SIEMENS

SIMATIC NET

Development Kit DK-5613 for the CP 5613/CP 5614

Manual

1 How to Use the DK-5613 3

2 The DOS Components 4

3 CP 5613 / CP 5614 Dual-port RAM Layout 8

4 Basic Operations 26

5 LDB File Structure 32

6 Glossary 39

C79000-G8976-C135-06 Release 6

We have checked the contents of this manual for agreement with the hardware described. Since deviations cannot be precluded entirely, we cannot guarantee full agreement. However, the data in this manual are reviewed regularly and any necessary corrections included in subsequent editions. Suggestions for improvement are welcome. Technical data subject to change.

C79000-G8976-C135-06

Siemens Aktiengesellschaft

Elektronikwerk Karlsruhe
Printed in the Federal Republic of Germany

SIEMENS

SIMATIC NET

Development Kit DK-5613 for the CP 5613/CP 5614

Manual C79000-G8976-C135-06

History of the document

Document release	Description	Comment
1	• Describes DK-5613 V1.0	Development release
2	• Describes DK-5613 V1.1	First customer release
3	• Describes DK-5613 V1.2 • Chapter 5 LDB File Structure	 Additional chapter
4	• Describes DK-5613 V2.0	OS-Dependent Parts of CI_DLL extracted
5	• Describes DK-5613 V3.0	Equidistance
6	• Describes DK-5613 V3.1	CP5613A2/CP5614A2

Note

We would point out that the contents of this product documentation shall not become a part of or modify any prior or existing agreement, commitment or legal relationship. The Purchase Agreement contains the complete and exclusive obligations of Siemens. Any statements contained in this documentation do not create new warranties or restrict the existing warranty.

We would further point out that, for reasons of clarity, these operating instructions cannot deal with every possible problem arising from the use of this device. Should you require further information or if any special problems arise which are not sufficiently dealt with in the operating instructions, please contact your local Siemens representative.

General

This device is electrically operated. In operation, certain parts of this device carry a dangerously high voltage.

WARNING !

!

Failure to heed warnings may result in serious physical injury and/or material damage.

Only appropriately qualified personnel may operate this equipment or work in its vicinity. Personnel must be thoroughly familiar with all warnings and maintenance measures in accordance with these operating instructions.

Correct and safe operation of this equipment requires proper transport, storage and assembly as well as careful operator control and maintenance.

Personnel qualification requirements

Qualified personnel as referred to in the operating instructions or in the warning notes are defined as persons who are familiar with the installation, assembly, startup and operation of this product and who possess the relevant qualifications for their work, e.g.:

– Training in or authorization for connecting up, grounding or labeling circuits and devices or systems in accordance with current standards in safety technology;

– Training in or authorization for the maintenance and use of suitable safety equipment in accordance with current standards in safety technology;

– First Aid qualification.

Contents

1 How to Use the DK-5613 ... 3

1.1 CP5613A2 / CP5614A2 ... 3

2 The DOS Components ... 4

2.1 Installation ... 5

2.2 DP_BASE.LIB / DPS_BASE.LIB ... 6

2.3 CI_DLL.LIB ... 7

3 CP 5613 / CP 5614 Dual-port RAM Layout ... 8

3.1 PLX Area ... 9

3.2 Download Area ... 10

3.2.1 CP 5613 and CP 5614 ... 10

3.2.2 CP 5613 A2 and CP 5614 A2 ... 11

3.3 Data Area ... 12

3.3.1 ctr_cpu ... 13

3.3.2 dp ... 13

3.3.3 cc ... 14

3.3.4 dp.ctr ... 14

3.3.5 dp.ef ... 15

3.3.6 dp.ef.input and dp.ef.diag ... 15

3.3.7 dp.pi ... 15

3.3.8 dp.pi.slave_in ... 16

3.3.9 dp.pi.slave_out ... 16

3.3.10 dp.pi.slave_diag ... 16

3.3.11 dp.info_watch ... 17

3.3.12 dp.info_watch.slave_info ... 18

3.3.13 dp.info_watch.master_info ... 18

3.3.14 dp.info_watch.aspc2_event ... 19

3.3.15 dp.info_watch.aspc2_buspara ... 20

3.3.16 dp.info_watch.slavemod_data ... 21

3.3.17 dp.info_watch.slavemod_data.sm_diag ... 22

3.3.18 cc.adm_cp ... 23

3.3.19 cc.block ... 24

3.3.20 cc.block.usr_header ... 25

4 Basic Operations ... 26

4.1 Initializing the CP ... 26

4.1.1 CP5613/CP5614 ... 26

4.1.2 CP 5614A2/CP5614A2 27

4.2 Sending Data to the Command Channel and Receiving the Response 28

4.3 Accessing Input, Output, and Diagnostic Data 28

4.4 Using the CP Interrupt 29

4.5 Using Fast Logic 30

4.6 Events 30

4.7 Setting the Diagnostic Data of the Slave Module (CP 5614 only) 31

5 LDB File Structure 32

5.1 Version-Block (optional) 32

5.2 System-Block 33

5.3 DP-Block 35

5.4 FMS-Block (optional) 38

5.5 End-Block 38

6 Glossary 39

1 How to Use the DK-5613

This chapter contains general information on porting the driver to any operating system.

The development kit allows customers to access the CP 5613/CP 5614 under operating systems other than those for which drivers are available.
The starting point of this process is the source code supplied with this package which provides an interface for MS-DOS systems as an example.

This document is not intended to enable you to write a driver from scratch, but will guide you through the source modules and help you to port the sources.

See chapter 2.1 for instructions for the installation of the required components.

Note: Always use this code in conjunction with the firmware file in this package! If you use driver and firmware versions that do not match, you may encounter unpredictable results!

The original tools used were Watcom C/C++ V10.5 or Open Watcom 1.1 from www.openwatcom.org.

You will need the required development tools for driver development of your target operating system.

1.1 CP5613A2 / CP5614A2

Concerning the handling by the DK there are only few differences between a CP 5613 and a CP 5613 A2.

There are

- new DeviceIDs
- the download structure in dual port ram has changed
- different firmware files to load
- start and reset are different

This differences are handled by CI_DLL. You can search for CP5613A2 in the sources of CI_DLL. See also Chapter 3 and 4 in this document.

2 The DOS Components

This chapter describes the structure of the components needed for the CP 5613 / CP 5614 under MS-DOS.

The DOS interface has the following restrictions:

- Semaphores have to be implemented depending on the target-OS (if needed)

2.1 Installation

- Copy the complete directory *cd:*\sw\dos to your *working-directory*\dos. Be sure to use a copy-command which removes the read-only attribute of all files or remove the read-only attribute manually from all copied files.
- Setup the path-information in the file setenv.bat in your *working-directory*\dos
- Adapt the calls to your development system and the make-syntax in the .mk*-files in all subdirectories of *working-directory*\dos, if you don't use the Watcom C/C++ compiler
- Adapt all operating system dependend parts in the file dos\ci_dll\src\os_adapt.c and dos\ci_dll\src\os_adapt.h
- Generate all modules by calling 'm.bat' in the make-directory of each subdirectory of 'dos'
- Test your generated components

2.2 DP_BASE.LIB / DPS_BASE.LIB

These libraries are the counterpart of the DP_BASE.DLL and DPS_BASE.DLL in the standard Windows NT interface.

The sources contain no code specific to a particular operating system outside the #ifdef WIN32, #endif brackets.

2.3 CI_DLL.LIB

This library is the counterpart of the CI_DLL.DLL and CP5613.SYS in the standard Windows XP software.

There are to modules for generating the CI_DLL.LIB. The general code is located in the file ci_dll.c, all operating-system dependent code is located in the file os_adapt.c.

You must adapt the code in os_adapt.c and os_adapt.h to your operating system.
Detailed information can be found in the function-headers of os_adapt.c.
The modules can only be compiled for 32-bit flat systems.

3 CP 5613 / CP 5614 Dual-port RAM Layout

This chapter describes the layout of the areas in the dual-port RAM in detail.

The dual-port RAM areas are mapped by the Plug&Play Bios during startup.
Drivers must read the addresses from the PCI configuration register:

PCI Config. Reg.	Content
...	...
0x10	Physical address of PLX
0x14	I/O address (unused)
0x18	Physical address of download area
0x1C	Physical address of data area
...	...

The vendor ID and device ID can be found in the #defines VENDOR_ID (0x110A), DEVICE_ID_5613A1 (0x3142) and DEVICE_ID_5613A2 (0x4029).

Some operating systems have the PCI functionality integrated, so there is no need to directly access the PCI configuration register.

3.1 PLX Area

The PLX area points to the PCI local configuration registers:

Offset	Type	Description	Comment
0x00	DPR_WORD	LAS0_Range_lsw;	reserved
0x02	DPR_WORD	LAS0_Range_msw;	reserved
0x04	DPR_WORD	LAS1_Range_lsw;	reserved
0x06	DPR_WORD	LAS1_Range_msw;	reserved
0x08	DPR_WORD	res_w_00[0x06];	reserved
0x14	DPR_WORD	LAS0_Base_lsw;	reserved
0x16	DPR_WORD	LAS0_Base_msw;	reserved
0x18	DPR_WORD	LAS1_Base_lsw;	reserved
0x1A	DPR_WORD	LAS1_Base_msw;	reserved
0x1C	DPR_WORD	res_w_01[0x06];	reserved
0x28	DPR_WORD	LAS0_Bus_lsw;	reserved
0x2A	DPR_WORD	LAS0_Bus_msw;	reserved
0x2C	DPR_WORD	LAS1_Bus_lsw;	reserved
0x2E	DPR_WORD	LAS1_Bus_msw;	reserved
0x30	DPR_WORD	res_w_02[0x06];	reserved
0x3C	DPR_WORD	CS0_lsw;	reserved
0x3E	DPR_WORD	CS0_msw;	reserved
0x40	DPR_WORD	CS1_lsw;	reserved
0x42	DPR_WORD	CS1_msw;	reserved
0x44	DPR_WORD	res_w_03[0x04];	reserved
0x4C	DPR_BYTE	LB_INT_CONTROL_lsw;	0x41: interrupt enabled 0x01: interrupt disabled (see Chapter 5.4)
0x4D	DPR_BYTE	HB_INT_CONTROL_lsw;	reserved
0x4E	DPR_WORD	INT_CONTROL_msw;	reserved
0x50	DPR_BYTE	LB_MULTI_CONTROL_lsw;	reserved
0x51	DPR_BYTE	HB_MULTI_CONTROL_lsw;	write 0x15 and then 0x1D to issue an interrupt to the CP
0x52	DPR_WORD	MULTI_CONTROL_msw;	reserved

3.2 Download Area

The download area is used to copy the firmware and the ldb file. This must be done before starting the CP. After the CP is started, the host cannot access this area.

This structure is different for CP5613/CP5614 and CP 5613A2/CP5614A2. The structure for each card type (CP5613_DOWNLOAD_T, CP5613A2_DOWNLOAD_T) is now part of the union CP5613_DOWNLOAD_UT.

3.2.1 CP 5613 and CP 5614

The structure shown in the figure below is defined in the C structure CP5613_DOWNLOAD_T in the file ci_5613.h. The layout has changed with DK 5613 V 3.1 because of the bigger firmware.

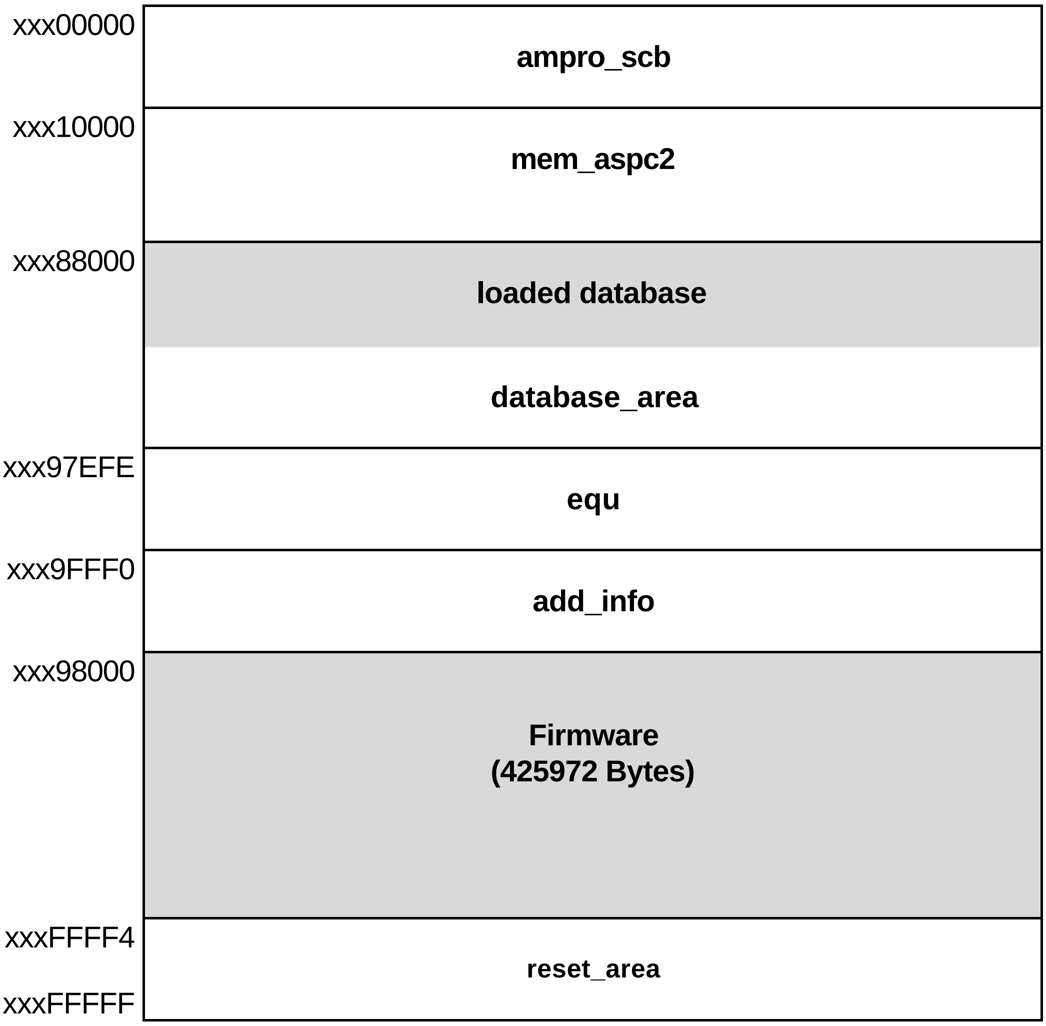


Figure 1: Download Area Layout

Ampro_scb, mem_aspc2 and reset area are reserved areas.
The add_info structure must be set to 0.

The ldb file must be located at the beginning of the database area.

3.2.2 CP 5613 A2 and CP 5614 A2

The structure shown in the figure below is defined in the C structure CP5613A2_DOWNLOAD_T in the file ci_5613.h.

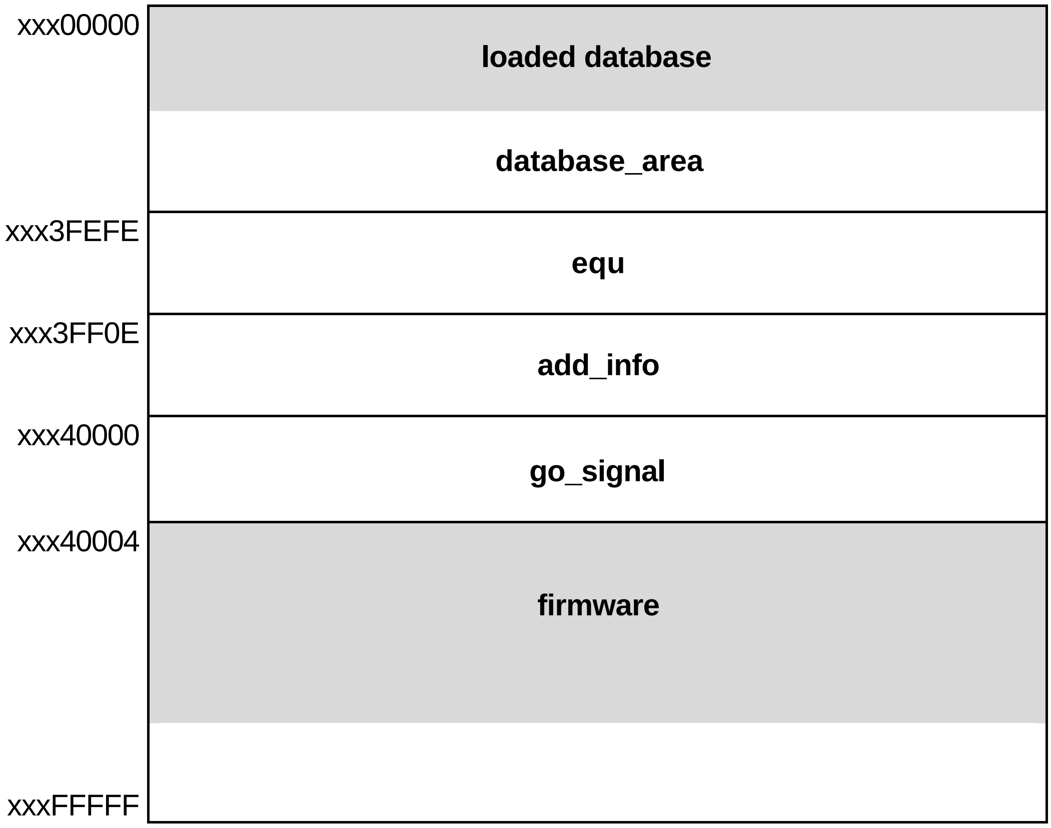


Figure 2: Download Area Layout A2

The add_info structure must be set to 0.

The ldb file and the firmware file must be located at the beginning of the area.

The go_signal should be readable as 0xa5a5a5a5. You can use this to check if the download area is correct mapped.

3.3 Data Area

The data area is used to communicate with the CP during runtime.

The structure shown in the figure below is defined in the C structure DPR_CP5613_ALL_T in the file ci_5613.h.

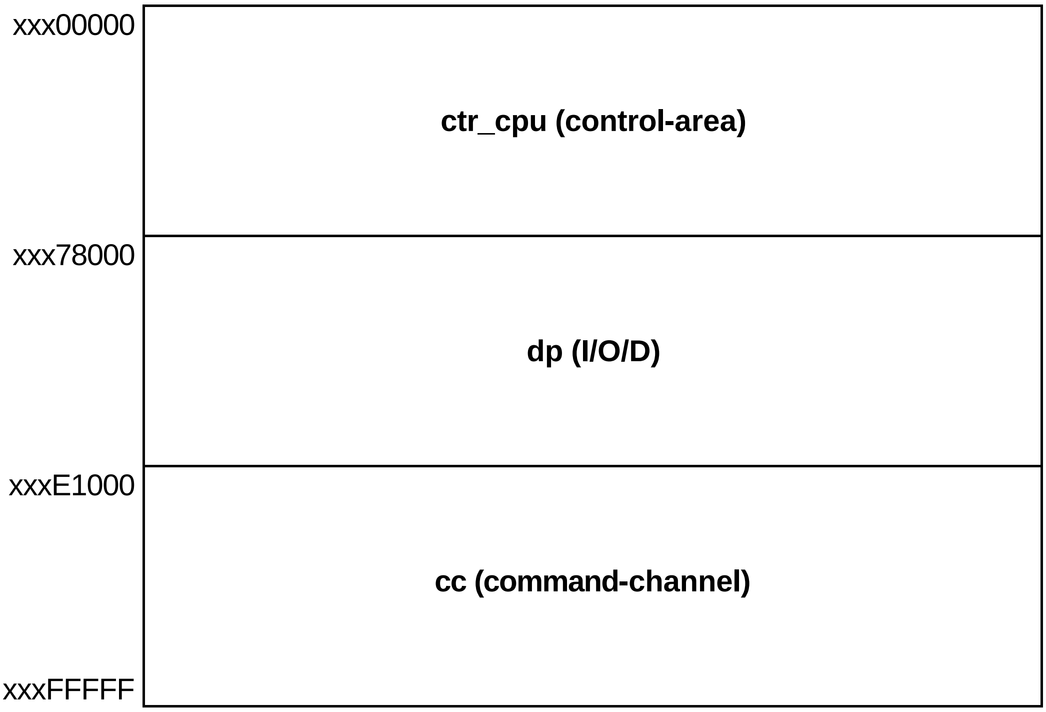


Figure 2: Data Area Layout

3.3.1 ctr_cpu

Layout of ctr_cpu (structure DPR_CP5613_CTR_CPU_T in file ci_5613.h):

Offset	Type	Description	Comment
0x00000	DPR_WORD	C_res_en	0x00 = go all 0x01 = reset CPU, go ASPC2 0x02 = go CPU, reset ASPC2 0x03 = reset all
0x04000	DPR_WORD	C_int_cp_to_host	not used from host side
0x08000	DPR_WORD	C_info_cp_int	Interrupt source bit 0: DP cycle start interrupt bit 1: input data changed bit 2: diagnostic data changed bit 3: new data in command-channel
0x0C000	DPR_WORD	C_ack_dp_cycle_int	host ackn. of DP cycle start int
0x0E000	DPR_WORD	C_ack_pae_int	host ackn. of input data change int
0x10000	DPR_WORD	C_ack_pad_int	host ackn. of diag data change int
0x12000	DPR_WORD	C_ack_cp_to_host_int	host acknowledge of cp int (new data in command channel)

3.3.2 dp

Layout of dp (structure DPR_CP5613_DP_T):

Name	Type	Description
ctr	DPR_CP5613_CTR_T	control data
ef	DPR_CP5613_EF_T	event and filter control
pi	DPR_CP5613_PI_T	process image
info_watch	DPR_CP5613_INFO_WATCH_T	additional information

3.3.3 cc

Layout of cc (structure DPR_CP5613_DP_T):

Name	Type	Description
adm_cp	ADM_CP_T	administrative data
block[CI_BLOCK_COUNT]	CI_DATA_BUFFER_T	communication tiles
reserved_b_01[0x0A8]	DPR_BYTE	unused area
trace_driver[0x1200]	DPR_BYTE	trace area for driver

3.3.4 dp.ctr

Layout of dp.ctr (structure DPR_CP5613_CTR_T):

Name	Type	Description
D_cycle_start_mask	DPR_WORD	for future use
no_ram_here_reserved_b_00[0x7FFE]	DPR_BYTE	unused
D_lock_in_slave_adr	DPR_WORD	slave-address for consistent input data
no_ram_here_reserved_b_01[0x7FFE]	DPR_BYTE	unused
D_out_slave_adr	DPR_WORD	slave address for consistent output data
no_ram_here_reserved_b_02[0x7FFE]	DPR_BYTE	unused
D_lock_diag_slave_adr	DPR_WORD	slave address for consistent diag data
no_ram_here_reserved_b_03[0x7FFE]	DPR_BYTE	unused
no_ram_here_reserved_b_04[0x8000]	DPR_BYTE	unused

3.3.5 dp.ef

Layout of dp.ef (structure DPR_CP5613_EF_T):

Name	Type	Description
input[DPR_MAX_SLAVE_NR]	DPR_MASK_DATA_INT_T	input masks; see below
reserved_b_00[0x8000]	DPR_BYTE	unused
diag[DPR_MAX_SLAVE_NR]	DPR_MASK_DATA_INT_T	diag masks; see below
reserved_b_01[0x8000]	DPR_BYTE	unused

3.3.6 dp.ef.input and dp.ef.diag

Layout of dp.ef.input and dp.ef.diag (structure DPR_MASK_DATA_INT_T):

Name	Type	Description
req_mask	DPR_BYTE	DPR_DATA_INT_CLEAR_AND_UNMASK: no request, int enabled DPR_DATA_INT_CLEAR_AND_MASK: no request, int disabled DPR_DATA_CHANGE: request, int disabled other values: invalid
no_ram_here_dont_touch[0x00FF]	DPR_BYTE	unused

3.3.7 dp.pi

Layout of dp.pi (structure DPR_CP5613_PI_T):

Name	Type	Description
slave_in[DPR_MAX_SLAVE_NR]	DPR_SLAVE_IN_T	input data
slave_out[DPR_MAX_SLAVE_NR]	DPR_SLAVE_OUT_T	output data
slave_diag[DPR_MAX_SLAVE_NR]	DPR_SLAVE_DIAG_T	diag data

3.3.8 dp.pi.slave_in

Layout of dp.pi.slave_in (structure DPR_SLAVE_IN_T):

Name	Type	Description
data[DPR_SLAVE_DATA_SIZE]	DPR_BYTE	input data
reserved_b_00[0x0A]	DPR_BYTE	reserved

3.3.9 dp.pi.slave_out

Layout of dp.pi.slave_out (structure DPR_SLAVE_OUT_T):

Name	Type	Description
data[DPR_SLAVE_DATA_SIZE]	DPR_BYTE	output data
reserved_b_00[0x0A]	DPR_BYTE	reserved

3.3.10 dp.pi.slave_diag

Layout of dp.pi.slave_diag (structure DPR_SLAVE_DIAG_T):

Name	Type	Description
data[DPR_SLAVE_DATA_SIZE]	DPR_BYTE	diag data
diag_len	DPR_WORD	actual diag len
diag_count	DPR_WORD	number of diag updates for this slave
reserved_w_00[0x03]	DPR_WORD	reserved

3.3.11 dp.info_watch

Layout of dp.info_watch (structure DPR_CP5613_INFO_WATCH_T):

Name	Type	Description
slave_info[DPR_MAX_SLAVE_NR]	DPR_SLAVE_INFO_T	see table below
master_info	DPR_MASTER_INFO_T	see table below
aspc2_event	DPR_ASPC2_EVENT_T	see table below
aspc2_buspara	DPR_ASPC2_BUSPARA_T	see table below
user_watchdog[0x170]	DPR_BYTE	for future use
activated_fast_logic[0x04]	DPR_WORD	0xFFFF: fast logic triggered
slavemod_data	DPR_SLAVEMOD_DATA_T	data-structure for slave module (CP 5614)
equ_error	DPR_CP5613_EQU_ERROR	for future use
reserved_b_10[0x3E8]	DPR_BYTE	

3.3.12 dp.info_watch.slave_info

Layout of dp.info_watch.slave_info (structure DPR_SLAVE_INFO_T):

Name	Type	Description
slave_in_database	DPR_WORD	0: slave is not configured 1: slave is configured
slave_type	DPR_WORD	DP_SLV_TYP_EMPTY: no DP slave DP_SLV_TYP_DP: DP slave DP_SLV_TYP_DPV1: DPV1 slave
slave_out_byte	DPR_WORD	number of output bytes
slave_in_byte	DPR_WORD	number of input bytes
slave_state	DPR_WORD	DPR_SLV_NOT_READY DPR_SLV_READY
detail	DPR_WORD	reserved
alarm_flag	DPR_WORD	DPV1 alarm pending
reserved_w_00	DPR_WORD	unused

3.3.13 dp.info_watch.master_info

Layout of dp.info_watch.master_info (structure DPR_MASTER_INFO_T):

Name	Type	Description
USIF_state	DPR_WORD	state of the master
ident_number	DPR_WORD	PNO ident number
hw_version	DPR_WORD	hardware version
fw_version	DPR_WORD	firmware version
reserved_w_00[4]	DPR_WORD	unused

3.3.14 dp.info_watch.aspc2_event

Layout of dp.info_watch.aspc2_event (structure DPR_ASPC2_EVENT_T):

Name	Type	Description
reserved_tick_5ms	DPR_WORD	reserved
on_double_token	DPR_WORD	counter for double token errors
on_timeout	DPR_WORD	counter for timeout errors
on_syni_error	DPR_WORD	counter for syni errors
on_hsa_error	DPR_WORD	counter for hsa errors
on_response_error	DPR_WORD	counter for response errors
on_las_useless	DPR_WORD	counter for las useless errors
on_rec_frame_overflow	DPR_WORD	counter for receive frame overflow errors
on_fifo_error	DPR_WORD	counter for fifo errors
on_req_length_error	DPR_WORD	counter for request length errors
off_pass_token_error	DPR_WORD	counter for pass token errors
off_ts_adr_error	DPR_WORD	counter for ts-adr errors
off_hsa_error	DPR_WORD	counter for hsa errors
in_ring	DPR_WORD	counter for in-ring events
out_of_ring	DPR_WORD	counter for out-of-ring events
bus_control_error	DPR_WORD	bus short cut

3.3.15 dp.info_watch.aspc2_buspara

Layout of dp.info_watch.aspc2_buspara (structure DPR_ASPC2_BUSPARA_T):

Name	Type	Description
ts	DPR_BYTE	station address
baud_rate	DPR_BYTE	DP_M_BAUDRATE_9K6, DP_M_BAUDRATE_19K2, DP_M_BAUDRATE_93K75, DP_M_BAUDRATE_187K5, DP_M_BAUDRATE_500K, DP_M_BAUDRATE_750K, DP_M_BAUDRATE_1M5, DP_M_BAUDRATE_3M, DP_M_BAUDRATE_6M, DP_M_BAUDRATE_12M, DP_M_BAUDRATE_31K25, DP_M_BAUDRATE_45K45
tsl	DPR_WORD	see EN50170
min_tsdr	DPR_WORD	see EN50170
max_tsdr	DPR_WORD	see EN50170
tqui	DPR_BYTE	see EN50170
tset	DPR_BYTE	see EN50170
ttr	DPR_DWORD	see EN50170
g	DPR_BYTE	see EN50170
hsa	DPR_BYTE	see EN50170
max_retry_limit	DPR_BYTE	see EN50170
station_type	DPR_BYTE	see EN50170
trdy	DPR_WORD	see EN50170
tid1	DPR_WORD	see EN50170
tid2	DPR_WORD	see EN50170
tmsi	DPR_DWORD	see EN50170
tmsi_reserve	DPR_WORD	see EN50170
tbus_control_in_ring	DPR_WORD	see EN50170
tbus_control_out_of_ring	DPR_WORD	see EN50170
acyc_req_ctr	DPR_WORD	see EN50170
mode_clock_sync	DPR_BYTE	see EN50170
delay_time_ctr_clock_sync	DPR_BYTE	see EN50170
mode_equ_dis	DPR_BYTE	see EN50170
master_equ_dis	DPR_BYTE	see EN50170
BpFlag	DPR_BYTE	see EN50170
MinSlaveInterval	DPR_WORD	see EN50170
PollTimeout	DPR_WORD	see EN50170

DataControlTime	DPR_WORD	see EN50170
reserved_b_00[1]	DPR_BYTE	unused

3.3.16 dp.info_watch.slavemod_data

Layout of dp.info_watch.slavemod_data (structure DPR_SLAVEMOD_DATA_T)
(CP 5614 only)

Name	Type	Description
sm_diag	DPR_SM_DIAG_T	diagnostic handling for slave module
act_gc	DPR_WORD	last global control of slave module
gc_ctr	DPR_WORD	count of global control commands received by slave module
baud_state	DPR_WORD	DPS_BAUD_SEARCH: baudrate search DPS_BAUD_FOUND: baudrate found DPS_BAUD_FOUND_WD: baudrate found, watchdog activated
baud_rate	DPR_WORD	DPS_BD_9K6, DPS_BD_19K2, DPS_BD_45K45, DPS_BD_93K75, DPS_BD_187K5, DPS_BD_500K, DPS_BD_1M5, DPS_BD_3M, DPS_BD_6M, DPS_BD_12M,
data_ex_state	DPR_WORD	DPS_OFFLINE, DPS_WAIT_PRM, DPS_WAIT_CFG, DPS_DATA_EX
reserved[246]	DPR_BYTE	unused

3.3.17 dp.info_watch.slavemod_data.sm_diag

Layout of dp.info_watch.slavemod_data.sm_diag (structure DPR_SM_DIAG_T) (CP 5614 only) :

Name	Type	Description
request	DPR_WORD	request counter: increment to set new diagnostic data
response	DPR_WORD	response counter: incremented when slave module accepted new diagnostic data
len	DPR_WORD	user diag data len
state	DPR_WORD	Bit field with bits to be set in the standard part of the diagnostic data: DPS_EXT_DIAG, DPS_STAT_DIAG, DPS_EXT_DIAG_OV
diag_data[238]	DPR_BYTE	user diag data
reserved[10]	DPR_BYTE	unused

3.3.18 cc.adm_cp

Layout of cc.adm_cp (structure ADM_CP_T):

Name	Type	Description
fw_version[0x20]	DPR_BYTE	firmware version string
fw_error	CP_ADM_ERR_T	firmware error information
to_send_index	DPR_WORD	index of tile to be sent
cp_ready	DPR_WORD	This flag is set by the CP and reset by the host. ADM_HC_DATA_HERE: tile inserted by CP ADM_HC_DATA_TAKEN: tile processed by host
host_ready	DPR_WORD	This flag is set by the host and reset by the CP. ADM_HC_DATA_HERE: tile inserted by host ADM_HC_DATA_TAKEN: tile processed by CP
block_return_head	DPR_WORD	index of tile to be released
data_return_head	DPR_WORD	index of data tile to be processed
status	DPR_WORD	must be set to STATUS_INIT before starting CP and is set to STATUS_RUNNING by the firmware if the CP is running.
in_ice	DPR_WORD	reserved
cpu_id	DPR_WORD	reserved
aspc2_id	DPR_WORD	reserved
cp_life_counter	DPR_DWORD	reserved
fw_int_status	DPR_WORD	FW_INT_NO_ACTION: no operation FW_INT_TRIGGER: triggers an empty interrupt to the host
pc_int_status	DPR_WORD	reserved
cp_perf_counter	DPR_WORD	reserved
test	ADM_TEST_S	reserved
fastlogic_req	DPR_WORD	fast logic event occurred
dp_access	DPR_WORD	for driver use (not processed by firmware)
uhr	CP_DPR_SCI_TIME_T	reserved

sap[SAP_SIZE]	CP_DPR_SAP_T	reserved
reserved_3[0x064]	DPR_BYTE	unused

3.3.19 cc.block

Layout of cc.block (structure CI_DATA_BUFFER_T):

Name	Type	Description
usr_header	CI_REQUEST_HEADER_T	header
next_block	DPR_WORD	chained block index
next_request	DPR_WORD	chained request index
block_fill_length	DPR_WORD	number of valid bytes in block when sending block to CP
own_index	DPR_WORD	index of this tile
search_pat_0	DPR_WORD	must be set to CI_SEARCH_PAT_FW, if block is to be sent; must be set to CI_SEARCH_PAT_DRV, if block has been received
res_for_subsys	DPR_DWORD	reserved
reserved[1]	DPR_WORD	reserved
data[]	DPR_BYTE	user data

3.3.20 cc.block.usr_header

Layout of cc.block.usr_header (structure CI_REQUEST_HEADER_T):

Name	Type	Description
user_handle	DPR_DWORD	used by driver
reserved_dw_00	DPR_DWORD	unused
opcode	DPR_DWORD	opcode of command
timeout	DPR_DWORD	timeout for receive
order_id	DPR_DWORD	for driver use (not processed by firmware)
response	DPR_DWORD	error code of firmware
buf_length	DPR_WORD	length of user buffer
buf_fill_length	DPR_WORD	number of valid data in block when receiving from CP
alloc_index	DPR_DWORD	internal counter of firmware
alloc_index	DPR_DWORD	internal counter of firmware
stamp_send	DPR_DWORD	internal timestamp of firmware
stamp_receive	DPR_DWORD	internal timestamp of firmware

4 Basic Operations

This chapter describes operations essential to working with the CP 5613 / CP 5614.

4.1 Initializing the CP

There are some differences between CP5613/CP5614 and CP5613A2/CP5614A2.
See also the function CI_start_cp in CI_DLL.

4.1.1 CP5613/CP5614

To start the CP do the following:

Get the pointers to the memory areas of the CP from the PCI-BIOS
Make sure that the CP has been reset, so that you can access the download area by setting dpr->ctr_cpu.C_res_en=0x03.
copy the database file to the start of down->database_area
copy the firmware-file to down->firmware
set all bytes of down.a1->add_info to 0 dpr->cc.adm_cp.status=STATUS_INIT
start the CP by setting dpr->ctr_cpu.C_res_en=0x02
wait until dpr->cc.adm_cp.status==STATUS_RUNNING

The card is then running and you can use the command channel to switch the master to operate.

4.1.2 CP 5614A2/CP5614A2

To start the CP do the following:

Get the pointers to the memory areas of the CP from the PCI-BIOS
Check access to download area: go_signal == 5a5a5a5a ?
copy the database file to the start of down->database_area
copy the firmware-file to the start of down->firmware
initialize variables: set all bytes of down.a2->add_info to 0 dpr->cc.adm_cp.status=STATUS_INIT
start the CP with *pGoSignal = 0xa5a5a5a5;
wait until dpr->cc.adm_cp.status==STATUS_RUNNING

The card is then running and you can use the command channel to switch the master to operate.

To reset a CP 5613A2 / CP5614A2 do:

ptr_dpram->ctr_cpu.C_res_en=CP5613_RESET
Make a delay Wait for 10 ms
cp_data[cp_index].ptr_dpram->ctr_cpu.C_res_en=CP5613_ENABLE;
Make a delay Wait for 300 ms
The timing is critical. If the delays are shorter than this values the PC may freeze.

4.2 Sending Data to the Command Channel and Receiving the Response

To send a request to the CP, follow the steps outlined below:

locate a free block in the dpr
initialize the block header
copy the user data to the block
wait until dpr->cc.adm_cp.host_ready is 0
set dpr->cc.adm_cp.to_send_index to the used buffer index
set dpr->cc.adm_cp.host_ready to 1
issue an interrupt to the CP by setting plx->HB_MULTI_CONTROL_lsw first to 0x15 and then to 0x1D

The CP then processes the command. You can now poll for the response or you can use the interrupt generated by the CP if the response data is available.

check dpr->cc.adm_cp.cp_ready. (this variable is set to 1 by the CP if it placed a response block in the command channel)
get the block index from dpr->cc.adm_cp.data_return_head
inform the CP, that you received the block by setting dpr->cc.adm_cp.data_return_head=0 and dpr->cc.adm_cp.cp_ready=0

You can now process the response data.

4.3 Accessing Input, Output, and Diagnostic Data

Once the CP is in operate mode you can access the input, output and diagnostic data directly in the dual-port ram. Refer to the CP 5613/CP 5614 manual for more information on how to access this data.

4.4 Using the CP Interrupt

The CP triggers an interrupt in some situations. The driver can use this interrupt instead of polling the events. This is, however, optional and is not mandatory.

It is also possible to get interrupts at start and end of each dp cycle if constant scan time mode is configured in your dp database. You can create such databases only with Step 7 or NCM PC Edition.

The CP uses the INTA# line of the PCI bus in shareable mode. The used IRQ can be seen in the PCI configuration register 3Ch.

The interrupt is disabled after startup. The interrupt can be enabled by writing 0x41 to the configuration register 0x4C. The interrupt can be disabled by writing 0x01 to the configuration register 0x4C.

After an interrupt is generated, the interrupt must be enabled again. The firmware also requires an acknowledgment of the interrupt (dpr->ctr_cpu.C_ack_cp_to_host_int = 0).

The causes of an interrupt are as follows:

- slave data changed
- slave diagnostic data changed
- dp cyle start and end
- new data available in command channel
- fast logic event occurred

See also do_cp_int in ci_dll.c

4.5 Using Fast Logic

The fast logic processing is handled by the firmware. The only action required of the user is to activate the fast logic by sending the fastlogic_on request through the command channel.

If a fast logic event occurs, the specified output bits are set by the firmware and the user is informed by an interrupt if this is enabled. Otherwise, the fast logic events can be polled in the dual-port RAM.

When using fast logic, make sure that the fast logic output bytes are not overwritten cyclically.

4.6 Events

Refer to the CP 5613/CP 5614 users manual for information on activating the events. If an event occurs, an interrupt is generated when data changes (if enabled).

4.7 Setting the Diagnostic Data of the Slave Module (CP 5614 only)

The diagnostic data of the slave module is set directly using the dual-port-ram instead of using the command channel.

fill the array dpr->infowatch.slavemod_data.sm_diag.diag_data with your diagnostic data
set dpr->infowatch.slavemod_data.sm_diag.diag_len with your diagnostic data length
set dpr->infowatch.slavemod_data.sm_diag.diag_state with your diagnostic data state
increment dpr->infowatch.slavemod_data.sm_diag.request to trigger the firmware
before setting another diag request you should wait until dpr->infowatch.slavemod_data.sm_diag.result contains the value of ... .request

5 LDB File Structure

The CP5613/CP5614 uses a binary configuration file. This file can be exported with the COM-Profibus NCM PC Edition or Step 7 configuration-tool. This chapter describes the structure of those files.

The LDB-file consists of several blocks of information. Not all generated bocks are necessary for a regular operation of the CP5613/CP5614 (see table below).

Grayed fields are constants. Non-grayed fields are calculated.

It is highly recommended to use the COM-Profibus, NCM PC or Step 7 - generated LDB-file. We cannot guarantee that the master is working correctly if you use other data, than that tools generates.

Name	Description
Version-Block	Version information (optional)
System-Block	System data block
DP-Block	DP Information
FMS-Block	(optional)
End-Block	(optional)

5.1 Version-Block (optional)

Name	Type	Description
Block-ID	BYTE	0x00
Block-length	DWORD	length of data-block (without header)
data-block	BYTE[]	data is not relevant

5.2 System-Block

Name	Type	Description
Block-ID	BYTE	0x79
Block-length	DWORD	length of data-block (without header)
busparameter header block Start		
parameter-block-number	BYTE	0x01
parameter-block-version	BYTE	0x00
parameter-block-length	WORD	0x0009
local-edit	BYTE	0x01
max-master	BYTE	0x7e
busparameter-calculation	BYTE	0x00: user-defined 0x01: universal 0x02: fast 0x03: dp
subnet-profile	BYTE	0x00
slave-count	BYTE	number of slaves
busparameter header block End		
busparameter block Start		
parameter-block-number	BYTE	0x02
parameter-block-version	BYTE	0x00
parameter-block-length	WORD	0x23
hsa	BYTE	default: 0x7e
master-addr	BYTE	station-address of master
station-type	WORD	0x01
baud-rate	WORD	0x00: 9.6kBd 0x01: 19.2 kBd 0x02: 93.75 kBd 0x03: 187.5kBd 0x05: 500 kBd 0x07: 1.5 MBd 0x08: 3 MBd 0x09: 6 MBd 0x0a: 12 MBd 0x0c: 45.45 kBd
redundancy	WORD	0x0000

retry-counter	WORD	0x0001
default-SAP	BYTE	0x34
network-SAP	BYTE	0x00
TSL	WORD	see EN50170 (Unit: tBit)
TQUI	WORD	not used
TSET	WORD	see EN50170 (Unit: tBit)
min.TSDR	WORD	see EN50170 (Unit: tBit)
max.TSDR	WORD	see EN50170 (Unit: tBit)
TTR	DWORD	target rotation time (Unit: tBit)
gap-factor	BYTE	see EN50170
in_ring_desired	WORD	0x01
phys -layer	WORD	0x00
busparameter block End		
block Start		
parameter-block-number	BYTE	0x05
parameter-block-version	BYTE	0x00
parameter-block-length	WORD	0x0004
block End		
Equidistance block Start		
BlockID	BYTE	30
ProfileFeatures	WORD	Bit 0:=1 Bit 1-15:=0 (reserved)
BlockLength	WORD	5 + number of Bytes in AddData
AddData	AddData	
Equidistance block End		

AddData

AddDataID	BYTE	0 :Busprofile Version 2.0
AddDataLen	WORD	Length of Add Data
reserved [63]	BYTE	

5.3 DP-Block

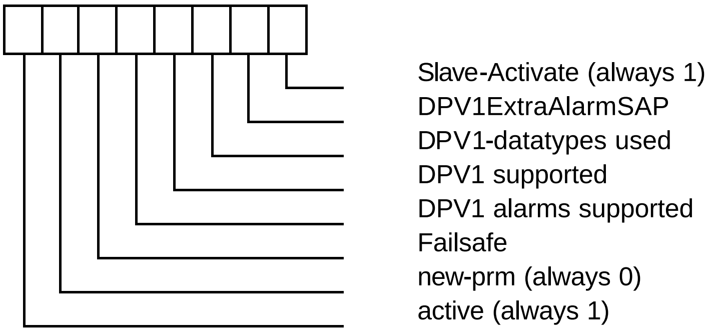
Name	Type	Description
Block-ID	BYTE	0x80
Block-length	DWORD	length of data-block (without header)
Parameter-block Start		
parameter-block-number	BYTE	0x01
parameter-block-version	BYTE	0x00
parameter-block-length	WORD	0x0025
CtrPollToResetStatLst	BYTE	0x01
DataControlTime	WORD	see EN50170 (Unit: 10 msec)
MasterPollTimeout	WORD	see EN50170 (Unit: 1 msec)
Master2PollTimeout	WORD	200
GSD-file name	BYTE[14]	"CP5412A2.GSD" (not NCM-PC)
reservedWdTimeout	WORD	see EN50170 (Unit: 10 msec)
reserved	BYTE	0
CyclGlobCtrlIdent	BYTE	see EN50170
MaxMinSlaveIntervall	WORD	see EN50170 (Unit: 1 msec)
DPDelayTime	WORD	0
NotNormSlaveCount	WORD	0x0000
NormSlaveCount	WORD	number of connected slaves
Parameter-block End		
slave blocks	slave-block-[NormSlaveCount]	slave-data blocks

Slave-Block

Name	Type	Description
parameter-block-number	BYTE	0x02
parameter-block-version	BYTE	0x00
parameter-block-length	WORD	length of this Slave-Block
Slv Header Start		
slave-header-length	WORD	0x002d
slave-address	BYTE	Station-Address of Slave
norm-slave-key	BYTE	0x00
slave-name	BYTE[24]	Name of Slave
GSD-filename	BYTE[14]	Name of GSD-Filename (not NCM PC)
min-slave-interval	WORD	from GSD-File (Unit: 100 usec)
slave-control	BYTE	see table below
Slv Header End		
PRM_DATA Start		
parameter-telegram length	WORD	inclusive this word
parameter telegram	BYTE[]	composed from GSD-File
PRM_DATA End		
Slave Module List Start		
slave-module list length	WORD	inclusive this word
output data length	BYTE	Number of Output-Bytes
input data length	BYTE	Number of Input-Bytes
Module count	BYTE	Number of modules
Module-entry	module-entry[]	module-data
Slave Module List End		
CFG_DATA Start		
config-data-len	WORD	inclusive this word
config-telegram	BYTE[]	composed from GSD-File
CFG_DATA End		
reserved	BYTE	0x02

reserved	BYTE	0x00

Slave control:



Module-entry:

Name	Type	Description
module-entry-length	WORD	inclusive this word (0x1E)
module-name	BYTE[24]	Name of module
output-offset	WORD	not used
input-offset	WORD	not used

5.4 FMS-Block (optional)

Name	Type	Description
Block-ID	BYTE	0x81
Block-length	DWORD	length of data-block (without header) (0x00000001)
data-block	BYTE[]	data is not relevant (0x00)

5.5 End-Block

Name	Type	Description
Block-ID	BYTE	0xFF
Block-length	DWORD	length of data-block (without header)
data-block	BYTE[]	data is not relevant

6 Glossary

AUTOCLEAR

1. Configuration property of a DP slave - the master changes to the AUTOCLEAR mode if this slave drops out.

2. This is the same as the DP_CLEAR mode of a DP master, when it changes to the mode due to the AUTOCLEAR property of a slave.

Bus parameter

Bus parameters control the data transmission on the bus - Each ⇒PROFIBUS ⇒node must use bus parameters that match the bus parameters of other nodes.

Bus segment

Part of a ⇒subnet. Subnets can be created using bus segments with connectivity devices such as repeaters and bridges. Segments are transparent for addressing.

COM PROFIBUS

Configuration tool for defining communications nodes and the bus parameters.

CP

Communications **P**rocessor - communications module/network card for installation in computers or programmable controllers.

CPU

Central **P**rocessing **U**nit - here processor of the PC

CPU load

Load on the CPU of the PC - here: resulting from DP communication

Data transmission rate

Transmission rate on the bus (unit: bps). A ⇒bus parameter for ⇒PROFIBUS. The data transmission rate used depends on various conditions such as distance.

DB

Here: **D**ata**b**ase - The local database describes the communications network from the view of the local system.

Deadlock

When more than one parallel process (here: threads) become blocked (A waits for B and B waits for A).

Distributed I/Os

An input/output module used in a distributed configuration. The connection between the programmable controller and the distributed I/Os is established via the ⇒PROFIBUS bus system. For the programmable controller, the distributed I/Os are no different from local process inputs/outputs.

DP

Distributed peripheral I/Os, communication protocol for PROFIBUS complying with EN 50 170 Volume 2.

DP Base

Name of the DP programming interface of the CP 5613/CP 5614, in contrast to the DP Lib interface of the CP 5412 (A2), CP 5611 and CP 5511.

DP I/O module

DP slaves are modular. A ⇐ DP slave has at least one DP I/O module.

DP I/O type

DP I/O type identifies a DP I/O module. The following types exist:

- input module
- output module
- input/output module

DP master

A ⇒node with master functionality in ⇒PROFIBUS DP - The DP master handles the exchange of data with the ⇒DP slaves assigned to it.

DP slave

A ⇒ node with slave functions in ⇒ PROFIBUS DP.

DP subnet

PROFIBUS subnet in which only ⇒ distributed peripheral I/Os are operated.

DP subsystem

A ⇒ DP master and all ⇒ DP slaves with which the DP master exchanges data.

DPC1

DP extended by acyclic read and write jobs and alarms between cyclic DP master and slave.

DPC2

DP extended by connection control and read and write jobs from a non-cyclic master.

DPRAM

Dual-**P**ort **R**andom **A**ccess **M**emory – allows simultaneous access to one memory area (RAM) by two computer units (CP and CPU).

DP-V1

DP extensions, this includes DPC1 and DPC2

Driver

Software that allows data exchange between application programs and the ⇒ CP.

Fast logic

Property of the CP 5613/CP 5614: an input value of a slave can be monitored. When it changes to a specified value, output data of another slave is set.

FDL

Fieldbus **D**ata **L**ink - Layer 2 for ⇒ PROFIBUS

Frame

Message from one PROFIBUS node to another.

Frame header

A frame header consists of an identifier for the ⇒ frame and the source and destination address.

Frame trailer

A frame trailer consists of a checksum and the end identifier of the ⇒ frame.

FREEZE mode

The FREEZE mode is a DP mode in which the process data are acquired at the same time from all (or a group) of DP slaves. The time at which the data are required is indicated in the FREEZE command (a synchronization control frame).

Gap update factor

A free address area between two active ⇒ nodes is checked cyclically by the node with the lower ⇒ PROFIBUS address to find out whether or not another station is requesting to enter the logical ring. The cycle time for this check is as follows:

gap update factor x target rotation time

Group identifier

DP slaves can be assigned to one or more groups using a group identifier. The ⇒ DP slaves can then be addressed by the group identifier when transferring control frames.

Highest PROFIBUS address

A ⇒ bus parameter for ⇒ PROFIBUS. This specifies the highest ⇒ PROFIBUS address of an active ⇒ node on the PROFIBUS. For passive nodes, PROFIBUS addresses higher than the HSA are permitted (possible values: HSA 1 to 126).

HSA

Highest **S**tation **A**ddress - one of the bus parameters. Specifies the highest node address used in the network.

ISO

International **S**tandards **O**rganization - International organization based in Geneva responsible for formulating general standards particularly in the area of data transmission.

LAN

Local **A**rea **N**etwork - local network for direct connection of computers.

LSB

Least **S**ignificant **B**it

Master

An active node on ⇒ PROFIBUS that can send frames on its own initiative when it is in possession of the token.

Maximum station delay

A ⇒ bus parameter for ⇒ PROFIBUS. The maximum station delay (max. TSDR) specifies the longest interval required by a ⇒ node in the subnet between receiving the last bit of an unacknowledged ⇒ frame and sending the first bit of the next frame. After sending an unacknowledged frame, a sender must wait for the max. TSDR to elapse before sending a further frame.

Minimum station delay

A ⇒ bus parameter for ⇒ PROFIBUS. The minimum station delay (min. TSDR) specifies the minimum time that the receiver of a ⇒ frame must wait before sending the acknowledgment or sending a new frame. The min. TSDR takes into account the longest interval required by a station in the subnet for receiving an acknowledgment after sending a frame.

MSAC_C1

"Master Slave Acyclic Class 1" - Name for DPC1 in the DP-V1 standard description.

MSAC_C2

"Master Slave Acyclic Class 2" - Name for DPC2 in the DP-V1 standard description.

MSB

Most **S**ignificant **B**it

MSCY_C1

"Master Slave cyclic class 1" - name of the normal DP master operation in the DP-V1 standard description.

Network

A network consists of one or more interconnected ⇒ subnets with any number of ⇒ nodes. Several networks can exist one beside the other. For each subnet, there is a common ⇒ node table.

Node

A node is identified by a ⇒ PROFIBUS address ⇒ in PROFIBUS.

PC

Personal **C**omputer

PG

Programming device (industrial PC) belonging to the SIMATIC product family from Siemens AG used for programming, configuring and in maintenance and service.

Process image The process image is a special memory area in the programmable controller. At the start of the cyclic program, the signal states of the input modules are transferred to the process input image. At the end of the cyclic program, the process output image is transferred as the signal state to the output modules.

PROFIBUS A fieldbus complying with EN 50 170 Vol. 2 (DIN 19245).

PROFIBUS address The PROFIBUS address is a unique identifier of a ⇒ node connected to a ⇒ PROFIBUS network. To address a node, the PROFIBUS address is transferred in the ⇒ frame.

PROFIBUS DP PROFIBUS distributed peripheral I/Os. Transmission services complying with PROFIBUS EN 50 170 Vol. 2 (DIN E 19245 Part 3) in ⇒ PROFIBUS.

PROFIBUS DP master A ⇒ node with master functions in PROFIBUS DP.

Protocol Rules governing the transmission of data - The rules specify not only the formats of the messages but also the data flow during data transmission.

Reorganization token ring All the ⇒ masters on ⇒ PROFIBUS form a logical token ring. Within this token ring, the token is passed on from station to station. If the transmission of the token is incorrect or if a master is removed from the ring, this leads to an error when the token is passed on (the token is not accepted by this station) and the station is excluded from the ring. The number of exclusions is counted in the internal Token_error_counter. If this counter reaches an upper limit value, the logical token ring is then reorganized.

S7 PLC Abbreviation for a programmable logic controller belonging to the SIMATIC product family from Siemens AG.

SCOPE PROFIBUS Diagnostic product for ⇒ PROFIBUS, with which the frames sent on the ⇒ network are detected and analyzed.

Segment Synonym for ⇒ bus segment

Semaphore Wait mechanism for synchronizing several programs, for example in Windows NT.

Services Services provided by a communication protocol.

Setup time

A ⇒ PROFIBUS ⇒ bus parameter - The setup time specifies the minimum time between receiving an acknowledgment and sending a new call.

Sign of life monitoring

A monitoring time that can be set on a ⇒DP slave to detect failure of the controlling ⇒DP master.

SIMATIC NET

Siemens Network and Communication - Product range for networks and network components from Siemens.

Slot time

A ⇒ PROFIBUS bus parameter - The slot time (TSL) is a monitoring time between a sender sending a ⇒ frame and receiving the acknowledgment of the receiver.

Subnet

A subnet is part of a ⇒ network whose ⇒ bus parameters (for example ⇒ PROFIBUS addresses) must match up. It includes the bus components and all the connected stations. Subnets can be interconnected, for example using ⇒ gateways, to form a network .
A ⇒ system consists of several subnets each with a unique ⇒ subnet number. A subnet consists of several ⇒ nodes with unique ⇒ PROFIBUS addresses.

Subnet number

A ⇒ system consists of several ⇒ subnets with unique subnet numbers.

SYNC mode

The SYNC mode is a DP mode in which several or all ⇒ DP slaves transfer data to their process outputs at a certain time. The time at which the data is transferred is indicated in the SYNC command (a control command for synchronization).

System

The entire electrical equipment belonging to a plant/system including programmable controllers, devices for operator monitoring and control, bus systems, field devices, drives, supply lines.

Target rotation time

A ⇒ PROFIBUS ⇒ bus parameter. The token is the right to send for a node on PROFIBUS. A node compares the actual token rotation time it has measured with the target rotation time and depending on the result can then send high or low priority frames.

Thread

A subprocess running parallel.

Watchdog

A mechanism for monitoring operability of nodes.