SIEMENS

SIMATIC

Data Highway

Reference Manual

This manual is part of the documentation “Loadable Drivers Data Highway” with the order number:

6ES7870-1EA00-0YB0

03/99
C79000-G7076-C309
Release 01

triangle and are marked as follows according to the level of danger:

Warning

indicates that death, severe personal injury or substantial property damage can result if proper precautions are not taken.

Note

draws your attention to particularly important information on the product, handling the product, or to a particular part of the documentation.

Qualified Personnel

The device/system may only be set up and operated in conjunction with this manual.

Only **qualified personnel** should be allowed to install and work on this equipment. Qualified persons are defined as persons who are authorized to commission, to ground, and to tag circuits, equipment, and systems in accordance with established safety practices and standards.

Correct Usage

Note the following:

Warning

This device and its components may only be used for the applications described in the catalog or the technical description, and only in connection with devices or components from other manufacturers which have been approved or recommended by Siemens.

This product can only function correctly and safely if it is transported, stored, set up, and installed correctly, and operated and maintained as recommended.

Warning

FM WARNING - DO NOT DISCONNECT WHILE CIRCUIT IS LIVE UNLESS LOCATION IS KNOWN TO BE NON-HAZARDOUS

Trademarks

SIMATIC®, SIMATIC NET®, and SIMATIC HMI® are registered trademarks of SIEMENS AG.

Third parties using for their own purposes any other names in this document which refer to trademarks might infringe upon the rights of the trademark owners.

Siemens AG
Bereich Automatisierungs- und Antriebstechnik
Geschaeftsgebiet Industrie-Automatisierungssysteme
Postfach 4848, D-90327 Nuernberg

Preface

Purpose of the Manual

The information provided in this manual enables you to establish a connection between a communications processor (CP) and an Data Highway DF1 communication module and place it in operation.

Contents of the Manual

This manual describes the function of the loadable driver and how it is integrated in the hardware and software of the communication processors CP 341 and CP 441-2.

It contains information about the following topics:

- Product description and assembly
- Driver startup
- Driver installation and parameter assignment
- Interface between the CPU and the communications processor (CP)
- Transmission protocol
- Driver diagnostics

Where is this Manual Valid?

This manual is valid for the following software:

Product	Order Number	from Version
Loadable driver for point-to-point CPs	6ES7 870-1AE00-0YA0	1.0

This manual contains the description of the driver that is valid at the time of publishing of the manual.

Area of Application

The driver described in this manual provides a loadable protocol for the communications processor which can be used in place of the following standard protocols: 3964R, RK512, ASCII, and printer.

Conventions

This documentation uses the designations CP (communications processor) or CP 341 (or CP 441-2).

Structure of this Manual

To facilitate rapid access to special information, the manual contains the following aids:

- At the start of the manual, you will find a complete list of contents.
- The appendix is followed by a glossary which defines the important technical terms used in the manual.
- At the end of the manual there is a detailed index to enable you to find the desired information quickly.

Additional Information

Any additional information required on this driver (such as for installation and startup) can be found in the manual for the communications processor being used. Further information on STEP 7 can be found in the following manuals:

SIMATIC Software
Standard Software for S7 and M7
Programming with STEP 7

SIMATIC Software
System Software for S7-300/400
System and Standard Functions
Reference Manual

Additional Assistance

If you have any questions regarding the use of the driver or function block described in this manual and cannot find an answer in this documentation, please contact the Siemens representative from whom you obtained the driver.

Note

Some driver procedures used for communication between communications processor and CPU may have been modified or extended. In particular, such modifications and extensions may affect the event classes and event numbers used for diagnosis.

Please also note that this manual only describes the modifications and extensions made to standard functions. Basic information about these standard functions can be found in the manual for the communications processor being used.

In addition, an exact knowledge of the function of the communications processor being used is required to ensure proper operation of this driver.

Contents

Product Description

1

1.1 Areas of Application

How the Product is Integrated in the System Environment

This driver is a software product for use with CP 341 (S7-300) and CP 441-2 (S7-400) communication processors.

The CP 341 and the CP 441-2 can be used within the S7 programmable logic controller (PLC) to establish serial communication links to partner systems.

How the Driver Functions

This driver enables a communication link to be established between an Data Highway DF1 communication submodule and a CP 341/CP 441-2 communication module.

Data is transmitted using the **Asynchronous Link Full-Duplex (DF1)** protocol. The communication submodule can be any module whose "Asynchronous Link" interface allows parameters to be assigned to the DF1 protocol. A connection to a second CPU interface from Allen-Bradley CPUs can also be established if such settings can be made in the DF1 protocol.

Suitable Interface Submodules

Both serial interfaces for the CP 441-2 can be independently operated using different standard protocols or loadable protocols.

The communications processor interface connection can be made using RS 232, TTY, or RS 422.

This driver will not support the use of the RS 232 auxiliary signal for the RS 232 interface.

Only a four-wire connection can be made with the X27/RS 422 interface. RS485 operation is not supported.

Possible System Configuration

The following schematic depicts a possible system configuration.

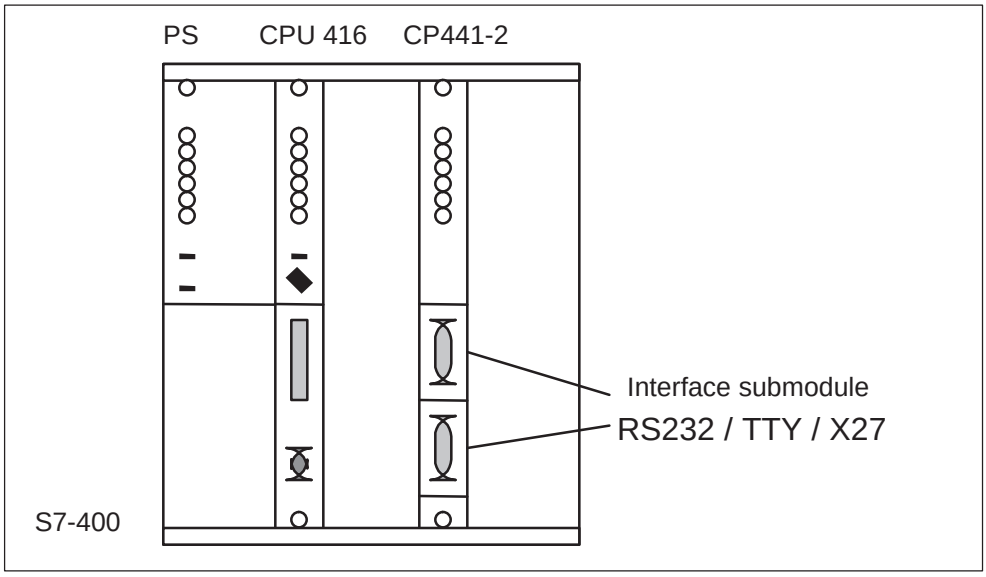


Fig. 1-1 System Configuration

1.2 Hardware and Software Requirements

Suitable Modules

This driver can be run on a CP 341 and on a CP 441-2 with order number 6ES7 441-2AA0x-0AE0 (x >1).

Dongle

Operating the communications processor with loadable drivers requires a dongle labeled C79459-A1715-B22. This dongle is included in the driver consignment.

Drivers can be loaded for both interfaces of the CP 441-2. When loading drivers, please note that you can also use the drivers using the dongle labeled C79459-A1715-B21 with the Data Highway dongle. However, please also note that this B21 dongle will not permit operation of Data Highway.

Inserting the Dongle

To insert the dongle, first remove the communications processor from the module rack. You can then insert the dongle in the submodule slot located on the back side of the communications processor above the plug for the backplane bus.

CPU Memory Card

When the CP 441-2 is in operation, the loadable drivers are assigned parameters, loaded into the CPU memory card, and then transferred to the communications processor memory on CPU startup.

Each communications processor interface used to assign parameters to loadable drivers requires approximately 25 Kbytes of the CPU memory card.

For this reason, a CPU memory card of sufficient capacity must be installed. If necessary, a RAM or FLASH **Memory Card** (MLFB 6ES7 952-…) will be needed to help meet these capacity requirements.

When the CP 341 is in operation, the loadable drivers are loaded directly into the CP 341. This means that the S7-300 CPU does not require a memory card. However, please note that this also means that modules cannot be exchanged without using a programming device (PG).

Software Release / Version

Drivers can be loaded for the CP 441-2 starting with STEP 7 version 4.0.

The PtP parameter interface *CP 441-2: CP: Point–to–Point Communication, Parameter Assignment* (MLFB 6ES7 441-2AA02-7xG0) must be at least version 4.

Drivers can be loaded for the CP 341 starting with STEP 7 version 3.2.

Installing the Drivers

2

General Information

The following information pertaining to STEP 7 specifically refers to STEP 7 versions 3.2 and 4.02.

Later versions may have changes in system functions as well as in the names of system functions and data folders.

2.1 Installing Drivers on the STEP 7 PG or PC

Diskettes Supplied

The driver, the related parameter assignment dialog boxes, the sample program, the SETUP file for installing the driver software on your programming device (PG), and this manual are all supplied on 3.5” high density MS-DOS diskettes. The driver consists of driver code and driver-specific dialog box files.

System Requirements

To be able to install the driver, you first must have a **STEP 7 package** as well as the **parameter assignment interface** *CP: Point–To–Point Communication, Parameter Assignment* installed on your system.

Installing the Driver

To install the driver, proceed as follows:

1. Insert the first diskette in the disk drive of the programming device or PC.
2. In Windows 95, start the dialog for installing the software by double-clicking on the “Add / Remove Programs” icon in the “Control Panel” window.
3. In the dialog box that appears, select the file **Setup.exe** on the appropriate disk drive; then start the installation.
4. Follow the step-by-step instructions displayed by the installation program.

   Result: The driver and the parameter assignment dialog boxes are installed in the following folder: Step7\S7fptp\S7Driver.

Among others, this folder contains the following files:

- S7wfpe1a.dll
- S7wfpe1x.cod
- S7wfpe2x.cod

Uninstalling the Driver

To uninstall the driver from the STEP 7 package in Windows 95, select the sequence “Control Panel,” “Add / Remove Programs,” and “Remove”. After this, check the folder Step7\S7fptp\S7Driver to confirm that all the files (S7wfpe1?.*, S7wfpe2?.*, S7wfpe3?.*) are deleted.

2.2 Assigning Parameters to Loadable Drivers

System Requirements

Before assigning parameters to a loadable driver, you must first have completely and correctly assigned parameters “Configuring Hardware“ to the communications processor being used. Further information about this can be found in the communications processor manual and the STEP 7 documentation.

Opening the Communications Processor–PtP Parameter Assignment Interface

To start “Configuring Hardware,” select the appropriate SIMATIC station and double-click “Hardware” (or “Edit –> Open object”).

Select the communications processor and then select **Edit –> Object Properties**.

After selecting the appropriate interface (in this case, only CP 441-2) and interface submodule (again, only CP 441-2), click the “**Protocol Parameters**” button to enter the dialog for selecting a protocol.

Selecting the Protocol

In addition to standard protocols, the selection box displayed also lists all loadable drivers installed. Select “**Data Highway**” to access these drivers.

To start the dialog for assigning protocol-specific parameters, double-click the icon (letter box) for the transmission protocol.

Driver-Specific Parameters

The parameters described in the following section can be assigned to this driver in the individual dialog boxes.

2.2.1 Parameters for the Data Highway DF1 Protocol

Speed, Character Frame		
Parameter	**Range of Values**	**Default Value**
Transmission rate	300 600 1,200 2,400 4,800 9,600 19,200 38,400 76,800	9,600
Data bits	8 7	8
Stop bits	1 2	1
Parity	None Odd Even	Even

Transmission Rate, Overall Transmission Rate

The overall, combined transmission rates for both interfaces of the **CP 441-2** cannot exceed 76,800 bps.

The transmission rate for the TTY interface must not exceed 19,200 bps.

You must set the **same** transmission rate on the CP and on the partner.

Data Bits

The number of data bits indicates how many bits are used to depict a transmitted character.

You must set the **same** number of data bits on CP and on the partner.

Stop Bits

The number of stop bits defines the smallest interval between two transmitted characters.

You must set the **same** number of stop bits on CP and on the partner.

Parity

The parity bit is added for data security. Depending on the parameters assigned, it extends the number of transmitted bits to an even or odd number.

A parity setting of “none” means that no parity bit is transmitted. This setting will reduce data transmission security.

You must set the **same** parity on CP and on the partner.

Overview of Protocol Parameters

Parameters	Range of Values	Default Value
Response monitoring time	30 to 10,000 (unit = 1 ms)	3,000 = 3 seconds.
No. of attempts on NAK	0 to 5	3
No. of ENQ requests	0 to 5	3
Message frame monitoring		All message frames received are transferred to the CPU.
Type of CP acknowledgement on receipt		If message frame has been received without error, CP sends immediate acknowledgement.

Response Monitoring Time for Acknowledgment Signal

This is the length of monitoring time for awaiting receipt of an acknowledgement from the connection partner after a message frame has been sent. Within this time period, a response of ACK or NAK is expected.
This time period can be set between 30 ms and 10 seconds in units of 1 ms.

This monitoring time setting must be **identical** on both the communications processor side and the communication partner side.

Number of Attempts on NAK

After the communications processor has sent a message frame and received an acknowledgement of NAK from the communication partner, the communications processor repeats the message frame for the ‘n’ amount of times set. After this, the SEND job is canceled with the message “Job completed with errors.”

If a parameter value of “0” is assigned, the communications processor will not repeat the message frame.

No. of ENQ Requests

If the communications processor has sent a message frame and received neither an ACK nor a NAK acknowledgement from the communication partner or if the monitoring time specified has expired, the communications processor asks the communication partner to repeat the last acknowledgment by sending it an ENQ request.

The communications processor repeats this ENQ request for the ‘n’ amount of times set. After this, the SEND job is canceled with the message “Job completed with errors.”

If a parameter value of “0” is assigned, the communications processor will not send an ENQ request.

Message Frame Monitoring

Available options:

- All message frames received error-free by the communications processor are transferred to the SIMATIC CPU.
- Suppressing duplicate frames:
  If the 2nd, 3rd, 5th, and 6th bytes of the message frame received are the same as the corresponding bytes of the message frame previously received, the message frame is not transferred to the CPU. Nevertheless, the communications processor still acknowledges receipt to the communication partner with ACK.

Type of Communications Processor Acknowledgement on Receipt

Available options:

- Acknowledgement sent immediately:
  A message frame received error-free by the communications processor is immediately acknowledged with ACK. After this, the data are transferred to the SIMATIC CPU.
  This setting allows a high level of data throughput to be achieved.
- Acknowledgement sent **after message is transferred to the CPU**:
  A message frame received error-free by the communications processor is first transferred to the SIMATIC CPU. If this data transfer occurs without error, the communications processor acknowledges the message with ACK. However, if error occurs during transfer, the message is acknowledged with NAK.
  This setting allows a high level of data security to be achieved.

  Please note the following:
  In this operating mode, the length of time until the acknowedgement is sent by the communications processor is governed by the CPU cycle time as well as the structure of the application program.

2.2.2 Parameters for the RS422 (X27) Interface

"Full–Duplex (RS422), Four–Wire Operation"

In this operating mode, message frames are sent on the sending lineT(A),T(B) and received on the receiving line R(A),R(B).

Overview

If you are using a communications processor with an RS 422 interface, a default setting for the receiving line can still be made.

X27 (RS 422/485) Interface			
Parameter	**Description**	**Range of Values**	**Default Value**
Default setting for the receiving line	No default Default "break detection" Default "high-level"	None R(A)5V,R(B)0V R(A)0V,R(B)5V	R(A)5V, R(B)0V

Default Setting for the Receiving Line

"None" (Float)

The receiving line R(A), R(B) has **NO** default setting. This means that no break in the transmission line can be detected.

Default setting "R(A) 5V, R(B) 0V" ("BREAK")

This default setting detects a "break signal" when the front panel plug is not connected or when the receiving line is interrupted.

Default setting "R(A) 0V, R(B) 5V" (High)

This default setting does **not** detect a "break signal" when the front panel plug is not connected or when the receiving line is interrupted.

Assigning Parameters

Assign the parameter settings necessary for your connection. When doing so, close each dialog box with "OK."

2.3 Loading the Configuration and Parameter Data for the CP 341

Data Management

After the dialog **“Configuring Hardware”** is closed with OK, the data are automatically stored in your STEP 7 project.

Loading Configuration and Parameter Data

You can now upload the configuration and parameter data online from the programming device (PG) to the CPU. Select the menu command **Destination System –> Load** to transfer the data to the CPU.

The module parameters for the communications processor are automatically transferred from the CPU to the communications processor during CPU startup and each time the system is switched from the STOP to the RUN mode as soon as the communications processor can be accessed through the S7 300 backplane bus.

The driver code is not stored in the CPU; instead, it directly stored along with the parameter interface in the retentive memory of the CP 341. For this reason, please note that modules cannot be exchanged without using a programming device.

2.4 Loading the Configuration and Parameter Data for the CP 441-2

Data Management

After the dialog “**Configuring Hardware**” or “**Configuring Connections**” is closed with OK, the data (including module parameters and driver code) are automatically stored in your STEP 7 project.

Loading Configuration and Parameter Data

You can now load the configuration and parameter data online from the programming device to the CPU. Select the menu command **Destination System –> Load** to transfer the data to the CPU.

The module parameters for the communications processor and driver code are loaded during CPU startup and are automatically transferred from the CPU to the communications processor as soon as the communications processor can be accessed through the S7 400 backplane bus.

2.5 Parameters for “CPU Startup”

The following information only applies to the CP 441–2. If you are using a CP 341, you can omit this section.

Configuring Hardware

To prevent problems during CPU–CP startup. the following settings should be made when using “**Configuring Hardware**” to **assign parameters** to the **CPU**.

To start assigning parameters, either double-click the CPU or select the CPU and then select the menu command **Edit –> Object Properties** to enter the dialog box “CPU Properties.”

In the tab “**Startup**,” the value for “**Monitoring time for**” for “**Transferring Parameters to module (100 ms):**” should be set to at least *1000* (= 100 s).

Reason: When parameters are assigned to a CP 441–2 interface with a loadable driver, the driver code is also transferred to the communications processor. In addition, the entire loading procedure is monitored during the time indicated above and so this value must be set at an appropriately large amount.

CPU-Communication Partner Interface 3

3.1 CPU-Communication Partner Interface for the CP 341

Communication Function Blocks (FBs)

The data exchange between the CP 341 and the CPU is carried out by the function blocks **P_SND_RK (FB 8 for sending)** and **P_RCV_RK (FB 7 for receiving)**. These blocks are standard components in the library of the CP 341.

Number of Transfered Data Bytes

The **length** of the data to be transferred can be at most 1,024 bytes.

The special driver expects a minimum data length of 4 bytes.

3.2 CPU–Communication Partner Interface for the CP 441–2

Communication Function Blocks (FBs)

The data exchange between the CP 441 and the CPU is carried out by the integrated system function blocks **BSEND** (SFB12) and **BRCV** (SFB 13).

Communication Link

The connection ID of the connection configuration must be indicated at the communication FB.

The **parameter ID** describes the unique communication link to a communication partner.

Number of Transferred Data Bytes

The **length** of the data to be transferred can be at most 4,096 bytes.

The special driver expects a minimum data length of 4 bytes.

Transmission Protocol

4

General Information

The descriptions in this chapter employ the same terminology used in the Data Highway DF 1 protocol.

This transmission protocol is full-duplex, which means that data can be sent and received simultaneously.

The full-duplex driver used here consists of a TRANSMITTER part and a RECEIVER part.

The TRANSMITTER sends messages (message frames) and waits for response messages (response message frames) to them. If the response is not correctly recognized, the TRANSMITTER can send a request to repeat the response.

The RECEIVER waits for messages and sends response messages. If it receives a request to repeat a response, the RECEIVER will resend the last response.

4.1 Message Frame Format

The full-duplex protocol consists of “Control Symbols” and “Data Symbols.”

Symbol			Type	Meaning
DLE	STX		control	Start of a message
DLE	ETX	BCC	control	End of a message
DLE	ACK		control	Positive response (OK)
DLE	NAK		control	Negative response (not OK)
DLE	ENQ		control	Request to repeat the last response.
APP DATA			data	Application data block:
				Character 00H – 0FH, 11H – 0FFH
DLE	DLE		data	DLE duplication within the APP DATA Block

Symbol	Hex Value
STX	02H
ETX	03H
ENQ	05H
ACK	06H
DLE	10H
NAK	15H

Protocol Messages

A data message begins with the start symbol DLE STX.

The data message connects to the application block APP DATA.

The message ends with the end sequence DLE ETX BCC.

		Application Data Block			
DLE	STX	APP DATA	DLE	ETX	BCC

BCC Character Generation

The BCC character is created as a two’s complement of the eight-bit sum of all characters in the APP DATA block.

For DLE duplication in APP DATA, only **one** DLE is included in the BCC character.

Example:

DLE	STX	08_H	09_H	06_H	00_H	10_H	10_H	04_H	03_H	DLE	ETX	$D2_H$

08H + 09H + 06H +00H + 10H + 04H + 03H = 2EH
Two's complement of 2EH = D2H

Without Embedded Responses

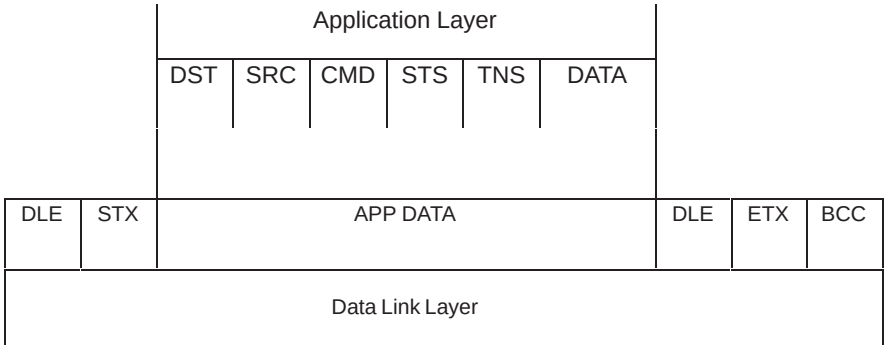
Data Highway DF 1 modules can process "embedded responses." This means that response symbols can be transmitted within a message. These modules can be assigned a parameter (Yes/No) that determines whether they will operate with or without "embedded responses."

This function is **NOT** available in the special driver, which means that no embedded responses are possible with it.

Data Exchange: SIMATIC S7 <–> Data Highway DF 1

The special driver implements the "Data Link Layer" for a data transmission.

The "Application Layer Data APP DATA" are created by the SIMATIC user program (when sending messages) or evaluated by it (when receiving messages from Allen–Bradley).

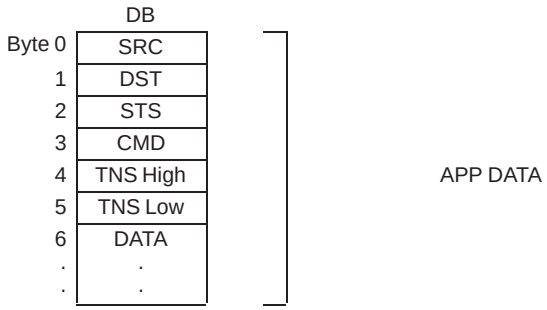


4.2 Sending Message Frames

Starting SEND Jobs

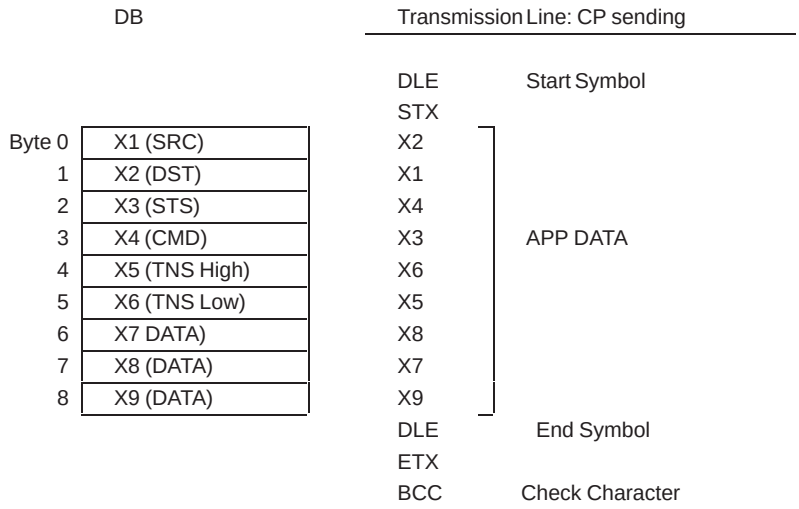
A message sent is started by calling a BSEND or a P_SND_RK.

The APP DATA to be sent is arranged within data blocks.



The maximum length of the APP DATA block is 2,048 words (4,096 bytes) for the CP 441 or 512 words (1,024 bytes) for the CP 341.

When sending the data on a line, the low byte in each DW n+x is transmitted first, followed by the high byte. This sequence is required because Data Highway communication modules transmit data on a word basis in which first the low byte and then the high byte are transmitted.



Data Transmission

Data transmission begins with the start symbol DLE STX. After this, the data are sent from the send DB (APP DATA). The message ends with DLE ETX BCC.

After the transmission is completed, a response is awaited for the value of the “Response Monitoring Time” parameter (default value = 3 seconds) assigned.

If the response received is DLE ACK, the SEND job is completed with “Job completed without error.”

——— DLE STX APP DATA DLE ETX BCC ———>
<——————— DLE ACK ———————

If the response received is DLE NAK, a check is made to determine whether the parameter value assigned to “No. of Attempts on NAK” will allow the message to be re-sent. If YES, then the message is re-sent (DLE STX APP DATA DLE ETX BCC) and a response is awaited for the “Response Monitoring Time” assigned.

After the limit to the “No. of Attempts on NAK” assigned is reached, and each repeated message has received a response of DLE NAK from Allen–Bradley, the SEND job is completed with “Job completed with errors.”

——— DLE STX APP DAT?? DLE ETX BCC ———>
<——————— DLE NAK ———————

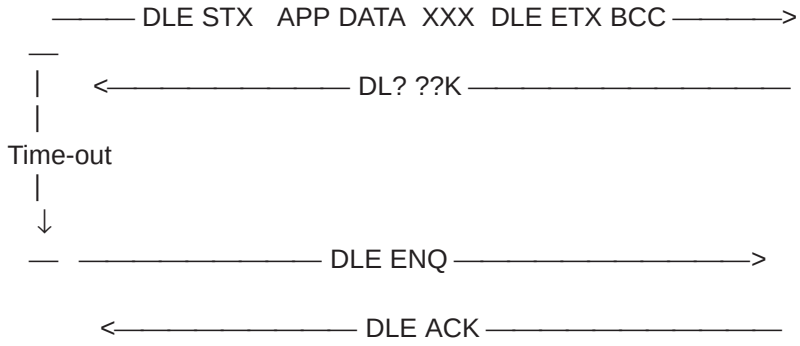
——— DLE STX APP DAT?? DLE ETX BCC ———>
<——————— DLE NAK ———————

——— DLE STX APP DAT?? DLE ETX BCC ———>
<——————— DLE NAK ———————

?? = unrecognized or bad character

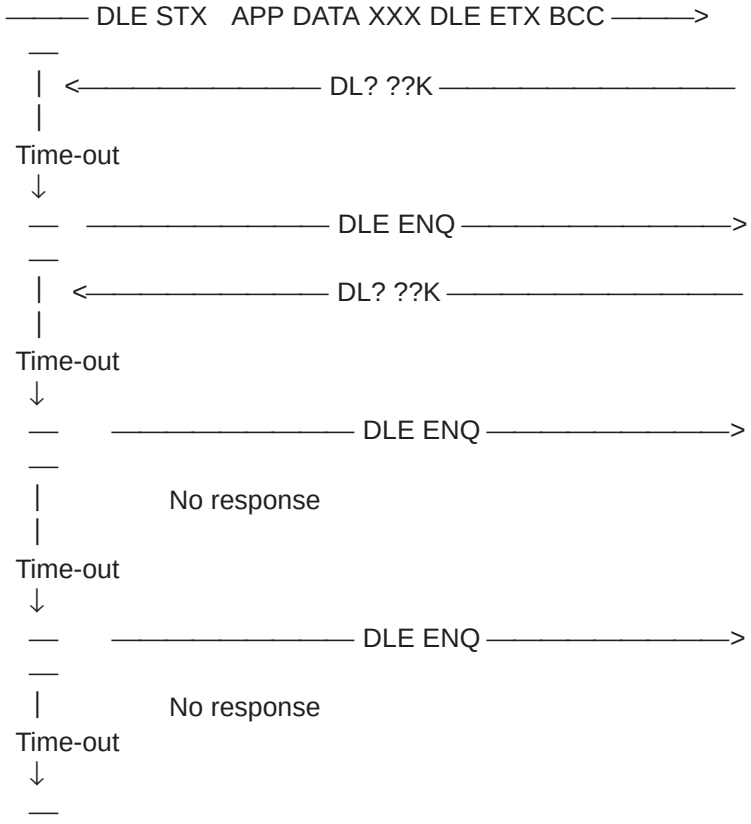
If a character sequence not equal to DLE ACK or DLE NAK is received or if bad characters are received, these are not regarded as a valid response and the communications processor will continue to await a response from the communication partner until the “Response Monitoring Time” assigned has expired.

Sequence of the “Response Monitoring Time”



If the “Response Monitoring Time” assigned expires, the communications processor sends DLE ENQ to request that the communication partner resend the Last_Response.
After the DLE ENQ is sent, the “Response Monitoring Time” period is restarted.

The maximum “No. of ENQ Requests” sent is determined by the parameter value assigned. If no proper response (DLE ACK or DLE NAK) is detected after the last attempt or if the request parameter value is “0,” the SEND job is also completed with “Job completed with error.”



The number of repeated attempts after NAK and the number of ENQ requests are independent of each other. Their behavior depends on their own parameter setting.

4.3 Receiving Message Frames

A message received from a communication partner is started by calling a BRCV or a P_RCV_RK.

If the RECEIVER part of the driver is in its initial state, it waits for the start of a message with DLE STX or for a DLE ENQ request from the communication partner.

If symbols other than DLE STX or DLE ENQ are detected, the “Last_Response” is set to NAK and an error is logged or reported.

Last_Response

After receiving a message from the communication partner, the RECEIVER part of the special driver acknowledges it with a positive response of DLE ACK or with a negative response of DLE NAK.

The last ACK / NAK response sent is stored in the special driver memory as the “Last_Response.”

“Last_Response” = ACK for a last response of DLE ACK,
“Last_Response” = NAK for a last response of DLE NAK.
“Last_Response” has the default value of NAK after a reset.

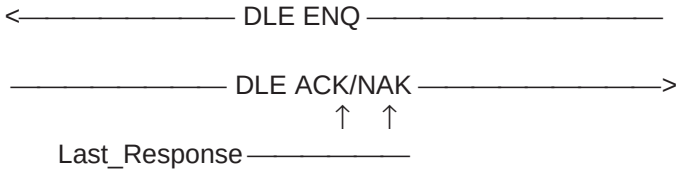
If any errors occur during operation, the “Last_Response” is always reset to NAK.

“Last_Response” Request from Communication Partner

If the RECEIVER part of the special driver receives the symbol DLE ENQ, the communications processor sends the “Last_Response” to the communication partner.

After this is completed, the RECEIVER returns to its initial state.

The “Last Response” is sent each time it is requested by the communication partner.



Receiving a Message

If the start of a message (DLE STX) from the communication partner is detected, the internal BCC character is reset.
After this, the APP DATA are received.
After the end symbol DLE ETX has been received, the system waits for the BCC character.

If any of the following events occur while the APP DATA Block is being received, then the RECEIVER part of the driver sends the negative response **DLE NAK** :

- Transmission error detected
- Character / signal delay time (200 ms) expired
- BCC character incorrect
- Message could not be transferred to the CPU

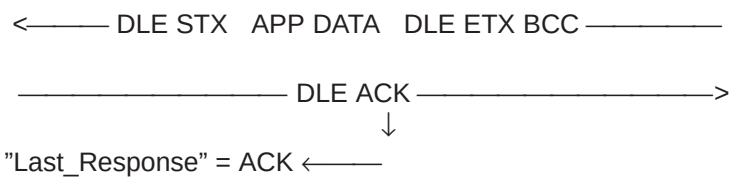
If the message was received without error or was transferred to the CPU without error, the RECEIVER part of the driver sends the positive response **DLE ACK**.

If the message received is a duplicate message and has been assigned parameters for “duplicate message detection,” the message is not transferred to the CPU. In this case, the message is immediately acknowledged with the response **DLE ACK**.

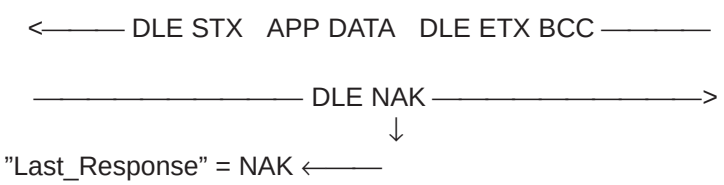
After the response has been sent, the driver returns to its initial state.

The response is stored in “Last_Response.”

Receipt of a message and a positive response:



Receipt of a message and a negative response:

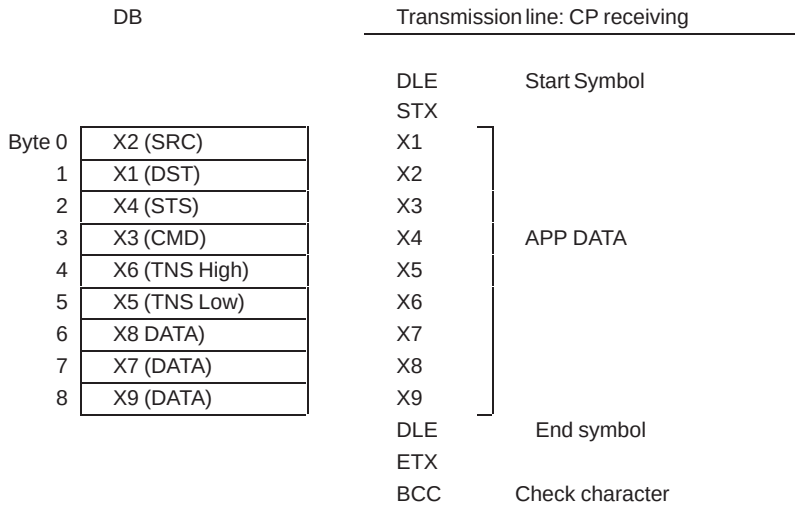


Entering the Messages into the Receive Data Block

If a message is received without error, it is entered into the data block of the BRCV.

Data Highway DF 1 modules transmit data in words, starting with the low byte, followed by the high byte. However, in S7 data are not processed from low to high byte but instead from high to low byte.

For this reason, the incoming high byte and low byte sequence is switched word-by-word by the special driver during data entry into the receive datablock.



Duplicate Message Transmission Detection (Message Frame Monitoring)

The PtP-Param software can be used to assign parameters that determine whether the special driver operates with or without message frame monitoring.

If parameters have been assigned to “Message Frame Monitoring” and a message has been correctly detected, the special driver compares the 2nd (SRC), 3rd (CMD), and 5th / 6th (TNS) bytes of the message (APP DATA) currently being received with those of the last message received. If these respective bytes are identical, the current incoming message is a duplicate message.

In this case, the message is not transferred to the CPU; instead, it is immediately acknowledged with the positive response DLE ACK.

“Embedded Responses”

Data Highway DF 1 modules can be assigned parameters that determine whether they operate with or without “embedded responses.”

The function “embedded responses” means that the control symbols DLE ACK, DLE NAK, DLE ENQ can be contained in the APP DATA block.

The “**embedded responses**” function is **not** available with this special driver. This means that the modules must be assigned parameters not allowing “embedded responses.”

Communications Processor Acknowledgement Characteristics

The PtP-Param software can be used to assign parameters that determine the acknowledgement behavior of the special driver when receiving a message.

Immediate Acknowledgement

Assigning the parameter “Acknowledge immediately” to the special driver causes it to send the positive response DLE ACK immediately after receiving a message, after which the the driver is again ready to receive messages.

After this, the data are entered into the CPU. This setting allows a high level of data throughput to be achieved.

The response is delayed if the data in the previously received message could not be completely transferred to the CPU. In this case, the maximum length of the time delay depends on the parameter value assigned to “Response Monitoring Time.” If the data still could not be transferred to the CPU after this set length of time, the special driver acknowledges the message currently being received with NAK (message sink full).

Acknowledgement after Transfer to the CPU

Assigning the parameter “Acknowledge after Transfer to the CPU” to the special driver causes it to send the positive response DLE ACK only after the message received has been completely transferred to the CPU.
With this setting, the positive response DLE ACK means that the data transfer to the next highest level (the CPU) was accomplished error free.

The data transfer to the CPU requires a certain amount of time, the length of which depends on the cycle time of the CPU in use.

The special driver limits the time-delay of the response to the parameter value assigned to the “Response Monitoring Time.” If the data still could not be transferred to the CPU after this set length of time, the special driver acknowledges the message currently being received with NAK.

Diagnostics

5

Diagnostic Functions

The diagnostic functions in the communications processor allow you to localize system errors quickly. The following diagnostic options are available:

- Diagnosis using the communications processor display
- Diagnosis using the STATUS output of function blocks
- Diagnosis using the error message area SYSTAT (only applies to CP 441-2)
- Diagnostic buffer in the communications processor

This chapter contains descriptions of the driver-specific error messages.

5.1 Table of Errors and Events

All error messages with an event class not equal to 14 (0FH) are already described in the manuals for the CP 341 or the CP 441 and are therefore not listed here.

The error messages listed below can occur in the diagnostic buffer as well as in the FB status (CP 341) or in the Systat (CP 441).

5.1.1 Special Driver Error Messages

Event Class 14 (0FH) "Loadable Driver – General Processing Errors"			
Event Class or Number (Hex)	**Event Number (Decimal)**	**Event Text**	**Remedy**
0F 01H	1	**Error during initialization** of driver-specific SCC process.	Reassign driver parameters and load the driver again.
0F 02H	2	**Error during driver start-up:** Incorrect SCC process active (SCC driver). The driver cannot be used with this SCC driver.	Reassign driver parameters and load the driver again.
0F 03H	3	**Error during driver start-up:** Incorrect data transmission process active (interface to SFBs). The driver cannot be used with this data transmission process.	Reassign driver parameters and load the driver again.
0F 04H	4	**Error during driver start-up:** Invalid interface submodule. The driver cannot be operated with the interface submodule assigned as a parameter.	Check parameter assignment and correct.
0F 05H	5	**No dongle inserted** or the dongle inserted is defective. The driver is not operational.	Use the correct dongle. See Section 1.2
0F 06H	6	**The dongle contents are invalid.** The driver is not operational.	Use the correct dongle. See Section 1.2
0F 10H	16	**Internal processing error: Sending device** in default mode.	Warm restart of CP (Power_ON).
0F 11H	17	**Internal processing error:** Receiving device in default mode.	Warm restart of CP (Power_ON).
0F 12H	18	**Internal error in active device.**	Warm restart of CP (Power_ON)

Event Class or Number (Hex)	Event Number (Decimal)	Event Text	Remedy
0F 13H	19	**Internal error in passive device**.	Warm restart of CP (Power_ON).
0F 20H	32	**Response Monitoring Time incorrectly set.**	Correct the driver parameters.
0F 21H	33	**Repeat number for NAKs incorrectly set.**	Correct the driver parameters.
0F 22H	34	**Repeat number for ENQs incorrectly set.**	Correct the driver parameters.
0F 2EH	46	**The driver is not operational.** Error when reading interface parameters.	Warm restart of CP (Power_ON).
0F 32H	50	**Error when transfering data with BRCV to the CPU:** Determine cause - detailed error text before this entry in SYSTAT. or STATUS.	Check CP-CPU interface.
0F 33H	51	**Invalid device state.** Internal error during CP-CPU data transfer:	Check CP-CPU interface.
0F 34H	52	**Minimum length of data to be transferred is less than 4 bytes.**	Check the first word in the Send data block.
0F 50H	80	**The set amount "Number of ENQ Requests" has been sent.** No correct response received from communication partner (The "Response Monitoring Time" has expired). The SEND job has been canceled.	Check the connection module or transmission link.
0F 51H	81	**The set amount of "Repeat Number for NAK" has been sent.** Communication partner responding with NAK. The SEND job has been canceled.	Check the connection module or transmission link.
0F 52H	82	**The SEND job has been canceled.**The message frame could not be sent within 30 seconds after being triggered because incoming message frames from communication partner are continuously arriving.	Check the connection module.
0F 53H	83	**The "Response Monitoring Time" has expired.** The communication partner has either sent no response during the monitoring time or the response was not received without errors by the CP.	Check the connection module or transmission link.

Event Class or Number (Hex)	Event Number (Decimal)	Event Text	Remedy
0F 54H	84	**Transmission error in a character during quiet mode** (waiting for DLE).	Check the transmission link. (Check parameters for transmission rate and character frame.)
0F 55H	85	**The character after DLE was not recognized.**After receiving the DLE character during quiet mode, the signal delay time (200 ms) has expired.	Check the connection module or transmission link with a line monitor.
0F 56H	86	**Transmission error detected while receiving character after DLE.**	Check the connection module or transmission link with a line monitor.
0F 57H	87	**The combination DLE DLE was detected** at the start of receiving data. The combination DLE DLE can only occur after the combination DLE STX. Synchronization error.	Check the connection module or transmission link with a line monitor.
0F 58H	88	**The combination DLE ETX was detected** at the start of receiving data. The combination DLE ETX can only occur after the combination DLE STX. Synchronization error.	Check the connection module or transmission link with a line monitor.
0F 59H	89	**CP has not yet sent an acknowledgment.** After transfer of a received message to the CPU a new message has been detected. Synchronization error À.	Check the connection module or transmission link.
0F 5AH	90	**During transfer of a received message to the CPU a new message has been detected.** New message being ignored Synchronization error À.	Check the connection module or transmission link.
0F 5BH	91	**While receiving a message from the communication partner BREAK occurred.** Message being ignored.	Check the connection module or transmission link.
0F 5CH	92	**An incorrect start combination DLE xxx was detected in the initial state of the receiver.** Message being ignored. Synchronization error.	Check the connection module or transmission link with a line monitor.
0F 5DH	93	**While receiving a message, a character combination not equal to DLE DLE or DLE ETX was detected.** This is invalid since no “embedded responses” are permitted. Message being ignored. Synchronization error.	Check the connection module or transmission link with a line monitor.

Event Class or Number (Hex)	Event Number (Decimal)	Event Text	Remedy
0F 5EH	94	**The signal delay time (200 ms, fixed setting) expired while receiving a message.** Message being ignored.	Check the connection module or transmission link with a line monitor.
0F 5FH	95	**The check sum BCC received does not correspond** to the one created internally by the CPU. Message being ignored.	Check the connection module or transmission link with a line monitor.
0F 60H	96	**Overflow of receive buffer for this procedure** (4,096 or1,024 bytes)	Check the Frame lengh of the communication partner.
0F 61H	97	**Data from previously received message still being transferred to the CPU.** The currently received message being acknowledged with NAK.	Call RECEIVE block more frequently. Check the RD1 parameter.
0F 62H	98	**Data transfer from the currently received message to the CPU** could not be completed within the “Response Monitoring Time” parameter assigned. (maximum delay response symbols). Message acknowledged with NAK.	Call RECEIVE block more frequently.
0F 63H	99	**Frame sequence error occured.** While the data from currently received message is still being transferred to the CPU, the start of the next message (DLE STX) has already been detected. New message being ignored. No acknowledgment sent to communication partner. Synchronization error.	The communication partner must first wait for a response from the CP.
0F 64H	100	**Illegal ENQ request.** While the data from currently received message is still being transferred to the CPU, the start of an ENQ request (DLE ENQ) has already been detected. ENQ request being ignored. No response sent to communication partner. Synchronization error.	The communication partner must first wait for a response from the CP.
0F 65H	101	**Transmission error occurred while receiving a character** (overrun, frame error, parity error).	Check the communication module or the transmission link. (Check parameters for transmission rate and character frame.)

Technical Data

A

Transmission Times

The following tables contain the transmission times measured for the transmission protocol.

The following components were used to take the measurements:

- CPU 315-2 6ES7 315-1AF01-0AB0
- CP 341-RS232C 6ES7 341-1AH00-0AE0
- CPU416-1 6ES7 416-1XJ01-0AB0
- CPU441-2 6ES7 441-2AE00-0AE0
- RS232C module 6ES7 963-1AA00-0AA0

The following processing times were measured:

- From start of the job in the application program including the processing time in the communications processor.
- The transmission time for the job over the serial interface to the communication partner.
- The processing time of remote CP
- The transmission time of acknowledgment over the serial interface

The two CPs were linked; the CP 341 with the CP 341 and the CP 441 with the CP 441 up. The schmematic below shows the measurement configuration:

A frame with 8 bits, even parity and 1 stop bit was used.

The following tables show the times for three different transmission rates.

With these drivers, it is possible to decide whether the acknowledgment of a received frame is sent immediately on receipt of the frame or only after it has been transferred to the CPU. Measurements were therefore made for both situations.

CP441

(Acknowledgment from remote CP immediately on receiving frame)

Trans. Rate	User Data in Bytes	Processing Time CPU+CP (ms)	Trasmission Time of Frame (ms)	Processing Time of Remote CP (ms)	Trasmission Time of Ack. (ms)
76800	4	19	1,9	3,2	0,38
	10	19	2,7	5,7	0,38
	20	19	4,3	9,6	0,38
	50	19	8,5	21,7	0,38
	100	19	15,7	42	0,38
	200	19	30	84	0,38
	500	47	73	206	0,38
	1000	74	146	408	0,38
	2000	139	290	814	0,38
	4000	262	580	1600	0,38
9600	4	19	10,4	1,1	2,3
	10	19	17,2	1,1	2,3
	20	19	29	1,1	2,3
	50	19	64	1,1	2,3
	100	19	121	1,1	2,3
	200	19	236	1,1	2,3
	500	47	580	1,1	2,3
	1000	74	1151	1,1	2,3
	2000	139	2300	1,1	2,3
	4000	262	4600	1,1	2,3
1200	4	19	32	1,5	18,5
	10	19	137	1,5	18,5
	20	19	229	1,5	18,5
	50	19	504	1,5	18,5
	100	19	962	1,5	18,5
	200	19	1880	1,5	18,5
	500	47	4630	1,5	18,5
	1000	74	9210	1,5	18,5
	2000	139	18400	1,5	18,5
	4000	262	37000	1,5	18,5

(Acknowledgment from remote CP after transferring frame to the CPU)

Trans. Rate	User Data in Bytes	Processing Time CPU+CP (ms)	Trasmission Time of Frame (ms)	Processing Time of Remote CP (ms)	Trasmission Time of Ack. (ms)
76800	4	19	1,9	33	0,38
	10	19	2,7	36	0,38
	20	19	4,3	39	0,38
	50	19	8,5	50	0,38
	100	19	15,7	74	0,38
	200	19	30	113	0,38
	500	47	73	263	0,38
	1000	74	146	506	0,38
	2000	139	290	970	0,38
	4000	262	580	1800	0,38
9600	4	19	10,4	32	2,3
	10	19	17,2	32	2,3
	20	19	29	32	2,3
	50	19	64	32	2,3
	100	19	121	32	2,3
	200	19	236	34	2,3
	500	47	580	62	2,3
	1000	74	1151	93	2,3
	2000	139	2300	160	2,3
	4000	262	4600	280	2,3
1200	4	19	32	32	18,5
	10	19	137	32	18,5
	20	19	229	32	18,5
	50	19	504	32	18,5
	100	19	962	32	18,5
	200	19	1880	34	18,5
	500	47	4630	62	18,5
	1000	74	9210	93	18,5
	2000	139	18400	160	18,5
	4000	262	37000	280	18,5

CP341

(Acknowledgment from remote CP immediately on receiving frame)

Trans. Rate	User Data in Bytes	Processing Time CPU+CP (ms)	Trasmission Time of Frame (ms)	Processing Time of Remote CP (ms)	Trasmission Time of Ack. (ms)
76800	4	7,5	1,5	2,5	0,3
	10	7,5	2,2	4,3	0,3
	20	7,5	3,8	7,4	0,3
	50	10,5	8,2	16,2	0,3
	100	16,5	15,3	31,1	0,3
	200	26	30	61	0,3
	500	53	73	150	0,3
	1000	101	146	300	0,3
9600	4	7,5	10,4	1	2,3
	10	7,5	17,2	1	2,3
	20	7,5	29	1	2,3
	50	10,5	64	1,6	2,3
	100	16,5	121	1,6	2,3
	200	26	236	1,6	2,3
	500	53	580	1,6	2,3
	1000	101	1151	1,6	2,3
1200	4	7,5	82	1,5	18,5
	10	7,5	137	1,5	18,5
	20	7,5	229	1,5	18,5
	50	10,5	504	7	18,5
	100	16,5	962	7	18,5
	200	26	1880	7	18,5
	500	53	4630	7	18,5
	1000	101	9210	7	18,5

(Acknowledgment from remote CP after transferring frame to the CPU)

Trans. Rate	User Data in Bytes	Processing Time CPU+CP (ms)	Trasmission Time of Frame (ms)	Processing Time of Remote CP (ms)	Trasmission Time of Ack. (ms)
76800	4	7,5	1,5	11	0,3
	10	7,5	2,2	19	0,3
	20	7,5	3,8	19	0,3
	50	10,5	8,2	31	0,3
	100	16,5	15,3	47	0,3
	200	26	30	94	0,3
	500	53	73	202	0,3
	1000	101	146	404	0,3
9600	4	7,5	10,4	10	2,3
	10	7,5	17,2	12	2,3
	20	7,5	29	12	2,3
	50	10,5	64	15	2,3
	100	16,5	121	22	2,3
	200	26	236	28	2,3
	500	53	580	56	2,3
	1000	101	1151	104	2,3
1200	4	7,5	82	10	18,5
	10	7,5	137	17	18,5
	20	7,5	229	17	18,5
	50	10,5	504	22	18,5
	100	16,5	962	28	18,5
	200	26	1880	32	18,5
	500	53	4630	63	18,5
	1000	101	9210	110	18,5

Reference

B

Data Highway DF 1 Protocol

/1/ DF1 Protocol and Command Set,. Reference Manual
Publication 1770–6.5.16 Oktober 1996
Allen Bradley

Glossary

A

Address

An address is part of a STEP 7 statement and specifies what the processor should execute the instruction on. An address can be absolute or symbolic.

B

Block

Blocks are discrete parts of a user program and can be distinguished by their function, their structure, or their purpose.
STEP 7 provides the following types of blocks:

- Logic blocks (FB, FC, OB, SFB, SFC)
- Data blocks (DB, SDB)
- User-defined data types (UDT)

Block Call

A block call is the branch into the called block taken during program processing.

Block Parameters

Block parameters are token values within multipurpose blocks which are supplied with current values when the corresponding block is called.

C

Communications Processor

A programmable module for carrying out communication tasks such as networking and point-to-point connection.

Configuring

Selecting and arranging individual components of a programmable logic controller or installing required software, such as an operating system on the M7 automation computer, and adapting them for a specific use, such as by assigning parameters to modules.

Connection Configuration (only applies to CP 441-2)

Connection configuration is the specifying of a connection ID in the system function block. By means of the connection ID, the system function blocks can communicate between two communication endpoints.

CPU

The “Central Processing Unit” is the central module in the S7 programmable logic controller in which the user program is stored and processed. It consists of an operating system, processing unit, and communication interfaces.

CPU Operating System

The CPU operating system organizes all functions and procedures in the CPU which are not linked to a specific control task.

Cyclical Program Processing

In cyclical program processing, the user program runs in a continuously repeating loop known as a cycle.

Cycle Time

The cycle time is the length of time needed by the CPU to complete one processing of the user program.

D

Data Block (DB)

Data blocks are areas in the user program which contain user data. There are shared data blocks, which can be accessed by all logic blocks, and there are instance data blocks, which are associated with a particular function block (FB) call. Data blocks contain no logic instructions, in contrast to all other types of blocks.

Data Type

With the help of data types, you can specify how the value of a variable or a constant is to be used in the user program. There following two data types according to IEC 1131-3 available to users of SIMATIC S7:

- Elementary data types
- Complex data types

Default Setting

A default setting is an appropriate basic setting that is always used if no other value is entered.

Diagnostic Buffer

The diagnostic buffer is a buffered memory area, such as in CPUs, which is set up as a ring buffer. Diagnostic events are stored there in their order of occurrence.

Diagnostic Events

A record of a diagnostic event is entered in the CPU diagnostic buffer. The following events can occur:

- Error at module
- Error in process circuitry / connections
- System error in the CPU
- Operating mode transition in the CPU
- Error in a user program
- User-defined diagnostic event

Diagnostic Functions

Diagnostic functions consist of the entire system diagnostics, including detecting, evaluating, and reporting of errors in the programmable logic controller (PLC).

F

Function (FC)

According to the International Electrotechnical Commission's IEC 1131-3 standard, functions are logic blocks which do not have a "memory." A function allows you to transfer parameters in the user program, which means they are suitable for programming complex functions that are required frequently, such as calculations. Important: Since there is no memory available, the calculated values must be processed immediately after the FC call.

Function Blocks (FBs)

According to the International Electrotechnical Commission's IEC 1131-3 standard, function blocks are logic blocks with static data. A function block allows you to transfer parameters within the user program, which means they are suitable for programming complex functions that are required frequently, such as control systems and operating mode selection. Since function blocks have a "memory" in the form of the associated instance data block, their parameters, such as outputs, can be accessed at any time and at any point in the user program.

H

Hardware

The hardware is the entire set of physical and technical equipment for a programmable logic controller (PLC).

I

Instance Data Block

An instance data block stores the formal parameters and static data from function blocks (FB). An instance data block can be associated with a function block call or a function block call hierarchy.

Interface Module

The CP 441-2 interface module is responsible for the physical conversion of signals. You can adapt the communication processor to the interface connection of the communication partner by exchanging plug-in interface modules.

L

Loading to Destination System

The uploading of objects from the programming device into the load memory (memory card) of a connected programmable module.

Loading to Programming Device (PG)

The downloading of load objects from the load memory (memory card) of a connected programmable module to the programming device.

M

Module

Modules are plug-in printed circuit boards for programmable logic controllers (PLC).

Module Parameters

Module parameters are values used to set the behavior characteristics of a module. Some of these parameters (module-specific) can be changed in the user program.

O

Online / Offline

"Online" describes the state when a data connection exists between the programmable logic controller (PLC) and the programming device (PG); "Offline" means that such a data connection does not exist.

Online Help

STEP 7 enables you to display context-sensitive help on the screen while you are working with the programming software.

Operating Mode

The CPU for programmable logic controllers in SIMATIC S7/M7 operate in the following modes:

- STOP
- START
- RUN
- HOLD
- CONNECT
- UPDATE
- DIAGNOSTICS

P

Parameter

1. A variable in a STEP 7 logic block (see also "Actual Parameter" and "Formal Parameter")
2. A variable for setting the behavior characteristics of a module (one or more per module. Each module is initially supplied with an appropriate basic setting that can be changed by configuring the hardware.

The are two types of parameters, static and dynamic.

Parameter Assignment

Parameter assignment is the process of making settings governing the behavior characteristics of a module.

Parameter Assignment Interface CP: Point-to-Point Communication, Parameter Assignment

The parameter assignment interface *CP Point–To–Point Communication, Parameter Assignment* is used to assign parameters to the communications processor and also to assign driver-specific parameters.
These new parameters extend the standard parameters already available in each loadable driver.

Point-to-Point Connection

The communications processor uses the point-to-point connection to create the interface between a programmable logic controller and a communication partner.

Procedure

In data transmission, the operational sequence based a specific protocol is known as a procedure.

Process Image

The signal states of the digital input and output modules are stored in a process image in the CPU. There is a process image for inputs (PII) and one for outputs (PIQ).

Process Image for Inputs (PII)

The process image for inputs is read from the input modules before processing of the user program by the operating system.

Process Image for Outputs (PIQ)

The process image for outputs is transmitted at the end of the user program to the output modules by the operating system.

Process Interrupt

A process interrupt is an interruption of program processing in the processor of the programmable logic controller (PLC) by an external interrupt.

Programmable Logic Controller

A programmable logic controller (PLC) consists of at least one central processing unit (CPU), various input and output modules, along with operator interfaces.

Protocol

The communication partners involved in a data transmission must obey set rules for processing and carrying out data transfer. Such rules are known as protocols.

R

Rack

A rack is a carrier containing mounting slots for modules.

S

Software

This is the entire group of programs being used on a computer system. Software includes the operating system and the user programs.

STARTUP

The STARTUP operating mode is the transition from the STOP mode to the RUN mode. It can be selected in the following ways:

- At the operating mode switch, or
- After power on, or
- At the programming device.

There are three startup modes to chose from, cold restart, warm restart, and hot restart.

STEP 7

STEP 7 is the programming software for SIMATIC S7.

System Blocks

System blocks are different from other types of blocks in that they are already integrated in the S7–300/400 system and are available for previously-defined system functions. There are system data blocks, system functions, and system function blocks.

System Functions (SFCs)

A system function (SFC) is a function integrated in the operating system of the S7 CPU which, if necessary, can be called in the user program like a function (FC).

System Function Blocks (SFBs)

A system function block (SFB) is a function block integrated in the operating system of the S7 CPU which, if necessary, can be called in the user program like a function (FC).

T

Tool

A tool is a software accessory used for configuring and programming.

U

User Program

The user program contains all statements and declarations as well as data for signal processing used to control a system or process. It is associated with a active (programmable) module (such as CPU, FM) and can be subdivided into smaller structural units (blocks in S7 and tasks in M7).

V

Variable

A variable defines a piece of data with variable content which can be used in the STEP 7 user program. A variable consists of an address (such as M 3.1) and a data type (such as Bool) and can be identified by means of a symbolic name (such as BAND_ON).

W

Working Memory

Working memory is RAM memory in the CPU that is accessed by the processor while the user program is running.

Index

P

R

S

T

Siemens AG
A&D AS E 81

Oestliche Rheinbrueckenstr. 50
D-76181 Karlsruhe
Federal Republic of Germany

From:
Your Name: _
Your Title: _
Company Name: _
Street: _
City, Zip Code_ _
Country: _
Phone: _

Please check any industry that applies to you:

❒ Automotive
❒ Chemical
❒ Electrical Machinery
❒ Food
❒ Instrument and Control
❒ Nonelectrical Machinery
❒ Petrochemical
❒ Pharmaceutical
❒ Plastic
❒ Pulp and Paper
❒ Textiles
❒ Transportation
❒ Other _ _ _ _ _ _ _ _ _ _ _

Remarks Form

Your comments and recommendations will help us to improve the quality and usefulness of our publications. Please take the first available opportunity to fill out this questionnaire and return it to Siemens.

Please give each of the following questions your own personal mark within the range from 1 (very good) to 5 (poor).

1. Do the contents meet your requirements? ☐
2. Is the information you need easy to find? ☐
3. Is the text easy to understand? ☐
4. Does the level of technical detail meet your requirements? ☐
5. Please rate the quality of the graphics/tables: ☐

Additional comments: