

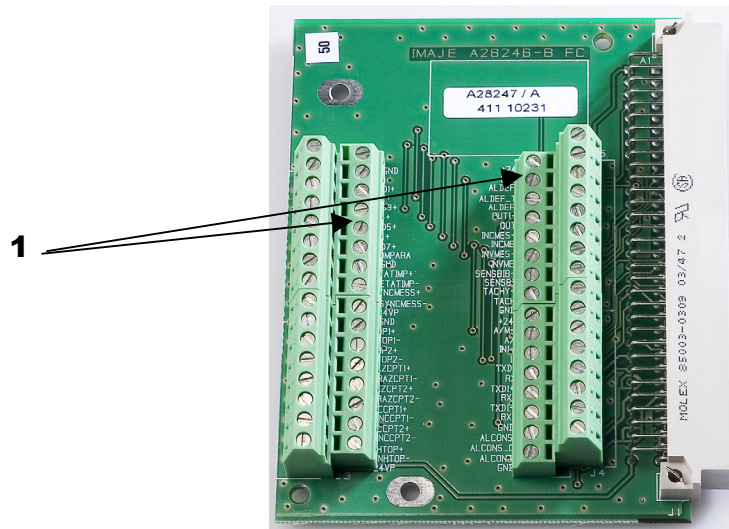


Message selection by Parallel interface

■ Connection

Both types of message selection are connected to the industrial interface terminal block board in the same way as the serial link.
See the “General-Connection” section.

Connect the wires to the corresponding terminals on the J2 or J5 connector (1) depending on the type of message selection used.



■ Parallel Interface

INCMES+
INCMES-

Message increment

D0
D1
D2
D3
D4
D5
D6
D7

Parallel Interface Data

COMPARA

SYNCMES+
SYNCMES-

Message Synchronization Output

ETATIMP+
ETATIMP-

Counter status Output



Message selection by Parallel interface

■ Message Increment

INCMES+	Message increment
INCMES-	
SENSBIB+	Message increment or decrement flow
SENSBIB-	
TOP+	Print start
TOP-	
SYNCMESS+	Message synchronization output
SYNCMESS-	

□ Supply:

0 V

24 V

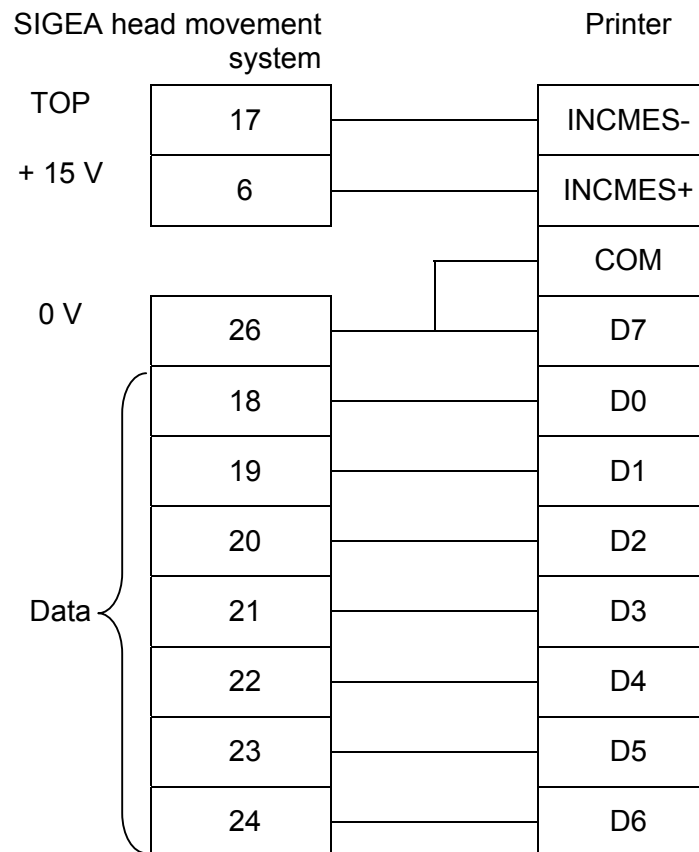
IMPORTANT

The shielding of the link cable used must be connected throughout the length of the metallic stuffing box and a similar connection must be made on the computer.



Message selection by Parallel interface

Connection of the parallel interface to a SIGEA head movement





Message selection by Parallel interface

■ Message selection by parallel interface

Operation

■ Inputs

INCMES: This input allows validation of the message number present on the data D0 to D7 and prints the message (this input is not filtered).

D0 to D7: Data for addressing the message number to be printed.

D7 = 1 the addresses are coded in BCD (up to 79 messages).

D7 = 0 the addresses are coded in hexadecimal (up to 127 messages).

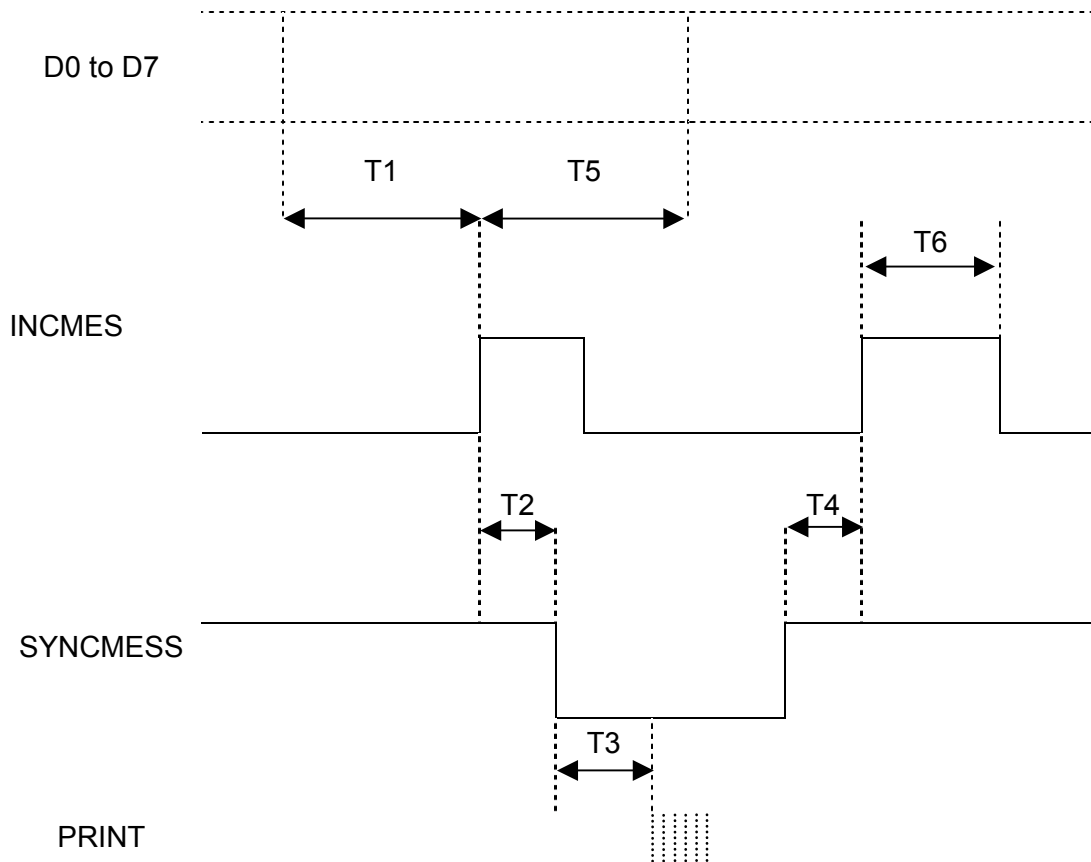
■ Output

SYNCMESS: This output is active during printing of the selected message.



Message selection by Parallel interface

■ Signals Diagram



- T1: Minimum time of presence of data D0 to D7 before the rise of the signal INCMES = 30 μ s.
- T2: Maximum time between the rise of the INCMES signal and the active state of the signal SYNCMESS = 10 ms (this time depends on the message content).
- T3: Time between the activation of the SYNCMESS signal and start of the message printing.(this time depends on the type of head, the printing speed and delay of the selected message).
- T4: Minimum time between the end of printing (SYNCMESS signal inactive) and a new message selection = 10 ms.
- T5: Minimum hold time for data D0 to D7 after the rise signal INCMES = 30 μ s.
- T6: Minimum width of the INCMES = 100 μ s.



Message selection by Parallel interface

■ Message selection by message increment interface

There are two operational modes with this type of Message Selection according to the signal present on the D7 bit:

D7 to 0: Parallel Interface Mode,

D7 to 1: Message Increment Mode.

IMPORTANT

The role of the Parallel Interface Mode corresponds above all to the initialization of the first of a series of messages during the Message Selection by Message Increment.

NOTE

These two operational modes can be used in conjunction depending on the D7 input level.

Parallel Interface Mode

■ Inputs

DATA D0 to D6:

Message number to be printed in hexadecimal.

INCMES:

This input allows validation of the message number present on the D0 to D6 data.

TOP1:

This input permits print start up of the message selected by the D0 to D6 data and validated by the INCMES input.

■ Output

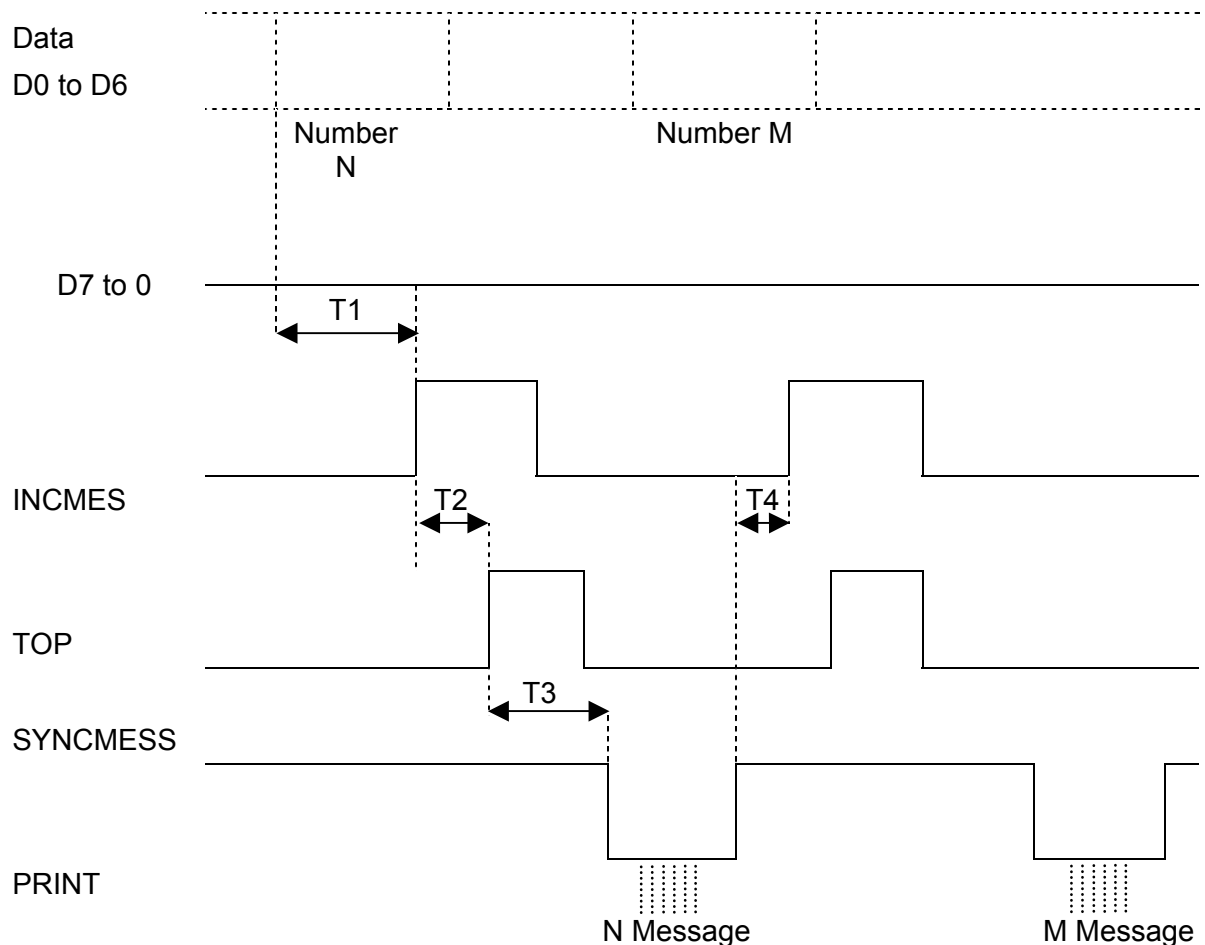
SYNCMES:

This output is active during printing of the selected message.



Message selection by Parallel interface

Signals Diagram



- T1: Minimum presence time between the D0 to D7 data and the rise of the INCMES input = 30 μ s.
- T2: Minimum time between the rise of INCMES and that of the TOP input = 10 ms (this time depends on the message content).
- T3: Time between the rise of the TOP signal and printing of the first trame of the selected mesage. This time depends on the type of head and on the speed or delay in printing the selected message.
- T4: Minimum time between the end of printing (return of the SYNCMESS signal to rest) and a new rise of the INCMES input = 10 ms.



Message selection by Parallel interface

Message Increment Mode

A first active message must be initialized either by the Operator Interface, or by the serial link, or by the preceding Parallel Interface Mode.

■ Inputs

INCMES:

Each pulse on this input develops the active message number to print depending on the BIB input level.

SEBSBIB:

The SENSBIB input at low level allows increments of the message number. The SENSBIB input at high level allows decrementation of the message number.

After a change of status of the SENSBIB input, the first impulse of the INCMES signal does not modify the number of the selected message on printing.

TOP1:

A pulse on this input starts up printing of the selected message.

■ Output

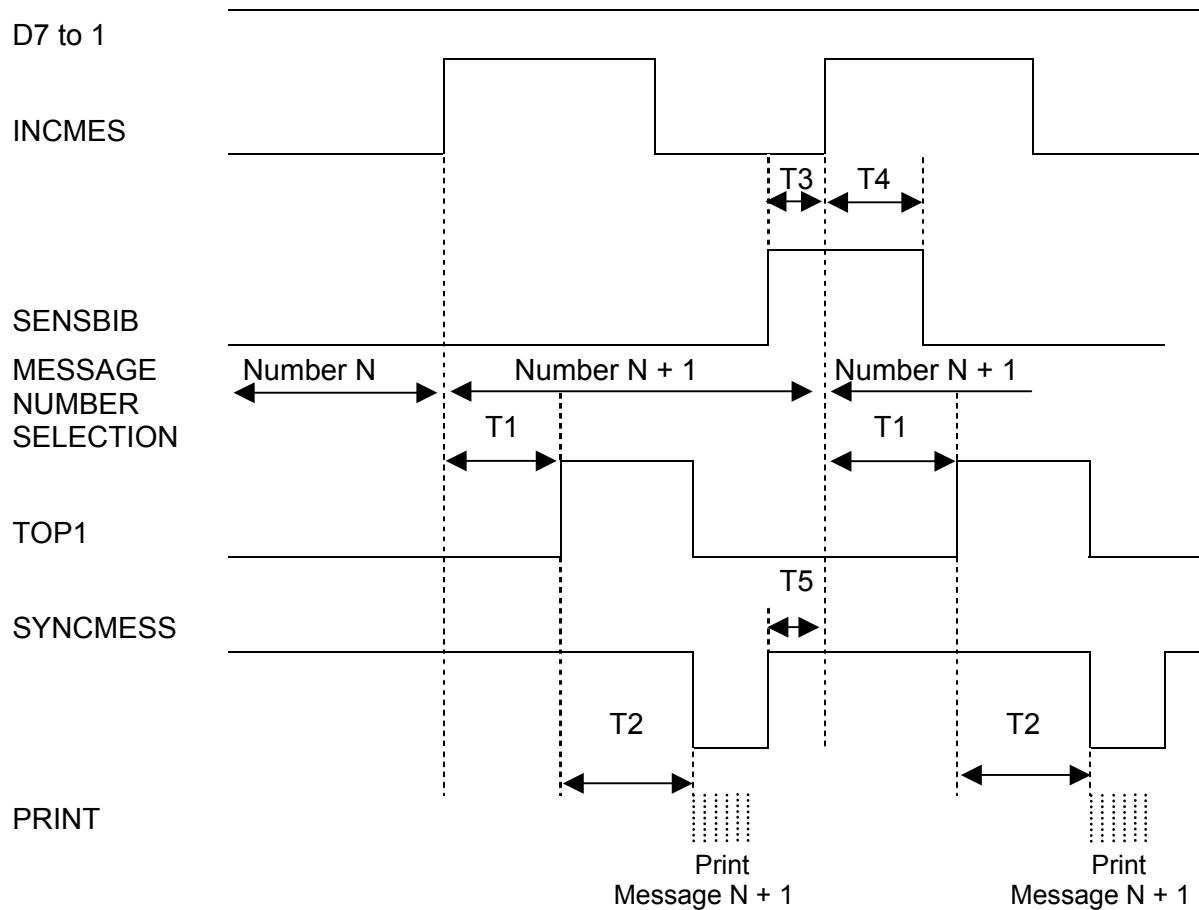
SYNCMES:

This output is active during printing of the selected message.



Message selection by Parallel interface

■ Signals Diagram



- T1: Minimum time between the rise of INCMES and that of the TOP input = 10 ms (this time depends on the message content).
- T2: Time between the rise of the TOP1 signal and printing of the first frame of the selected message (SYNCMESS signal active). This time depends on the type of head and on the speed or delay in printing the selected message.
- T3: Minimum presence time of the SENSBIB input before rise of the INCMES input = 30 μ s.
- T4: Minimum hold time of the SENSBIB signal after rise of the INCMES input = 1 ms.
- T5: Minimum time between the end of printing (return of the SYNCMESS signal to rest) and a new rise on the INCMES input = 10 ms.



Revision

■ Manual for serials and parallel interfaces Imaje 9020 – Imaje 9030

Update of the manual

- Revision index A corresponds to the initial edition of this manual.
- The revision index changes with every modifications.

Date of edition	Document revision index	Software index
04/2005	A	A12

The present document is drafted in French. Only the French text shall be deemed authentic and prevails in case of litigation.
Imaje S.A. reserves the right to alter the specifications and the design of this product at any time without giving prior notice.
Reproduction, in whole or in part, is not allowed.
Edition Anglaise





A35265-A



A **DOVER** COMPANY

Image S.A.
Head office:
9, rue Gaspard Monge
BP 110
26501 Bourg-lès-Valence Cedex
France
Tel.: (33) 4 75 75 55 00
Fax: (33) 4 75 82 98 10
[http:// www.image.com](http://www.image.com)