

SIMATIC NET

DPC31 Siemens PROFIBUS-DP Controller  
with C31 Core

Hardware Description

Date May, 19, 2004



# PROCENTEC

Turfschipper 41  
2292 JC Wateringen  
The Netherlands  
tel. +31 (0)174 671 800  
fax. +31 (0)174 671 801  
e-mail info@procentec.com

## Products

### ASICs

LSPM2 RoHS (6-pack)	<b>EUR 95,- per pack</b>
LSPM2 RoHS (66-pack)	<b>EUR 919,- per pack</b>
SPC3 Step-C RoHS (6-pack)	<b>EUR 75,- per pack</b>
SPC3 Step-C RoHS (96-pack)	<b>EUR 1052,- per pack</b>
SPC3 Step-C <b>Low Voltage</b> RoHS (5-pack)	<b>EUR 85,- per pack</b>
SPC3 Step-C <b>Low Voltage</b> RoHS (160pack)	<b>EUR 1600,- per pack</b>
SPC4-2 RoHS (5-pack)	<b>EUR 92,- per pack</b>
SPC4-2 RoHS (160-pack)	<b>EUR 2288,- per pack</b>
DPC31 Step-B RoHS (6-pack)	<b>EUR 95,- per pack</b>
DPC31 Step-B RoHS (60-pack)	<b>EUR 877,- per pack</b>
DPC31 Step-C1 RoHS (6-pack)	<b>EUR 95,- per pack</b>
DPC31 Step-C1 RoHS (60-pack)	<b>EUR 868,- per pack</b>
SIM1-1 RoHS (160-pack)	<b>EUR 4890,- per pack</b>
SIM1-2 RoHS (60-pack)	<b>EUR 900,- per pack</b>
ERTEC 200 (5-pack)	<b>EUR 120,- per pack</b>
ERTEC 200 (70-pack)	<b>EUR 950,- per pack</b>
ERTEC 400 (5-pack)	<b>EUR 250,- per pack</b>
ERTEC 400 (70-pack)	<b>EUR 2300,- per pack</b>

**Please contact us for larger quantities or check out: [www.procentec.com](http://www.procentec.com)**

# PROCENTEC

Turfschipper 41  
2292 JC Wateringen  
The Netherlands  
tel. +31 (0)174 671 800  
fax. +31 (0)174 671 801  
e-mail info@procentec.com

## Firmware and development kits

SPC3 ANSI-C software	<b>EUR 680,-</b>
DPC31 Development kit	<b>EUR 2470,-</b>
ERTEC 200 Development Kit	<b>EUR 4500,-</b>
ERTEC 200 Productline license	<b>EUR 4900,-</b>
ERTEC 400 Development Kit	<b>EUR 4500,-</b>
ERTEC 400 Productline license	<b>EUR 4900,-</b>

## Tools

ProfiScript - Certification software	<b>EUR 800,- (HW excl.)</b> <a href="http://www.procentec.com/profiscrypt">www.procentec.com/profiscrypt</a>
ProfiTrace - Analyzer on USB	<b>EUR 1400,-</b> <a href="http://www.procentec.com/profitrace2">www.procentec.com/profitrace2</a>
ProfiCaptain - DP Master	<b>EUR 500,- (HW excl.)</b> <a href="http://www.procentec.com/proficaptain">www.procentec.com/proficaptain</a>

## Training

PROFIBUS Product Development	<b>EUR 995,-</b> <a href="http://www.procentec.com/ppd">www.procentec.com/ppd</a>
Certified PROFIBUS Engineer	<b>EUR 1850,-</b> <a href="http://www.procentec.com/cpe">www.procentec.com/cpe</a>
Certified PROFIBUS Installer	<b>EUR 750,-</b> <a href="http://www.procentec.com/cpi">www.procentec.com/cpi</a>

**SIEMENS**

# **SIMATIC NET**

## **DPC31 Hardware Description of Step C**

Siemens PROFIBUS-DP Controller  
(according to IEC 61158) with  
Integrated C31 Core

Version: 2.1  
Date: 05/04

**Liability Exclusion**

We have tested the contents of this document regarding agreement with the hardware and software described. Nevertheless, deviations can't be ruled out, so that we are not guaranteeing complete agreement. However, the data in this document is checked periodically. Required corrections are included in subsequent editions. We gratefully accept suggestions for improvement.

**Copyright**

Copyright Siemens AG 2004 All Rights Reserved  
Unless permission has been expressly granted, passing on this document or copying it, or using and sharing its content is not allowed. Offenders will be held liable. All rights reserved, in the event a patent is granted, or a utility model or design is registered.

The trade marks SIMATIC, SINEC L2 are protected by law for Siemens through application/registration.  
All other product and system names are (registered) trade marks of their respective proprietors, and are to be treated as such.

Subject to technical change

**Overview of Versions**

<b>Version No.</b>	<b>Date</b>	<b>Page</b>	<b>Comment</b>
0.x			First version
1.0	Dec.14, 2000	div.	RS485 wiring corrected RXD_RXS and XCTS_RXA Data Buffer
1.1	Apr. 25, 2002	div.	Casing, wiring of external processor
1.2	Apr. 9, 2003	Chap. 12.2.2	Optimization of the bus cycle
2.0	Feb. 16, 2004	div.	Detailed description for DPC31 STEP C
2.1	May, 19, 2004	Chap. 13.3	Correction of the Order number

**Table of Contents**

<b>1</b>	<b>INTRODUCTION</b>	<b>7</b>
<b>2</b>	<b>OVERVIEW</b>	<b>8</b>
2.1	General Data	8
2.2	Differences Between the DPC31 and the SPC3/SPC4	8
2.3	Function Overview (Block Diagram)	9
2.4	Pin Description	10
<b>3</b>	<b>MEMORY ASSIGNMENT</b>	<b>12</b>
3.1	Memory Area Distribution in the DPC31	12
3.2	Control Unit Parameters (Latches/Registers)	12
3.3	Organizational Parameters (RAM)	15
<b>4</b>	<b>ASIC INTERFACE</b>	<b>23</b>
<b>5</b>	<b>COMMUNICATION FUNCTIONS OF THE SEQUENTIAL CONTROL SYSTEM</b>	<b>24</b>
5.1	Structure of SAP List	25
5.1.1	Configuration of Buffer	28
5.1.2	Request Interface for DPS and FDL (Instruction Queue)	30
5.1.3	Acknowledgement Interface for FDL and FMA (Indication_Queue)	32
5.2	DPS Module, Description of the Interface	33
5.2.1	Set_Slave_Address, SSA (SAP55)	35
5.2.2	Set_Param, Prm (SAP61)	36
5.2.3	Check_Config, Chk_Cfg (SAP62)	38
5.2.4	Slave_Diagnosis (SAP60)	38
5.2.5	Write_Read_Data (Default SAP)	40
5.2.5.1	DX-Entered-Interrupt	41
5.2.6	Global_Control (SAP58)	42
5.2.7	Read_Inputs (SAP56)	44
5.2.8	Read_Outputs (SAP57)	45
5.2.9	Get_Config (SAP59)	45
5.2.10	Data Exchange Broadcast (DXB)	45
5.3	Additional FDL Functions	48
5.3.1	FDL_Monitoring Timer	48
5.3.2	Time of Day Synchronization	49
5.3.2.1	Support of MSACSI SAPs	50

<b>6</b>	<b>USER FUNCTIONS ON THE C31 CONTROLLER</b>	<b>50</b>
<b>7</b>	<b>DESCRIPTION OF THE HARDWARE BLOCKS</b>	<b>50</b>
7.1	Universal Processor Interface	50
7.1.1	Bus Interface Unit (BIU)	51
7.1.2	IO Interface	55
7.1.3	Interface Signals	56
7.1.4	Interrupt Controller of the $\mu$ P Interface in the DPC31	56
7.2	Synchronous Serial Interface (SSC Interface)	59
7.3	80C31 Core and Interface	62
7.3.1	Reset Phase of the C31	63
7.3.1.1	Boot Type Setting	63
7.3.1.2	Boot Type 2	63
7.3.2	80C31 Core	64
7.3.2.1	Data Area	64
7.3.2.2	Code Area	66
7.3.3	Code RAM	66
7.3.4	Expansion Interface to the 80C31 Core	67
7.3.5	Interface Signals	67
7.4	C31 Interrupt Controller in the DPC31	68
7.5	Serial PROFIBUS Interface	68
7.5.1	Asynchronous Physics Unit (NRZ)	68
7.5.1.1	Transmitter	68
7.5.1.2	Receiver	68
7.5.1.3	Interface Signals	69
7.5.2	Synchronous Physics Unit (Manchester)	69
7.5.2.1	Transmitter	69
7.5.2.2	Receiver	71
7.5.2.3	Power-Saving Serial Interface	72
7.5.2.4	Interface Signals	78
7.6	DPS Watchdog Timer	78
7.7	Watchdog Timer	78
7.7.1	Automatic Baud Rate Detection	78
7.7.2	Baud rate Monitoring	78
7.7.3	Response Monitoring	79
7.8	Clock Supply	79
7.8.1	PLL	79
<b>8</b>	<b>TEST SUPPORT</b>	<b>81</b>
8.1	Emulator Connection for the C31	81
<b>9</b>	<b>ELECTRICAL SPECIFICATIONS</b>	<b>83</b>
9.1	Maximum Limits	83
9.2	Permitted Operating Values	83
9.3	Guaranteed Operating Range for the Specified Parameters	84

9.4	Power Loss	84
9.4.1	Power-Up of the Supply Voltage	84
9.4.2	Structure of the Pad Cells with 5V Tolerance	85
9.4.3	DC Specification of the Pad Cells	87
9.5	AC Specification	88
9.5.1	Driver Capability	88
9.5.2	Timing Diagrams, Signal Run Times	88
9.5.2.1	Clock Supply (XPLEN = '1')	89
9.5.2.2	Clock Outputs	89
9.5.2.3	Reset	89
9.5.2.4	Interrupt	89
9.5.2.5	µP Interface	90
9.5.2.5.1	Synchronous Intel Mode (80C32)	90
9.5.2.5.2	Asynchronous Intel Mode (X86 Mode)	92
9.5.2.5.3	Synchronous Motorola Mode (E_Clock mode; for example, 68HC11)	93
9.5.2.5.4	Asynchronous Motorola Mode (for example, 68HC16)	95
9.5.2.6	C31 Memory Interface (internal C31 on external memory)	97
9.5.2.7	Emulator-Interface	99
9.5.2.8	SSC Interface (SPI)	102
<b>10</b>	<b>MECHANICAL SPECIFICATION</b>	<b>103</b>
10.1	PQFP 100 Casing	103
<b>11</b>	<b>DPC31 PINOUT</b>	<b>106</b>
<b>12</b>	<b>APPLICATION NOTES</b>	<b>107</b>
12.1	DPC31 Wiring	107
12.2	PROFIBUS Interface	107
12.2.1	Pin Assignment	107
12.2.2	Wiring Example RS485 Interface	109
<b>13</b>	<b>APPENDIX</b>	<b>110</b>
13.1	Addresses	110
13.2	General Definitions of Terms	110
13.3	Order Numbers	111

## 1 Introduction

Siemens offers to its users several ASICs for data exchange between automation devices that, on the basis of IEC 61158, support or completely process the data traffic between the individual automation stations.

To support intelligent master/slave solutions, that is, implementations with a microprocessor, the following ASICs are available. All ASICs do the following: support the transmission rates of 9.6 kBits/s ... 12000 kbit/s, autonomously set themselves to the transmission rate specified by the master and monitor it. After these ASICs receive a correct message, they autonomously generate the requested response messages.

In the **ASPC2** (Advanced Siemens PROFIBUS Controller), many components of Layer2 of the OSI model are already integrated according to ISO, but it still needs the support of a processor. This ASIC supports baud rates up to 12000 kbit/s; however, in its complexity, it is conceived more for master applications.

The **SPC3** (Siemens PROFIBUS Controller), through the integration of the complete PROFIBUS DP slave protocol, considerably relieves the processor of an intelligent PROFIBUS slave.

However, in the field of automation, there are also simple devices such as switches, thermoelements, etc. that do not require a microprocessor for recording their states.

For a low cost adaptation of such devices, two additional ASICs are available: the **SPM2** (Siemens PROFIBUS Multiplexer, Version 2) and **LSPM2** (Lean Siemens PROFIBUS Multiplexer). These chips process as DP slaves in the bus system.

The LSPM2 has the same functions as the SPM2 but with a lower number of I/O and diagnostic ports.

The **DPC31** (DP Controller with integrated 8031 core) is a highly integrated PROFIBUS slave ASIC. The DPC31 is a slave controller for both PROFIBUS DP/DPV1 and PA applications.

The uses of this chip cover a wide area. On the one hand, it can be used for simple, intelligent applications that make do with the integrated C31 core.

On the other hand, it can be used for high performance slave solutions that have increased communication requirements. This requirement is met with an internal RAM that has been increased to 6kByte.

Approximately 5.5kByte of communication memory is available to the user.

### The DPC31 has the following main features:

- integrated standard C31 core with an additional 3<sup>rd</sup> timer (Timer 2)
- low processor load through the integration of the complete DP slave protocol
- simple processor interface for a large number of processors:

INTEL:	8032, 80x86
Siemens:	C166
Motorola:	HC11, HC16, HC916
- SSC interface (SPI) for interfacing serial EEPROMs, A/D converters, etc.
- integration of synchronous as well as asynchronous bus physics

This document explains the hardware configuration and the wiring of the DPC31.

In addition, Siemens offers a separate software package that relieves the user of local H/W register manipulations and memory calculations. The package provides a convenient C-interface for interfacing Profibus communication with the slave process.

## 2 Overview

### 2.1 General Data

Package:	100 Pin PQFP
Baud rate:	Asynchronous: 9.6, 19.2, 45.45, 93.75, 187.5, 500 kBd, 1.5, 3, 6 & 12 MBd Synchronous: 31.25 kBd
Bus Interface:	8-Bit asynchronous/synchronous Intel and Motorola interface
C31 Ports:	Standard Port Interface (4 Ports) for external memory expansion and emulator interface
SSC Interface:	Synchronous serial interface (SPI) for connecting serial E <sup>2</sup> PROMs, A/D converters, etc.
Memory Area	6 kByte (approx. 5.5 kByte utilizable) can be directly addressed and can be broken down into data and code memory
Environmental Cond.:	3.3V ±10%; -40 to +85 °C

### 2.2 Differences Between the DPC31 and the SPC3/SPC4

Characteristics	DPC31	SPC3	SPC4
<b>General:</b>			
Package	100 Pin PQFP	44 Pin PQFP	44 Pin PQFP
External µP Interface	parallel, 8 bits	parallel, 8 bits	parallel, 8 bits
Family	Siemens, Intel, Motorola	Siemens, Intel, Motorola	Siemens, Intel, Motorola
Preprocessing	yes, via int. C31	no	no
External Memory Expansion (C31)	yes, Flash, RAM etc.	no	no
SSC Interface (SPI)	yes, for example E <sup>2</sup> PROM up to 64 kByte, A/D conv. (AD7714)	no	no
I/O Interface	yes, up to 40 bits	no	no
Internal PLL	yes, input 12 MHz	no	no
Communication RAM	max. 5.5 kByte	1.4 kByte	1.14 kByte (1.64 for SPC41)
<b>PB Communication:</b>			
Baud rate			
async. RS485	9.6 kBd to 12 MBd	9.6 kBd to 12 MBd	9.6 kBd to 12 MBd
sync. Manchester	31.25kBd	no	31.25 kBd
DP Slave	fully integrated	fully integrated	partially integrated
Receive Resources	exchange buffer	exchange buffer	polling list
<b>Integrated User Functions:</b>			
E <sup>2</sup> PROM Read/Write	No	no	no
DPV1 Protocol	No	no	no

Table 2.2-1: Differences with respect to SPC3 and SPC4

### 2.3 Function Overview (Block Diagram)

Figure 2.3-1 shows the block diagram of the DPC31. The DPC31 has a **bus interface** for connecting an external microprocessor. It is a parameterizable, synchronous/asynchronous 8-bit interface for various Siemens, Intel, and Motorola micro-controllers/processors. Via the 13-bit address bus, the user can directly access the internal 5.5k RAM or the register cells. If the application does not need an external processor, the ports of the bus interface can be used as I/O. This makes 27 I/O bits available that the internal C31 can address individually.

The sequence control enters various events (for example, indication events, error events, etc.) in the **interrupt controller** that are signaled to the slave firmware via the interrupt pin. These events can be enabled individually via a mask register. Acknowledgement is made via the acknowledge register.

The **SSC interface (SPI)** is used for connecting a serial E<sup>2</sup>PROM or an A/D converter (such as AD7714). This interface is laid out only as a master interface. The user must create an appropriate program for the C31 for control of this interface

The **C31 interface** includes the ports of the standard controller. Via this interface, an external memory- and I/O expansion can be implemented. Via corresponding CS signals, the code and data address areas are coded out that are not used internally. In addition, up to 13 bits of I/O can be connected via these ports. The C31/32 emulator (Hitex etc.) is also controlled via this interface.

Via the **register cells**, the following are accessed: internal registers, the DPS(DP Slave) control units and the SSC module. The DPS control units represent the user interface to the DPS layer that is implemented via individual buffers. These control units exchange the buffers.

The integrated **C31** is fully compatible with the standard microcontroller. Also integrated is a **256 byte data RAM**. Via a second **interrupt controller**, the interrupt events mentioned above can also be entered in the C31. This makes it possible to distribute interrupt events between an external and an internal application.

The **bus physics unit** includes the asynchronous Layer1 (RS485: 9.6kBd to 12 MBd) and the synchronous Layer1 (IEC 1158-2; Manchester encoded: 31.25kBd), which also allows the chip to be operated in an intrinsically safe environment.

In the **clock unit**, an analog **PLL** is integrated, to which an external 12MHz quartz must be connected. With it, the PLL generates the internal 48MHz clock pulse for the asynchronous mode. In the synchronous mode, the PLL is switched off and an external clock pulse of 2 to 16 MHz is applied. In addition, power management is implemented in the clock unit that switches off internal clock pulses in certain states. As outputs, the internal working clock pulse divided by 2 and by 4 is available.

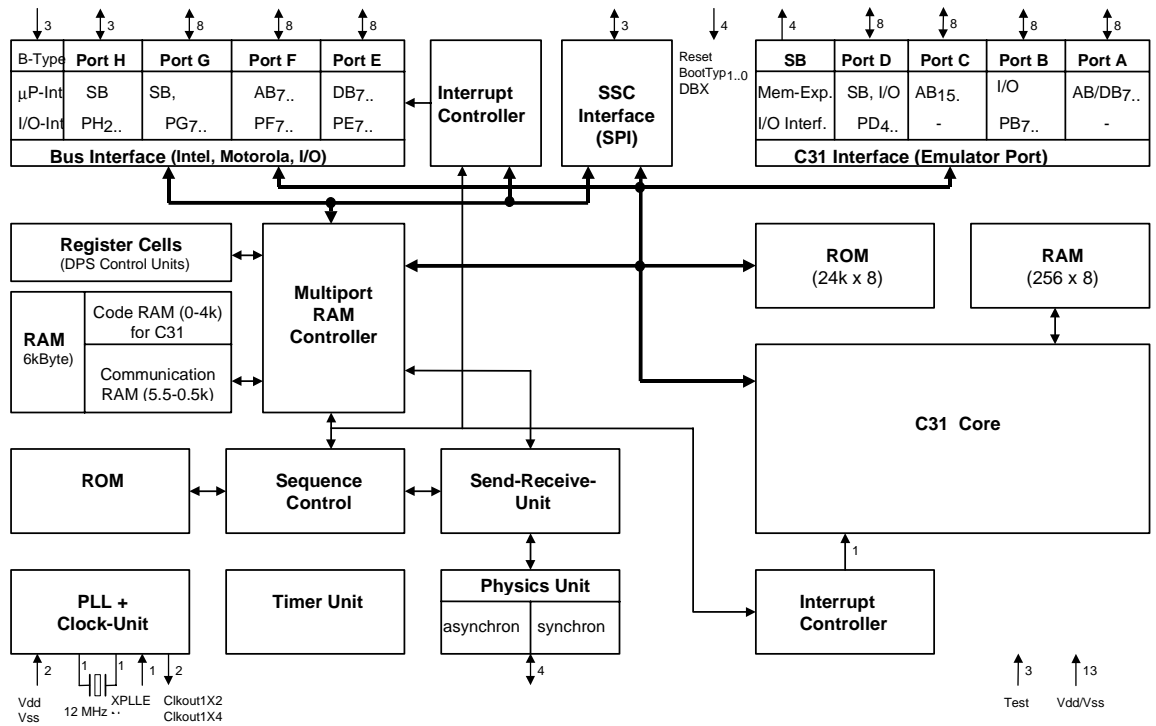


Figure 2.3-1: Block Diagram DPC31

## 2.4 Pin Description

The DPC31 has a 100 pin PQFP package with the following signals:

Function Group	Name	Pins	Type	Voltage Proof	Function
C31 Interface	PA	8	I/O	5V	Corresponds to P0 for the discrete type
	PB	8	I/O	5V	Corresponds to P1 for the discrete type
	PC	8	I/O	5V	Corresponds to P2 for the discrete type
	PD	8	I/O	5V	Corresponds to P3 for the discrete type
	ALE	1	I/O	5V	Address Latch Enable
	XPSSEN	1	I/O	5V	For emulation only
	XCSDATA	1	O	5V	Chip select for external RAM
	XCSCODE	1	O	5V	Chip select for external ROM
	BOOTTYP	2	I	5V	Type for loading the user program
	DBX	1	I	5V	Switch to In Circuit Emulator
µP Interface	PE	8	I/O	5V	
	PF	8	I/O	5V	
	PG	8	I/O	5V	
	PH	3	I/O	5V	
	BUSTYP	3	I	5V	
SSC Interface	SSCLK	1	O	5V	Connection for SPI Chips, Clock
	SSDO	1	O	5V	Connection for SPI Chips, Data_Out
	SSDI	1	I	5V	Connection for SPI Chips, Data_In
PLL + Clock Unit	XTAL1_CLK	1	I	3.3V	Quartz connection / Clock supply
	XTAL2	1	O	3.3V	Quartz connection
	AVDD	1			Separate V <sub>DD</sub> supply for PLL
	AGND	1			Separate GND supply for PLL
	XPLLEN	1	I	5V	Switching off the PLL and supply clock pulse via XTAL1_CLK
	CLKOUT1X2	1	O	5V	Clock pulse output CLK/2 (without reset)
CLKOUT1X4	1	O	5V	Clock pulse output CLK/4 (without reset)	
Physics Unit	RTS_TXE	1	O	3.3V	
	TXD_TXS	1	O	3.3V	
	XCTS_RXA	1	I	5V	
	RXD_RXS	1	I	5V	
General	RESET	1	I	5V	Reset Input
Test	NTEST1	1	I	5V	Test Pin
	NTEST2	1	I	5V	Test Pin
	TST1	1	I	5V	Test Pin
Supply	VDD	4			+3.3V
	GND	9			0V
Total		100			

**Figure 2.4-1:** DPC31 Pin List

Because of the 5V-tolerant I/O, and in order to ensure the least possible power loss, no pull-up or pull-down resistors are integrated in the pad cells; that is, all unused inputs or all output ports (since these are switched as input after reset) are to be applied to one defined level (Ports A, B, D, E, F, G, and H). This is not necessary for Port C since it is permanently configured as output. A bus contention is permitted for a maximum of 20ns.

### 3 Memory Assignment

#### 3.1 Memory Area Distribution in the DPC31

Table 3.1-1 shows the distribution of the internal 8k address space of the DPC31. Via this address space, the user interface to communication (DPS) is mapped. It does not matter whether the user program is running internally on the C31 or on the external microprocessor; the interface is identical in both cases.

The address area is subdivided into a 2K address space for the register cells and a 6k address space for the internal RAM. The internal registers (interrupt controller, Mode Register1, DPS control units, SSC interface) are located in the register area. Certain registers can only be read or written.

The RAM starts at address 800h. In the first area, the internal work cells are located (bit array, variables). The user is not to access this area. The sequential control system uses these cells for processing the protocol. Starting with address 0840h, the organizational parameters (parameter cells, buffer ptr(pointer) are located in the RAM. In the parameter cells, general parameter assignment data is transferred (Param Register, station address, Ident No., etc.), or status displays are stored (status register, GC\_Command, Score\_Register, etc.). The buffer pointers describe the entire buffer management for the SAPs. At address 08A0h, the buffers generated by the user start, corresponding to the parameter assignment of the organizational parameters. The sequence of the buffers can be selected as required. All buffers or liDXB must be located on segment addresses (32 bytes segmentation).

1FFFh		<b>Code Area for the Internal C31</b>	
	<b>Communication Area</b>	<b>Buffer Area</b>	
08A0h		<b>Organizational Parameters</b>	
0840h		<b>Internal Work Area</b>	
0800h	<b>RAM</b>	<b>SSC-Interface Control Unit Parameters Latches/Registers</b>	
0000h	<b>Register</b>		

**Table 3.1-1: Memory Area Distribution in the Internal RAM of the DPC31**

The stack for the sequential control system needs 64 bytes. A buffer for temporarily storing the receive message requires 32 bytes.

#### 3.2 Control Unit Parameters (Latches/Registers)

The register cells that are, for example, in the interrupt controller and the DPS control units, are located in the address area of 0000-0044h (XDATA). These cells can either be read or written only. The address assignments are shown in Table 3.2-1. When writing the register cells, the unassigned bit positions are 'don't care'.

Address	Name	Meaning (read access!)
0000h	Int-Req-Reg <sub>7..0</sub>	Interrupt Controller Register
0001h	Int-Req-Reg <sub>15..8</sub>	
0002h	Int-Req-Reg <sub>23..16</sub>	
0003h	Int-Req-Reg <sub>28..24</sub>	
0004h	Int-Reg <sub>7..0</sub>	
0005h	Int-Reg <sub>15..8</sub>	
0006h	Int-Reg <sub>23..16</sub>	
0007h	Int-Reg <sub>28..24</sub>	
0008h	Reserved	
...		
000Fh		
0010h	C31_Control Register <sub>7..0</sub>	Refer to Chapter 4
0011h	Reserved	
...		
0013h		
0014h	Delay-Timer <sub>7..0</sub>	Content of delay timer
0015h	Delay-Timer <sub>15..8</sub>	
0016h	Reserved	
...		
001Fh		
0020h	SSC_Rcv-Buf <sub>7..0</sub>	Receive buffer of the SSC interface
0021h	SSC_Sta-Reg <sub>3..0</sub>	Status register of the SSC interface
0022h	SSC_Ctrl1-Reg <sub>7..0</sub>	Control register of the SSC interface
0023h	SSC_Ctrl2-Reg <sub>2..0</sub>	Control register of the SSC interface
0024h	Reserved	
...		
002Fh		
0030h	User_SSA_Ok Cmd <sub>1..0</sub>	The user acknowledges the user SSA data of an SSA message positively
0031h	User_Prm_Ok Cmd <sub>1..0</sub>	The user acknowledges the user parameter assignment data of a prm message positively
0032h	User_Prm_Not_Ok Cmd <sub>1..0</sub>	The user acknowledges the user parameter assignment data of a prm message negatively
0033h	User_Ext_Prm_Ok Cmd <sub>1..0</sub>	The user acknowledges the ext. user parameter assignment data of a prm message positively
0034h	User_Ext_Prm_Not_Ok Cmd <sub>1..0</sub>	The user acknowledges the ext. user parameter assignment data of a prm message negatively
0035h	User_Cfg_Ok Cmd <sub>1..0</sub>	The user acknowledges the configuring data of a Cfg message positively
0036h	User_Cfg_Not_Ok-Cmd <sub>1..0</sub>	The user acknowledges the configuring data of a Cfg message negatively
0037h	User_Diag_Read-Cmd	The user makes a new diag buffer available
0038h	User_Get_Cfg_Read-Cmd	The user makes a new Get_Cfg buffer available
0039h	User_New_Din-Cmd <sub>1..0</sub>	The user makes a new Din buffer available
003Ah	User_Din_Buffer-State <sub>7..0</sub>	The user reads the current Din buffer assignment
003Bh	User_New_Dout-Cmd <sub>3..0</sub>	The user fetches the last Dout buffer from the N state
003Ch	User_Dout_Buffer-State <sub>7..0</sub>	The user reads the current Dout buffer assignment
003Dh	User_New_DXB0_Cmd <sub>2..0</sub>	The user fetches the last DXB0 buffer from the N state
003Eh	User_New_DXB1_Cmd <sub>2..0</sub>	The user fetches the last DXB1 buffer from the N state
003Fh	User_New_DXB2_Cmd <sub>2..0</sub>	The user fetches the last DXB2 buffer from the N state
0040h	User_New_DXB3_Cmd <sub>2..0</sub>	The user fetches the last DXB3 buffer from the N state
0041h	User_New_DXB4_Cmd <sub>2..0</sub>	The user fetches the last DXB4 buffer from the N state
0042h	User_New_DXB5_Cmd <sub>2..0</sub>	The user fetches the last DXB5 buffer from the N state

0043h	User_New_DXB6_Cmd2..0	The user fetches the last DXB6 buffer from the N state
0044h	User_New_DXB7_Cmd2..0	The user fetches the last DXB7 buffer from the N state
0045h	Reserved	
...		
07FFh		

**Table 3.2-1:** Assignment of the Internal Register Cells for READ

0000h	Int-Req-Reg <sub>7..0</sub>	Interrupt Controller Register
0001h	Int-Req-Reg <sub>15..8</sub>	
0002h	Int-Req-Reg <sub>23..16</sub>	
0003h	Int-Req-Reg <sub>28..24</sub>	
0004h	Int-Ack-Reg <sub>7..0</sub>	
0005h	Int-Ack-Reg <sub>15..8</sub>	
0006h	Int-Ack-Reg <sub>23..16</sub>	
0007h	Int-Ack-Reg <sub>28..24</sub>	
0008h	Int-Mask-Reg <sub>7..0</sub>	
0009h	Int-Mask-Reg <sub>15..8</sub>	
000Ah	Int-Mask-Reg <sub>23..16</sub>	
000Bh	Int-Mask-Reg <sub>28..24</sub>	
000Ch	Int-EOI-Reg <sub>0</sub>	
000Dh	reserved	
000Eh		
000Fh		
0010h	C31_Ctrl-Reg <sub>6..0</sub>	Refer to Chapter 6
0011h	Mode-Reg1-Set <sub>7..0</sub>	Refer to Chapter 6
0012h	Mode-Reg1-Reset <sub>7..0</sub>	Refer to Chapter 6
0013h	User_InstQ_Write-Cmd <sub>7..0</sub>	Transfers a new request to the sequential control system
0014h	reserved	
...		
001Fh		
0020h	SSC_Transmit-Buf <sub>7..0</sub>	Receive buffer of the SSC interface
0021h	SSC_Sta-Reg <sub>7..0</sub>	Status register of the SSC interface
0022h	SSC_Ctrl1-Reg <sub>7..0</sub>	Control register of the SSC interface
0023h	SSC_Ctrl2-Reg <sub>2..0</sub>	Control register of the SSC interface
0024h	SSC_Int_Enable-Reg <sub>3..0</sub>	Interrupt_Enable register of the SSC interface
0025h	SSC_Baudrate-Reg <sub>7..0</sub>	Baud rate register of the SSC interface
0026h	reserved	
...		
07FFh		

**Table 3.2-1:** Assignment of the Internal Register Cells for WRITE

### 3.3 Organizational Parameters (RAM)

The organizational parameters are stored by the user in the RAM under the addresses specified in the table below. These parameters primarily describe the parameter cells and the buffer pointers of the communication profile (buffer management).

0800h	reserved	Internal working area
...		
0827h		
0828h	Master_Add <sub>7..0</sub>	Address of parameter assignment mask
0829h	reserved	Internal working area
...		
0833h		
0834h	U_Buffer_State	Status of user buffers, if "Disable C Buffer" is in parameter register
0835h	reserved	Internal working area
...		
083Eh		
083Fh	Param_Ext_1-Register <sub>7..0</sub>	see below
0840h	Status-Register <sub>7..0</sub>	see below
0841h	Status-Register <sub>15..8</sub>	see below
0842h	Param-Register <sub>7..0</sub>	see below
0843h	Param-Register <sub>15..8</sub>	see below
0844h	Param-Register <sub>23..16</sub>	see below
0845h	Param-Register <sub>31..24</sub>	see below
0846h	TS_Adr_Register <sub>6..0</sub>	Profibus station Address of the DPC31 (this slave)
0847h	Real_No_Add_Change <sub>7..0</sub>	This parameter indicates whether the DP slave address may be changed at a later time. After reset, the slave firmware must set this parameter if it permits the Set_Slave_Address SAP. 0 = Address may be changed Otherwise = Address may not be changed If the DPC31 then receives a Set_Slave_Address message, it enters the current value here.
0848h	WD_Baud_Control_Val <sub>7..0</sub>	The root value for baud rate monitoring is parameterized.
0849h	Interframe_GAP_Time <sub>5..0</sub>	The Interframe GAP time (4...32 bits) is to be parameterized here for synchronous bus physics.
084Ah	DPS_User_Wd_Val <sub>7..0</sub>	In the DPS_Mode, the user is monitored with an internal 16-bit watchdog timer. The timer is decremented every 10 ms and must be reset by the user cyclically to the start value 'DPS_User_WD_Value <sub>15..0</sub> '. Resetting, enabling, and disabling the timer is initiated with 'DPS_User-Wd' request in the Instruction_Queue.
084Bh	DPS_User_Wd_Val <sub>15..8</sub>	
084Ch	Score_Register <sub>7..0</sub>	Specification for the next WD cycle for monitoring the configured slave-to-slave connections
084Dh	GC_Command <sub>7..0</sub>	GC command last received
084Eh	Ident_Low <sub>7..0</sub>	PNO Ident Number Low
084Fh	Ident_High <sub>7..0</sub>	PNO Ident Number High
0850h	DXB_Sync <sub>7..0</sub>	Specify links that are subject to Sync control
0851h	Group Clock	Specify group under which the clock pulse beater is triggered
0852h	Temp_Base-Ptr <sub>7..0</sub>	Segment address that points to the start of the temp buffer of the S/R unit
0853h	SAP_List_Base-Ptr <sub>7..0</sub>	Segment address that points to the start of the SAP list
0854h	SAP_SCB_Base-Ptr <sub>7..0</sub>	Segment address that points to the start of the SAP SCB (system control block)
0855h	DPS_SAP_Start-Ptr <sub>7..0</sub>	Offset address that points to the start of the DPS SAPs being established by the DPC31 in the SAP SCB
0856h	InstQ_Base-Ptr <sub>7..0</sub>	(segment pointer to the instruction queue)
0857h	InstQ_Length <sub>7..0</sub>	(length of the instruction queue in bytes (multiple of the length of an entry -> n*5))
0858h	InstQ_Read-Ptr <sub>7..0</sub>	Byte offset to the next entry to be read
0859h	InstQ_Write-Ptr <sub>7..0</sub>	Byte offset to the next free entry

085Ah	IndQ_Base-Ptr <sub>7..0</sub>	(segment pointer to the indication queue)
085Bh	IndQ_Length <sub>7..0</sub>	(length of the indication queue in bytes (multiple of the length of an entry - > n*3))
085Ch	IndQ_Read-Ptr <sub>7..0</sub>	Byte offset to the next entry to be read
085Dh	IndQ_Write-Ptr <sub>7..0</sub>	Byte offset to the next free entry
085Eh	Dout_Buffer-Length <sub>7..0</sub>	Length of the 4 Dout buffers
085Fh	Dout_Buffer1-Ptr <sub>7..0</sub>	Segment Pointer to Dout Buffer1
0860h	Dout_Buffer2-Ptr <sub>7..0</sub>	Segment Pointer to Dout Buffer2
0861h	Dout_Buffer3-Ptr <sub>7..0</sub>	Segment Pointer to Dout Buffer3
0862h	Dout_Buffer4-Ptr <sub>7..0</sub>	Segment Pointer to Dout Buffer4
0863h	Din_Buffer-Length <sub>7..0</sub>	Length of the 3 Din buffers
0864h	Din_Buffer1-Ptr <sub>7..0</sub>	Segment Pointer to Din Buffer1
0865h	Din_Buffer2-Ptr <sub>7..0</sub>	Segment Pointer to Din Buffer2
0866h	Din_Buffer3-Ptr <sub>7..0</sub>	Segment Pointer to Din Buffer3
0867h	User_SSA_Buffer-Ptr <sub>7..0</sub>	Segment Pointer to User SSA Buffer
0868h	MAC_SSA_Buffer-Ptr <sub>7..0</sub>	Segment Pointer to Mac SSA Buffer
0869h	User_Prm_Buffer-Ptr <sub>7..0</sub>	Segment Pointer to User Prm Buffer
086Ah	MAC_Prm_Buffer-Ptr <sub>7..0</sub>	Segment Pointer to Mac-Prm Buffer
086Bh	User_DDB_Prm_Buffer-Ptr <sub>7..0</sub>	Segment Pointer to User DDB Prm Buffer
086Ch	MAC_DDB_Prm_Buffer-Ptr <sub>7..0</sub>	Segment Pointer to Mac DDB Prm Buffer
086Dh	User_Cfg_Buffer-Ptr <sub>7..0</sub>	Segment Pointer to User Cfg Buffer
086Eh	MAC_Cfg_Buffer-Ptr <sub>7..0</sub>	Segment Pointer to Mac Cfg Buffer
086Fh	User_Diag_Reply_Buffer-Ptr <sub>7..0</sub>	Segment Pointer to User-Diag Buffer
0870h	MAC_Diag_Reply_Buffer-Ptr <sub>7..0</sub>	Segment Pointer to Mac Diag Buffer
0871h	User_GCfg_Reply_Buffer-Ptr <sub>7..0</sub>	Segment Pointer to User-Get-Cfg Buffer
0872h	MAC_GCfg_Reply_Buffer-Ptr <sub>7..0</sub>	Segment Pointer to Mac-Get-Cfg Buffer
0873h	MAC_GC_Buffer-Ptr <sub>7..0</sub>	Segment Pointer to Global Ctrl Buffer
0874h	DXB0_Buffer1-Ptr <sub>7..0</sub>	Segment Pointer to DXB0 Buffer1
0875h	DXB0_Buffer2-Ptr <sub>7..0</sub>	Segment Pointer to DXB0 Buffer2
0876h	DXB0_Buffer3-Ptr <sub>7..0</sub>	Segment Pointer to DXB0 Buffer3
0877h	DXB1_Buffer1-Ptr <sub>7..0</sub>	Segment Pointer to DXB1 Buffer1
0878h	DXB1_Buffer2-Ptr <sub>7..0</sub>	Segment Pointer to DXB1 Buffer2
0879h	DXB1_Buffer3-Ptr <sub>7..0</sub>	Segment Pointer to DXB1 Buffer3
087Ah	DXB2_Buffer1-Ptr <sub>7..0</sub>	Segment Pointer to DXB2 Buffer1
087Bh	DXB2_Buffer2-Ptr <sub>7..0</sub>	Segment Pointer to DXB2 Buffer2
087Ch	DXB2_Buffer3-Ptr <sub>7..0</sub>	Segment Pointer to DXB2 Buffer3
087Dh	DXB3_Buffer1-Ptr <sub>7..0</sub>	Segment Pointer to DXB3 Buffer1
087Eh	DXB3_Buffer2-Ptr <sub>7..0</sub>	Segment Pointer to DXB3 Buffer2
087Fh	DXB3_Buffer3-Ptr <sub>7..0</sub>	Segment Pointer to DXB3 Buffer3
0880h	DXB4_Buffer1-Ptr <sub>7..0</sub>	Segment Pointer to DXB4 Buffer1
0881h	DXB4_Buffer2-Ptr <sub>7..0</sub>	Segment Pointer to DXB4 Buffer2
0882h	DXB4_Buffer3-Ptr <sub>7..0</sub>	Segment Pointer to DXB4 Buffer3
0883h	DXB5_Buffer1-Ptr <sub>7..0</sub>	Segment Pointer to DXB5 Buffer1
0884h	DXB5_Buffer2-Ptr <sub>7..0</sub>	Segment Pointer to DXB5 Buffer2
0885h	DXB5_Buffer3-Ptr <sub>7..0</sub>	Segment Pointer to DXB5 Buffer3
0886h	DXB6_Buffer1-Ptr <sub>7..0</sub>	Segment Pointer to DXB6 Buffer1
0887h	DXB6_Buffer2-Ptr <sub>7..0</sub>	Segment Pointer to DXB6 Buffer2
0888h	DXB6_Buffer3-Ptr <sub>7..0</sub>	Segment Pointer to DXB6 Buffer3

0889h	DXB7_Buffer1-Ptr <sub>7..0</sub>	Segment Pointer to DXB7 Buffer1
088Ah	DXB7_Buffer2-Ptr <sub>7..0</sub>	Segment Pointer to DXB7 Buffer2
088Bh	DXB7_Buffer3-Ptr <sub>7..0</sub>	Segment Pointer to DXB7 Buffer3
088Ch	reserved	Preset with 0x00
...		
0896h	Buffer Area	
0897h		
...		
1FFFh		

**Table 3.3-2:** Assignment of the Organizational Parameters

**Meaning of the Register Cells:**

**Status Register:**

WD-State <sub>1..0</sub>		DPS-State <sub>1..0</sub>		0	Diag_Flag	0	MAC State
1	0	1	0				
7		4			2		0

DPC31 Release <sub>3..0</sub>				Baudrate <sub>3..0</sub>			
3	2	1	0	3	2	1	0
15				8			

The status register displays the current MAC status, the DPS status, and the watchdog timer status. In addition, the baud rate that was found, and the release number of the DPC31 is also entered.

- MAC State:           The state of the MAC  
 =0 The MAC is in the 'Offline' state  
 =1 The MAC is in 'Passive Idle'
  
- Diag\_Flag:           State Diagnostic Buffer  
 =0 The diagnostic buffer was fetched by the master (if Diag.Stat\_Diag=0).  
 =1 The diagnostic buffer was not fetched by the master.
  
- DPS-State<sub>1..0</sub>:      The state of the DPS State Machine  
 =00 State 'Wait\_Prm'  
 =01 State 'Wait\_Cfg'  
 =10 State 'Data\_Exchange'
  
- WD-State<sub>1..0</sub>:       The state of the Watchdog SM  
 =00 State 'Baud\_Search'  
 =01 State 'Baud\_Control'  
 =10 State 'DP\_Control'
  
- Baudrate<sub>3..0</sub>:       The baud rate found by the DPC31  
 =0000 12 MBd (asyn.)  
 =0001 6 MBd (asyn.)  
 =0010 3 MBd (asyn.)  
 =0011 1.5 MBd (asyn.), 31.25 kBd (syn.)  
 =0100 500 kBd (asyn.)  
 =0101 187.5 kBd (asyn.)  
 =0110 93.75 kBd (asyn.)  
 =0111 45.45 kBd (asyn.)  
 =1000 19.2 kBd (asyn.)  
 =1001 9.6 kBd (asyn.)
  
- DPC31-Release<sub>3..0</sub>: Release number of the DPC31: The release number consists of two groups.  
 DPC31-Release<sub>1..0</sub>: numbers the compatible versions  
 DPC31-Release<sub>3..2</sub>: is the index within a compatible version.

Bit (3:0)	Meaning
0000	DPC31 Step A
0001	DPC31 Step B
0010	DPC31 Step C
Rest	not possible so far

**Param Register:**

In the Param Register, individual parameter bits are transferred that are to be changed only in the MAC state 'Offline', however. When the request 'MAC\_Start' (refer to Chapter 5.1.2) is executed, these parameters are distributed by the sequential control system to the individual modules. Subsequent changes are not taken into account.

En Clock Sync 7	Early_Ready	EOI_Timebase	Quick_Sync_New	GIM_EN	XRTS/ADD	0	0	0
Check_No_GC_Reserved 15	Check No Prm Reserved	New_GC_Int_Mode	XDP/FDL MinTSDR	Failsafe Mode	Freeze_Supported	Sync_Supported	DP_Mode	8
0	Check DP V1 Enable	SD4_WD_Reset	En_Change_Cfg_Buffer	XAsyn/Syn	Syn Baud	Syn_Clkin <sub>1</sub>	Syn_Clkin <sub>0</sub>	23
Disable C Buffer	En LM Val1	0	EN Erw Prm	Check En Prm Cmd	En DXB Publisher	Preamble <sub>1</sub>	Preamble <sub>0</sub>	27 24

- XRTS/ADD:** Switchover Output TxE (syn. physics) for different driver control  
 =0 RTS Signal.  
 =1 ADD Signal
- GIM\_EN:** Galvanic Isolation Mode for syn. physics  
 =0 The power-saving interface is switched off  
 =1 The power-saving interface is switched on (possible only for 31.25kBd)
- Quick\_Sync\_New:** Switching on the improved quick sync  
 =0 The improved quick synchronizer is off.  
 =1 The improved quick synchronizer is on.
- EOI\_Timebase:** Time base of the EOI timer  
 =0 The interrupt inactive time is 1 to 2 μsec .  
 =1 The interrupt inactive time is 1 to 2 msec.
- Early\_Ready:** Early Ready Signal  
 =0 Ready is generated if the data is valid (Read) or if the data is taken over (write).  
 =1 Ready is moved ahead by one clock pulse.
- En\_Clock\_Sync:** Enable time of day synchronization  
 =0 The time of day synchronization is blocked  
 =1 The time of day synchronization is enabled
- DP\_Mode:** Enable of DPS  
 =0 DPS is not enabled.  
 =1 DPS is enabled.
- Sync\_Supported:** Support of Sync\_Mode  
 =0 The Sync\_Mode is not supported.  
 =1 The Sync\_Mode is supported. Data is made available in the N-Buffer of the Dout-SM (not comparable to ASIC LSPM 2).
- Freeze\_Supported:** Support of Freeze\_Mode  
 =0 The Freeze\_Mode is not supported.

- =1 The Freeze\_Mode is supported. Data is frozen from the N-buffer of the Din SM (not comparable to LSPM 2).
- Failsafe\_Mode: Support of Failsafe\_Mode  
 =0 Failsafe\_Mode is not supported.  
 =1 Failsafe\_Mode is supported.
- XDP/FDL\_MinTSDR: Default setting of MinTSDR according to 'Baud rate Detect' for DPS or combo operation  
 =0 Pure DPS operation  
 =1 Combo operation

XDP/FDL_MinTSDR	XAsyn/Syn	Baud rate	T <sub>MinTSDR</sub> / t <sub>Bit</sub>
0	-	-	11
1	0	9.6 kBd	30
1	0	19.2 kBd	60
1	0	45.45 kBd	160
1	0	93.75 kBd	125
1	0	187.5 kBd	250
1	0	500 kBd	255
1	0	1.5 MBd	255
1	0	3 MBd	255
1	0	6 MBd	255
1	0	12 MBd	255
1	1	31.25 kBd	100

- New\_GC\_Int\_Mode: Interrupt Mode for 'New\_GC\_Command'  
 =0 The 'New\_GC\_Command Int' is generated only if there is a change in the 'GC\_Command' (basic setting).  
 =1 The 'New\_GC\_Command Int' is generated for each receipt of a GC message.

- Check\_No\_Prm\_Reserved: Reserved bits in Prm frame are not tested  
 =0 Reserved bits are tested.  
 =1 Reserved bits are not tested.

- Check\_No\_GC\_Reserved: Reserved bits in the GC frame are not tested  
 =0 Reserved bits are tested.  
 =1 Reserved bits are not tested.

- XAsyn/Syn: Setting the bus physics  
 =0 Asynchronous physics; the work clock pulse is fixed at 48 MHz (via PLL)  
     Baud rate: 9.6 kBd to 12 MBd (basic setting)  
 =1 Synchronous physics; the work clock pulse can be set: 2, 4, 8 or 16 MHz  
     Baud rate: fixed at 31.25 kBd

- Syn\_Baud: Setting of synchronous baud rate.  
 =0 Only 31.25 kbaud is supported as a synchronous baud rate, irrespective of the clock pulse supply (basic setting).  
 =1 The synchronous baud rate is variable, irrespective of the clock pulse supply (see below).

- En\_Change\_Cfg\_Buffer: Enable of the buffer exchange (User\_Cfg\_Buffer for MAC\_GCfg\_Rbuffer)  
 =0 The buffers won't be exchanged.

- =1 With 'User\_Cfg\_Ok Cmd', the above-mentioned buffers are exchanged. The exchange is confirmed with the interrupt 'Get\_Cfg\_Buffer\_Changed' .
- Syn\_ClkIn<sub>1..0</sub>: Setting the external clock pulse supply at Pin XTAL1\_CLK (not via PLL). The internal C31 processes with half the clock frequency!  
 =00 External clock = 2 MHz ⇒ Baud rates: 31.25  
 =01 External clock = 4 MHz ⇒ Baud rates: 31.25  
 =10 External clock = 8 MHz ⇒ Baud rates: 31.25  
 =11 External clock = 16 MHz ⇒ Baud rates: 31.25
- Check\_DPV1\_Enable: Control of the MSAC1S-SAPs (SAP50-52) is tied to the cyclic machine if "DPV1\_Enable=1" in the Set\_Prm frame..  
 =0 No check of DPV1\_Enable in Set\_Prm frame.  
 =1 DPV1\_Enable is checked in Set\_Prm frame
- Preamble<sub>1..0</sub>: For the syn. physics, the preamble length is parameterized in number of bytes.  
 =00 ⇒ 1 byte  
 =01 ⇒ 2 bytes  
 =10 ⇒ 4 bytes  
 =11 ⇒ 8 bytes
- En\_DXB\_Publisher: Enable as Publisher for internode communication  
 =0 Publisher mode is blocked, DDB request frames are filtered (basic setting).  
 =1 Publisher mode is enabled for all SAPs (for the FDL-SAPs, the user must set Access\_Value=8h).
- Check\_En\_Prm\_Cmd: In the Set\_Prm frame, the 'En\_Prm\_Cmd' bit (DPV1\_Status\_3, Bit7) is scanned and brings about the appropriate responses  
 =0 No check of the En\_Prm\_Cmd bit in the Set\_Prm frame.  
 =1 The En\_Prm\_Cmd bit is checked in the Set\_Prm frame.
- En\_Erw\_Prm: This bit enables the expanded parameter register at address 083Fh.  
 =0 No expanded parameter register present, i.e., all bits of this register are set to the default value. The register is read only.  
 =1 The expanded parameter register is enabled and can be written to.
- En\_LM\_Val1: This bit enables additional evaluation of the Value1 byte for the request interface when "User Leave Master" is requested.  
 =0 No evaluation of Value Byte 1  
 =1 The content of Value Byte 1 is evaluated also.
- Disable\_C\_Buffer: This bit determines whether the Clear Buffer is applied by the user or automatically by the DPC31. However, due to a hardware defect, automatic application of Clear Buffer is not available on the DPC31 Step B.  
 =1 User must output the Clear state to the outputs when required.  
 =0 When required, the DPC31 automatically applies the Clear Buffer using the DP-DOut\_SM.

If the user outputs the Clear Buffer to the outputs, this is signaled using two means:

### Expanded Parameter Register

Additional parameter bits are transferred in the expanded Param Register. These bits may only be changed in MAC state 'Offline.' When the 'MAC\_Start' request is executed, these

parameters are distributed to the individual modules. Subsequent changes are not taken into account by KRISC. The expanded parameter register is available starting with DPC31 Step C but only if Bit #28 (En\_Erw\_Prm) is set to log. '1' in the parameter register.

0	0	0	0	0	0	Sampling_ Mode Bit 1	Sampling_ Mode Bit 0
7							0

Sampling\_Mode<sub>1..0</sub>: Depending on the sampling mode setting, a window of uncertainty results for the differentiation of a short pulse or long pulse when the physics unit is operated in energy saving mode (see Section 5.7.3.3)

=00	⇒ 2.5 µs to 3,0 µs
=01	⇒ 3.0 µs to 3,5 µs
=10	⇒ 3.5 µs to 4,0 µs
=11	⇒ 4.0 µs to 4,5 µs

## 4 ASIC Interface

### Interrupt Controller Register (Int Mask Reg, Int Ack Reg, Int Req Reg, Int and Int EOI Reg):

The meaning of these registers will be explained in later chapters. The interrupt controller exiDXB twice (for ext. µP and C31). Both are instances are mapped onto the same addresses.

#### Mode Register1: (ext. µP and C31, write access)

Mode Register1 is used for parameterizing single bits. These bits are control bits and internally directly affect the hardware. The meaning is described below. Different addresses are used for setting and resetting (Mode Register1 set/reset). A logical '1' is written to those bit positions that are to be changed. All other bit positions must be logical. '0'

Int_ Polarity	0	0	0	Dis_ C31	Dis_ Clkout1X4	Dis_ Clkout1X2	Stop_ Del aytimer
7							0

Stop\_Delay-Timer: The delay timer is stopped  
 =0 The delay timer is not stopped (default).  
 =1 The delay timer is stopped.

Dis\_Clkout1X2: The clock output 'Clkout1X2' is switched off ( $\frac{1}{2}$  of the internal clock: asyn=24MHz, syn=1 to 8 MHz). After being switched on and in the reset phase, the output is initially active.  
 =0 Clkout1X2 is active (default).  
 =1 Clkout1X2 is inactive.

Dis\_Clkout1X4: The clock output 'Clkout1X4' is switched off ( $\frac{1}{4}$  of the internal clock: asyn=12MHz, syn=0.5 to 4 MHz). After being switched on and in the reset phase, the output is initially active.  
 =0 Clkout1X4 is active (default).  
 =1 Clkout1X4 is inactive.

Dis\_C31: The internal C31 is switched off (clock switched off).  
 =0 C31 is active (default).  
 =1 C31 is inactive (absolute powerdown mode).

Int\_Polarity: Polarity of the interrupt output  
 =0 The interrupt output is low-active (basic setting).  
 =1 The interrupt output is high active.

**C31\_Control Register:** (ext.  $\mu$ P and C31, read/write access)

In the C31\_Control register, the settings specific to the C31 are made. The boot type bits are not to be parameterized by the user; the assignment of the chip pins 'BOOTTYP\_0/\_1' determines the boot type.

	Enable DX_Out_Port	Enable_GC_Clock	0	Reserved (0)		Boot Type	
				Bit1	Bit0	Bit1	Bit0
Bit Position	6	5	4	3	2	1	0
Default Value	0	0	0	0	0	-	-
	r/w	r/w	r/w	r/w	r/w	r	r

**Boot type:** These two bits can be used to scan the status at the BOOTTYP0 and BOOTTYP1 pins.

These two pins and the two bits in the register have no specified function any longer. Nevertheless, the pins are to be connected as follows: Bit 0 =0; Bit 1=1.

**Enable GC\_Clock:** The clock pulse striker 'GC\_Clock' is fed to Port PB<sub>2</sub>

=0 Port PB<sub>2</sub> is controlled by C31 (default).

=1 The clock pulse striker 'GC\_Clock' is fed to Port PB<sub>2</sub>

**Ena. DX\_Out\_Port:** The Interrupt 'DX\_OUT' (IRR<sub>14</sub>) can be applied directly to Port PB<sub>3</sub>.

=0 Port PB<sub>3</sub> is controlled by C31 (default).

=1 The interrupt 'DX\_OUT' is fed to Port PB<sub>3</sub>.

## 5 Communication Functions of the Sequential Control System

PROFIBUS Layer2 and the DP slave module are implemented in the sequential control system. Layer2 is composed of a MAC (media access control) part and an FLC (interface services) part. In the following, the Layer2 module is simply called MAC module. The user can influence only the cells that are described here.

## 5.1 Structure of SAP List

All liDXB and buffer elements are located in the communication memory and are addressed via an 8-bit segment buffer pointer. When a list or buffer is accessed, the communication control adds an 8-bit offset address to the segment address shifted by 5-bits (result: 13-bit physical address). A granularity of 32 bytes results for the list and buffer start addresses. The SAP structure includes an 'SAP\_List (addressed via the SAP\_List\_Base-Ptr)' and an 'SAP\_SCB (System Control Block, addressed via the SAP\_SCB\_Base-Ptr)'. In the SAP\_List, a pointer is entered for each SAP (Default\_SAP, SAP0-63 and the SM\_Time\_SAP), which points to the SAP-SCB as an offset. The SAP\_List (66 bytes) must always be available, but only the enabled SAPs are entered in the SAP\_SCB. SAPs not enabled are entered in the SAP\_List with '**SAPx\_Ptr=FFh**'! The entries in the SAP\_SCB must not exceed 256 bytes (10 DPS, 26 FDL and 1 SM Time SAP).

The user enters all FDL\_SAPs in the SAP management and sets the 'DPS\_SAP\_Start Ptr' to the next position after the FDL\_SAPs. With the 'MAC\_Start' request, the MAC affixes the DP-SAPs following this if Bit 'DP\_Mode=1' is set in the Param Register. The entries in the SAP\_SCB differ between the DP SAPs and the FDL SAPs. However, the first three entries for each SAP are always identical. Specifically, the following entries are stored:

req\_sa: When a Request frame is received, the MAC makes the received Remote\_Adr plausible with this specified value. If this field is preassigned with '**=FFh (All)**', access protection is not set. If the field is assigned '**=7Fh**', SAP is blocked and the MAC replies with 'no Service activated (RS)'. Otherwise, the MAC compares the received Remote\_Adr with the specified req\_sa. If they do not match, the frame is rejected and replied to with 'RS'. Attention must be paid that the highest bit has to be set to log. '0' in the default SAP and to log. '1' in all other SAPs. For the FDL-SAPs, the MAC enters the access protection automatically. This always occurs if this field is preassigned "All" and a Request Pdu is received in 'Idle' ISM mode. The access protection can then only be removed via the request interface using 'SAP\_Act/Deact' !

Service\_Supported:

This field is divided into two. The Access value is in the lower 4 bits Ser\_Supported<sub>(3..0)</sub>. If '**=0H (All)**' is preassigned, plausibility step is omitted. Otherwise, the MAC makes the function code from the FC field (FC<sub>3..0</sub>) of the Request frame plausible with the specified Serv\_Supported<sub>3..0</sub>. If no hits occur, the MAC replies with 'RS'.

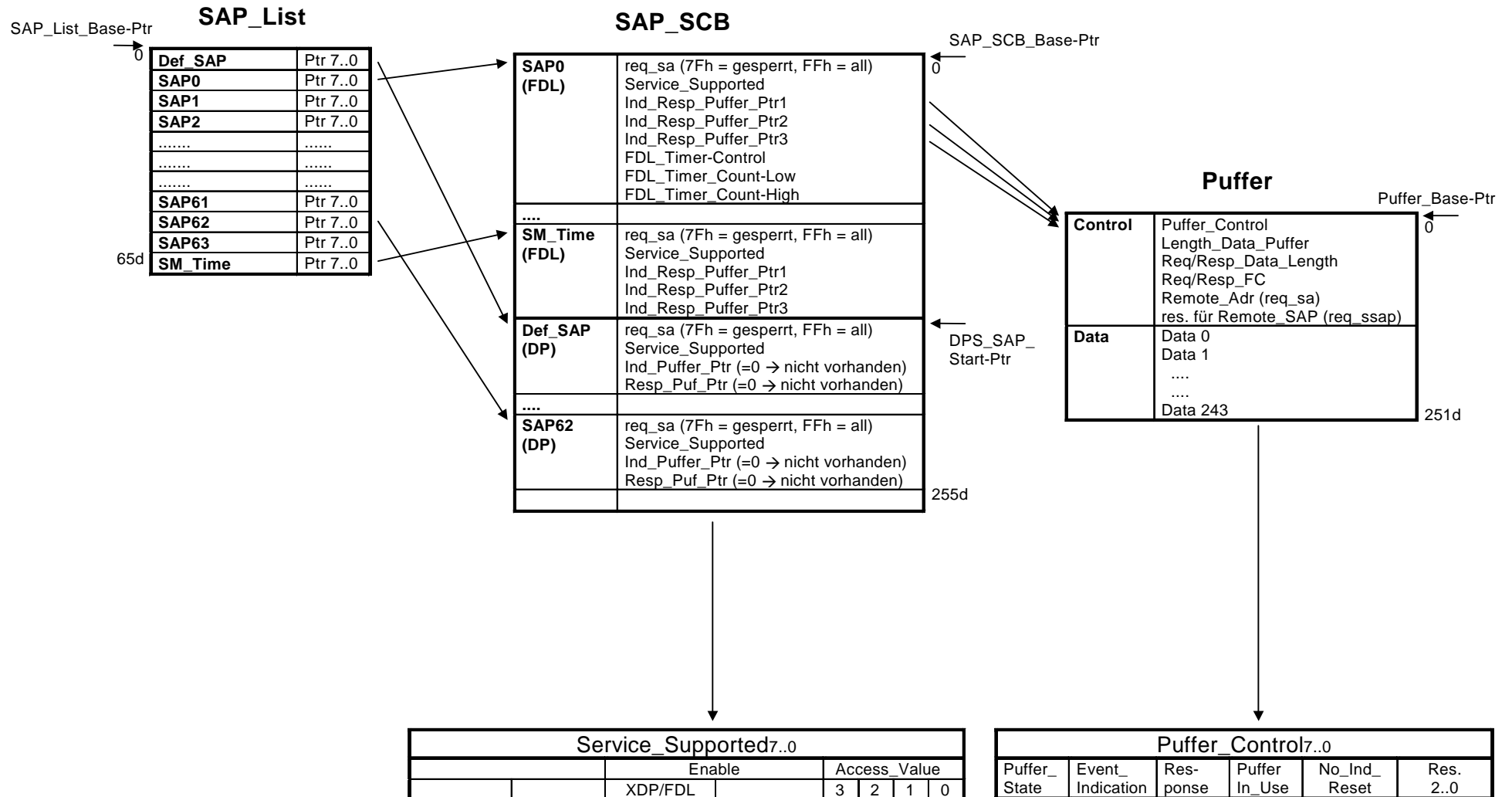


Figure 5.1-1: Structure of SAP List, Buffer Structure

<<Puffer = buffer, gesperrt = blocked>>

Service	Access_Value <sub>3,0</sub>	Service	Access_Value <sub>3,0</sub>
All	0h	SRD-Low/High or DDB-Request	8h
SDN-Low	1h	SRD-Low	9h
SDN-High	2h	SRD-High	Ah
SDN-Low/High	3h	SRD-Low/High	Bh
not allowed	4h	not allowed	Ch
SDA-Low	5h	not allowed	Dh
SDA-High	6h	not allowed	Eh
SDA-Low/High	7h	not allowed	Fh

**Table 5.1-1:** Coding of Access\_Value

In the 'Enable' field, the 'XDP/FDL' bit specifies whether a DP\_SAP (XDP/FDL=0) or an FDL\_SAP (XDP/FDL=1) is involved.

#### Ind\_Resp\_Buffer-Ptr1/2/3:

In the case of FDL-SAPs, up to 3 resources can be hooked in. The Ind\_Resp\_Buffer-Ptr points to the swapped out indication or response buffer. Each buffer operates in '1-buffer mode,' that is, the buffer is available to the SAP as an Indication buffer or as a Response buffer. When buffer lengths are different, the smallest buffer must be assigned the first buffer and the largest buffer the last. The MAC operates according to the 'Best Fit method', i.e., it always attempts to accommodate the received data in the first possible Indication buffer. If all of the Indication buffers are too small or occupied (see next section), it replies with 'No Resource (RR).' If only two buffers or one buffer is made available, more than one pointer must point to the same buffer. When the response is generated, the MAC of Ind\_Resp\_Buffer-Ptr1→3 searches for a Response buffer and sends the response data from the first one found.

#### Ind\_Buffer\_Ptr:

In the case of DP\_SAPs, a fixed Indication\_Buffer is available. In so doing, the respective MAC\_Buffer-Ptr (e.g., Ind\_Buffer-Ptr = MAC\_Prm\_Buffer-Ptr in SAP61) is always hooked in. Because all Indication resources are designed as alternating buffers, a receive resource is available at all times. For DP\_SAPs that require no receive resources (e.g., diagnostic SAP60), the 'Ind\_Buffer-Ptr=00h' is set (by the MAC).

#### Resp\_Buffer-Ptr:

In the case of DP\_SAPs, the Response\_Buffer-Ptr points to the swapped out Response buffer. In the case of DP\_SAPs without a Response buffer (e.g., Set\_Param, Check\_Config, etc.) the MAC enters a value of '00h' in the pointer during startup. It then answers a Request with a short acknowledgement.

For the FDL\_SAPs, the following entries still have to be made:

#### FDL\_Timer-Control:

The control for the FDL monitoring timer is stored in FDL\_Timer control. The control includes an Enable\_Bit and a Reaction coding for timer expiration (see Section 5.3.1).

#### FDL\_Timer\_Count-Low/High:

16-bit timer for monitoring of FDL connections. The timer interval is 10 ms. Times up to 655 s can be realized. The value must be at least 2.

**SAP\_List and the SAP\_SCB for all FDL\_SAPs (incl. SM\_Time) must be configured by the user (int. C31 or ext. µP) during startup!** Changes in the SAP\_List and SAP\_SCB may only be made in Offline status. Dynamic reconfiguring of the SAP list is not permitted! During operation, the DP\_SAPs are established and controlled by the DPS module (access protection). FDL\_SAPs can be opened or locked dynamically. The request 'SAP Act/Deact' is used for this purpose.

### 5.1.1 Configuration of Buffer

The figure below presents the configuration of the Indication and Response buffer for the FDL and DPS SAPs. Both types of buffer appear the same but do not occupy all of the header bytes in the Response buffer.

<b>Indication Buffer1/2, Response Buffer:</b>	Header Field:	Buffer_Control Length_Data_Buffer Request/Response_Data_Length Request/Response_FC Remote_Adr (req_sa) Remote_SAP (req_ssap)
	Data Field:	Data 0 Data 1 ..... Data 243

**Figure 5.1-2:** Configuration of Indication/Response Buffer

The individual entries are explained in more detail below:

#### Header Field:

##### Buffer\_Control:

The Indication buffers are controlled using the 'Buffer\_State' bit. In the case of an Indication, the MAC sets 'Buffer\_State = 1', thus assigning the buffer to the user. In addition, 'Event\_Indication = 1' is set. For a DPS\_Indication buffer, these bits have no significance. The alternating buffer is always available here, i.e., in the case of an Indication, the Indication buffer in the SAP\_SCB is exchanged for another and the corresponding DPS\_SM is triggered, thus generated a DPS interrupt. In the case of FDL, an entry is generated in the Indication\_Queue of 'Request\_Pdu Indication' type and the 'IndQ\_Entry-Interrupt' is also generated. The user must then fetch the entry from the Indication\_Queue. This enables him to receive the SAP\_No. and Indication\_Buffer Address. After readout, he must set bit 'Event\_Indication = 0'.

If the user provides a Response Buffer for FDL, he must also set bit 'Response=1' in 'Buffer\_State=1'. The data are sent at the next Request frame to this SAP (Request Pdu or Poll Pdu). After the Response buffer has been sent, an entry is made in the Indication\_Queue of type 'Poll\_End-Indication' (timing: only after any new request frame has been received). In so doing, the bit 'No\_Ind\_Reset' controls whether the buffer is available afterwards in the SAP as an Indication buffer (No\_Ind\_Reset=0 ⇒ Buffer\_State=0, Response=0) or continues to be available to the user for entering a new response (No\_Ind\_Reset=1 ⇒ Buffer\_State=1, Response=0).

In the case of a DP\_SAP, the Response buffer is available in SAP\_SCB until the user causes it to be swapped.

The 'Buffer\_In\_Use' bit is a MAC-internal control bit, which, in the case of an Indication buffer, indicates that an Indication has been buffered and thus the buffer is occupied (not for DP\_SAPs) or, in the case of the Response buffer, indicates that the Response data are being sent. With the corresponding FDL indication 'Req\_Pdu-Indication or Poll\_End-Indication', the MAC resets this flag.

Buffer_Control <sub>7,0</sub>					
Buffer_State	Event_Indication	Response	Buffer_In_Use	No_Ind_Reset	2..0 Reserved

**Figure 5.1-3:** Assignment of Buffer\_Control

Buffer\_State: Access control to the buffer (only relevant for FDL-SAPs)

0 = MAC

1 = User

Event\_Indication: Display an Indication (only relevant for FDL-SAPs)

- 0 = No Indication
- 1 = Indication
- Response: Control of Response data
  - 0 = The user has provided no response data.
  - 1 = The user has provided response data.
- Buffer\_In\_Use: Indication buffered, or response data are being sent.
  - 0 = No indication buffered, or response-data are not being sent.
  - 1 = An indication is stored in the Indication buffer (not for DP\_SAPs), or the data are being fetched from the response buffer.
- No\_Ind\_Reset: Control of Response buffer after the 'Poll\_End-Indication' (only relevant for FDL-SAPs). In the case of 'Req\_Pdu-Indication', this bit is overwritten by DPC31.
  - 0 = The buffer becomes the Indication buffer and is assigned to the MAC (Buffer\_State=0, Response=0).
  - 1 = The buffer remains for the user and can be used again as a Response buffer (Buffer\_State=1, Response=0).

**Length\_Data\_Buffer:**

This value specifies the length of the data field in the Indication buffer (unoccupied in the Response buffer). If the net data length of the Request frame exceeds the available buffer length, the MAC responds with 'no Resource'.

Comment: SAP is made plausible according to the following sequence:

- Request\_SA: resp-status RS
- Access-Value: resp-status RS
- Length\_Data\_Buffer: resp-status RR

**Request/Response\_Data\_Length:**

This value specifies the length of the entered net data in the Indication or Response buffer (0 to 244 bytes).

**Request/Response\_FC:**

The MAC enters the function code of the Request frame here in the Indication buffer. The following codes are available:

Function	Code
SM Time 0	01xx0000b
SM Time 1	11xx0000b
Send Data with Acknowledge low	01xx0011b
Send Data with no Acknowledge low	01xx0100b
Send Data with Acknowledge high	01xx0101b
Send Data with no Acknowledge high	01xx0110b
DXB-Request	01xx0111b
DXB-Response low	00xx1000b
DXB-Response high	00xx1010b
Send and Request Data low	01xx1100b
Send and Request Data high	01xx1101b

xx: received FCB/FCV

The user stores the Responder status in this field in the Response buffer. The following codes are permitted:

Function	Code
Response FDL/FMA1/2-Data low	00001000b

(& Send Data okay) Response FDL/FMA1/2-Data high (& Send Data okay)	00001010b
---	-----------

Remote\_Adr (req\_sa):

The MAC enters the received Remote\_Address in the Indication buffer (unoccupied in the Response buffer).

Remote\_SAP (req\_ssap):

The MAC enters the received Remote\_SAP here in the Indication buffer (unoccupied in Response buffer).

Special handling for default SAP:

In the case of the default SAP, this byte has a different meaning

**Data Field:**

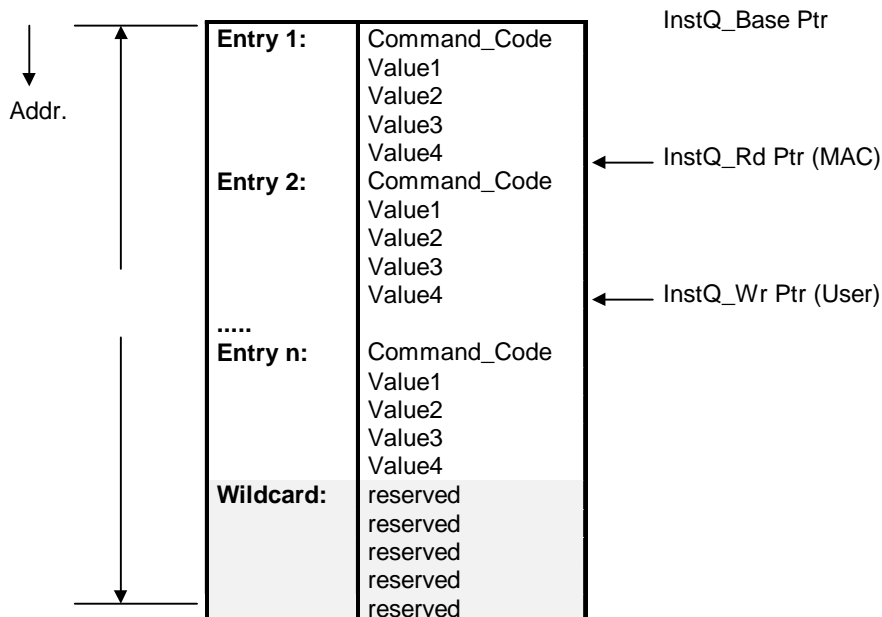
The MAC stores the received net data here in the Indication buffer. The user provides the Response net data in the Response buffer. The maximum data length is 244 characters.

### 5.1.2 Request Interface for DPS and FDL (Instruction Queue)

User request to the DPS and MAC modules are transferred via a request interface. This request list is a polling list onto which the user transfers communication request. Figure 5.1-4 shows the organization of the Instruction\_Queue. With each entry (5 bytes respectively), the user must also transfer the command to the sequential control system. This is done with a write operation with any data value to the register cell 'User\_InstQ\_Write Cmd'. The organization of the Instruction\_Queue includes the following parameters:

InstQ_Base Ptr:	The Instruction_Queue segment pointer
InstQ_Length:	Describes the length of the Instruction_Queue and is a multiple of the length of an entry (n*5)
InstQRd Ptr:	An Offset_Pointer which points to the next entry that is to be read (and is managed by the DPC31)
InstQ_Wr Ptr:	An Offset_Pointer which points to the next free entry (and is managed by the user)

The queue is empty if 'InstQ\_Wr ptr' and 'InstQ\_Rd ptr' point to the same position. **One entry in the queue always must remain empty (wildcard, any content!); otherwise, an empty queue can't be distinguished from a full queue. The user must control the wrap in the queue. After each entry, the user places the InstQ\_Wr ptr behind this entry on the next free position. If this is the end of the queue, the InstQ\_Wr ptr will then have to be placed on the beginning of the queue (wrap around).**



**Figure 5.1-4:** Organization of the Instruction\_Queue

Table 5.1-2 lists all possible requests the necessary command codes.

Request	Com_Code	Value1	Value2	Value3	Value4	Comment
MAC_Start	10h	-	-	-	-	MAC enters Pas_Idle
MAC_Stop	11h	-	-	-	-	MAC enters Offline
MAC_New_TRDY	12h	T <sub>RDY7..0</sub>	-	-	-	Transfer of T <sub>RDY</sub>
SAP_Act/Deact	13h	SAP_No	req_sa	-	-	Activate/deactivate FDL_SAP
En_New_SM_Time-Sync	14h	-	-	-	-	New synchronization is enabled.
User_Leave-Master <sup>1</sup>	15h	-	-	-	-	User initiates a 'Leave Master' without Cfg_Fault
User_Leave-Master <sup>2</sup>	15h	Bit 0: Cfg_Fault Bit 7 to 1: 0	-	-	-	User initiates a: Bit0=1: 'Leave-Master' with Cfg_Fault Bit0=0: 'Leave-Master' without Cfg_Fault
Stop_C1_Con	16h	-	-	-	-	User_Confirmation of 'Stop_C1-Indication'
Set_Addr	17h	Station_Addr	-	-	-	Set new station address.
DPS_User_Wd	20h	00h=reset 01h=enable 02h=disable	-	-	-	Control of DPS User_Watchdog Timer
FDL_Timer_Change	21h	SAP_Nr	Timer_Control	Timer_Low_Val	Timer_High_Val	Change to an FDL Timer
DDB_Enable/Disable	22h	00h=disable 01h=EN_DP_DDB	-	-	-	Activate/deactivate internode communication
Get_Ind_Resp_Buffer	23h	SAP_Nr	Ind_Resp_Buffer_Ptr	-	-	Retrieve an Indication/Response buffer

<sup>1</sup> Im Parameterregister ist das Bit En\_LM\_Val1=0

<sup>2</sup> Im Parameterregister ist das Bit En\_LM\_Val1=1

**Table 5.1-2:** Overview of User\_MAC/DPS Requests

The FMA requeDXB 'SAP\_Act/Deact', 'MAC\_Stop' and 'Get\_Ind\_Resp\_Buffer' are confirmed to the user after execution. For this purpose, an appropriate entry is made in the Indication\_Queue. Only one 'SAP\_Act/Deact Request and one 'Get\_Ind\_Resp\_Buffer Request' can be transferred each time. A new 'SAP\_Act/Deact Request' or 'Get\_Ind\_Resp\_Buffer Request' is possible only after the corresponding confirmation!

With 'Stop\_C1\_Con', the user acknowledges that the MSAC1S SAPS (50-52) are closed. This enables a new opening (entrance in Data\_Exchange)!

#### User-Leave-Master:

If bit En\_LM\_Val1=0 in the parameter register, the user then uses this request to initiate a 'Leave\_Master' without the bit 'Diag.Cfg\_Fault' being set simultaneously in the diagnostics. If on the other hand, bit En\_LM\_Val1=1 is set in the parameter register, the user uses this request—if bit 0 in Value1 is set—to initiate a 'Leave\_Master & Diag.Cfg\_Fault=1' or—if the bit is not set—to initiate a 'Leave\_Master' without bit 'Diag.Cfg\_Fault' being set simultaneously in the diagnostics.

#### Special Operating Cases:

1. If a 'User-Leave-Master' is provided with 'Cfg\_Fault' in the request interface and the diagnostic buffer is being sent during this evaluation, then:
  - Only one internal memory bit is set
  - Diagnostic bit 'Diag.Cfg\_Fault' is set once the diagnostic buffer is available again due to the set memory bit.
2. If two 'User-Leave-Master' (one with and one without Cfg\_Fault) are placed in direct succession in the request interface, diagnostics are not fetched at the present time. That is, the diagnostic buffer is available, then:
  - The request sets the diagnostic bit 'Diag.Cfg\_Fault' with 'Cfg\_Fault'
  - A subsequent request without 'Cfg\_Fault', on the other hand, does **not** reset the diagnostic bit 'Diag.Cfg\_Fault'.
3. If two 'User-Leave-Master' (one with and one without a 'Cfg-Fault') are placed in direct succession in the request interface and the diagnostic buffer is being sent, i.e., the diagnostic buffer is occupied during evaluation of both requeDXB, then:
  - The request with 'Cfg\_Fault' sets only one internal memory bit
- A subsequent request without 'Cfg\_Fault' resets this internal memory bit, i.e., the 'User-Leave-Master' with 'Cfg\_Fault' is suppressed in this case.

#### Set Addr:

The user uses this request to initiate an immediate change of the node address. The node address is evaluated by DPC31 only when each new frame is received. The new node address must be  $\leq 125$ .

The new node address can be read back after the successful conversion in the TS\_Adr\_Register (846h).

### 5.1.3 Acknowledgement Interface for FDL and FMA (Indication\_Queue)

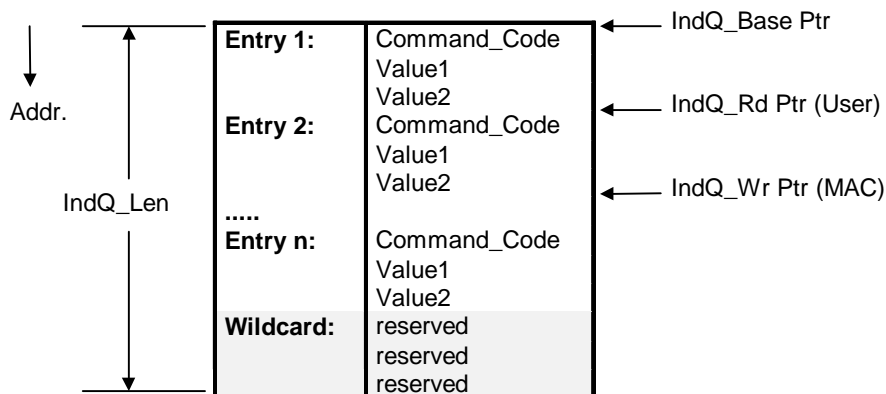
FMA confirmations (for example, MAC\_Reset con; refer to Chapter 5.1.2) are transferred to the user in an Indication\_Queue (polling list). Figure 5.1-5 shows the organization of the Indication\_Queue. With each entry (3 bytes respectively), the 'IndQ\_Entry Int' is additionally generated for the user. If the queue is full and the MAC is to make another entry, this indication is abandoned and the 'IndQ\_Full Int' is set (refer to Chapter 7.1.4). The user should avoid this condition by dimensioning the queue accordingly large. There is no effect on the bus (for example, no RR if the queue is full).

The organization of the Indication\_Queue includes the following parameters:

- IndQ\_Base Ptr: The Indication\_Queue segment pointer
- IndQ\_Length: Describes the length of the Indication\_Queue, and is a multiple of the length of an entry (n\*3)
- IndQRd Ptr: An Offset\_Pointer and points to the next entry that is to be read (and is managed by the user)
- IndQ\_Wr Ptr: An Offset\_Pointer and points to the next free entry (and is managed by the MAC)

The queue is empty if 'IndQ\_Wr Ptr' and 'IndQ\_Rd Ptr' point to the same position. **One entry in the queue always has to remain empty.**

Table 5.1-3 lists all possible indications with the associated command codes.



**Figure 5.1-5: Organization of the Indication\_Queue**

Request	Com_Code	Value1	Value2	Comment
Req_Pdu-Ind	40h	SAP_Nr.	Buf_Ptr	Request_Pdu received <sup>3</sup>
Poll_End-Ind	41h	SAP_Nr.	Buf_Ptr	Response sent, Poll end
Stop_C1-Ind	42h	-	-	MSAC1S-SAPs have been closed
SAP_Act/Deact-Confirmation	80h	SAP_Nr.	req_sa	FDL_SAP activated/deactivated
MAC_Stop Confirmation	81h	-	-	MAC_Stop was executed
Get_Ind_Resp_Buffer-Confirmation	82h	SAP_Nr.	Ind_Resp_Buffer_Ptr	The requested Indication/Response-buffer has been retrieved.
FDL_Timer Expired	83h	SAP_Nr.	FDL_Timer_Control	FDL monitoring timer expired
DPS_User WD Expired	84h	-	-	DPS_User Watchdog timer expired

**Table 5.1-3: Overview of Indications and Confirmations**

**Note:**

MAC\_Stop confirmation confirms the MAC transition to the *Offline mode* after the current request has been processed.

SAP\_Act/Deact-Confirmation confirms the execution of this FMA request.

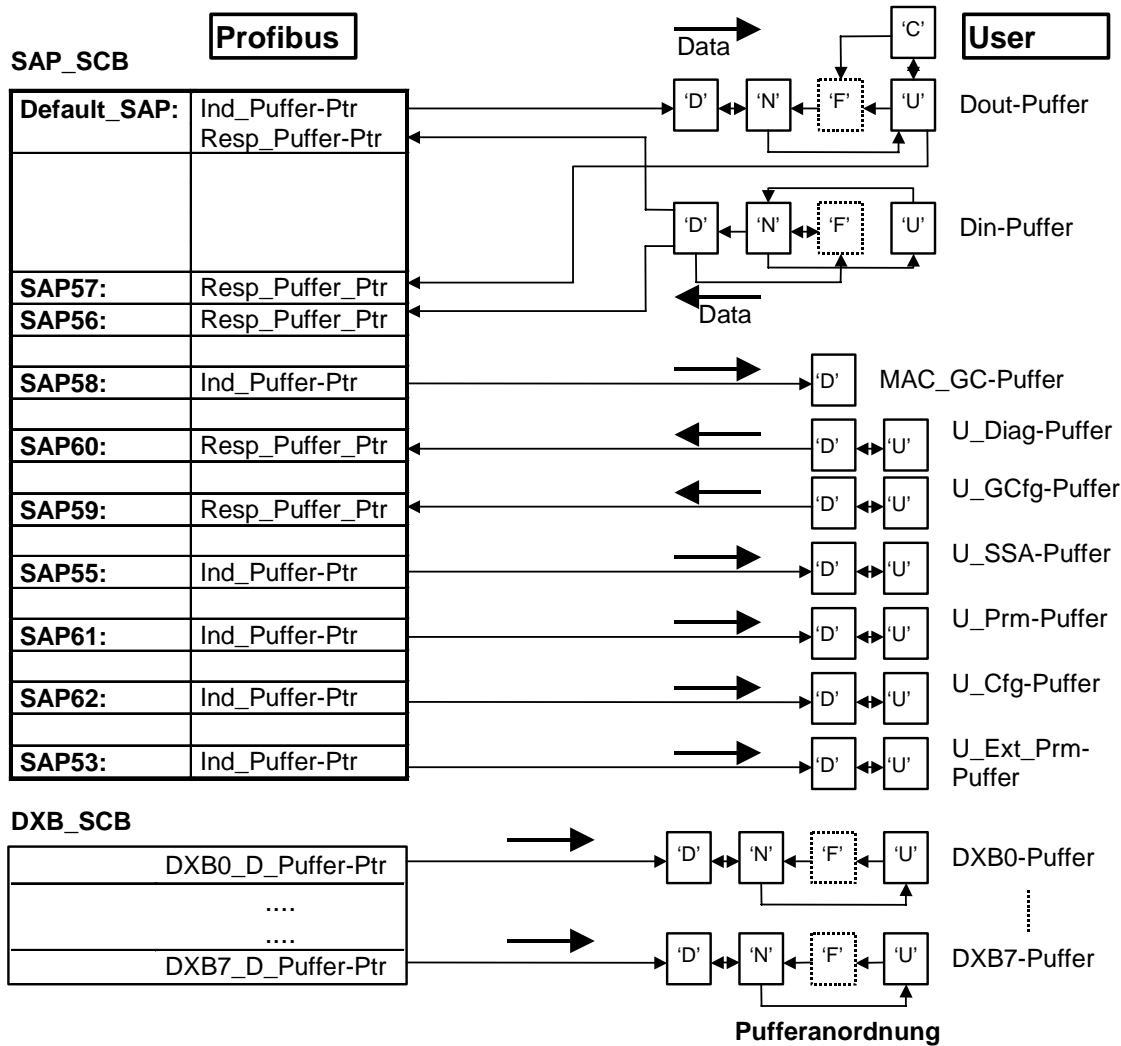
Get\_Ind\_Resp\_Buffer-Confirmation confirms retrieval of the buffer.

## 5.2 DPS Module, Description of the Interface

DPS is enabled in the param register with 'DP\_Mode=1', and started in the Instruction\_Queue with the MAC request 'MAC\_Start'. The user can disable the SAP55 (Set\_Slave\_Address).

The DPS protocol is integrated completely into the DPC31. All other DP SAPs are always enabled except for the following: default SAP, SAP 56, SAP57, and SAP58. The remaining four SAPs are enabled only when the 'Data\_Exchange' mode is entered.

<sup>3</sup>



Fig

ure 5.2-1: DPS Buffer Structure

Figure 5.2-1 shows the DPS buffer structure. The buffers (length and buffer ptr) are configured by the user in the 'Offline Mode' in the DPS buffer management. For the Dout data, four buffers of the same length are available and, for the DXB0-7 data (separate management in DDB\_SCB), three buffers each of the same length are available, which are implemented as alternating buffers. One buffer each is assigned to the incoming data transfer 'D' and the user 'U'. The third buffer is either in a Next 'N' or Free 'F' mode, whereby one of the two modes is always unoccupied. The MAC clears the data in 'D'. After receiving, 'D' is moved to 'N', and a new buffer is fetched from the 'N' or 'F'. The user fetches its output data from 'N'. In the fourth buffer 'C', the user makes the substitute values available for the Clear mode (failsafe). If the DPC31 receives Clear messages or if DPS leaves the 'Data\_Exchange' mode, the 'C' buffer is transferred to the user in the state 'U'. The buffers are moved through the corresponding exchange. DPS then also performs the buffer exchange for the user.

The Din data is controlled via three exchange buffers of the same length. One buffer each is assigned to the data transfer 'D' and the user 'U'. The third buffer is either in a Next "N" mode, or Free 'F' mode. When sending, the MAC fetches the Din data from 'D'. The user prepares new Din data in 'U' and then moves it to 'N'. DPS then changes the buffers from 'N' to 'D'.

For the diagnostic SAP and the Get\_Cfg SAP (SAP60/59), two buffers respectively are available that may have different lengths. The 'D' buffer is always assigned to the MAC for sending and the 'U' buffer belongs to the user for preparing new data. DPS exchanges the buffers upon user request.

In SAP55 (Set\_Slave\_Address), SAP53 (Set\_Ext\_Param), SAP61 (Set\_Param), and SAP62 (Check\_Config), one indication buffer, respectively, is available, in which the received data are cleared. At the indication, this buffer is exchanged for the corresponding buffer in DPS buffer management (User\_SSA buffer, User\_DDB\_Prm buffer, User\_Prm buffer, or User\_Cfg buffer) and then the corresponding DPS control unit is triggered.

### 5.2.1 Set\_Slave\_Address, SSA (SAP55)

Two exchange buffers of the same length are available for this SAP. One buffer is integrated as indication buffer in the SAP\_SCB (MAC\_SSA buffer) and the other is included in DPS buffer management as User\_SSA buffer. The indication is always transferred to the user in User\_SSA Buffer.

The user can disable the SSA service by setting the 'MAC\_SSA\_Buffer Ptr=00h' at power-up. The DPC31 then responds to an SSA request with 'no service activated'.

The new 'Station Address' and the parameter 'Real\_No\_Add\_Change' are stored by the user and retransferred to the software modules "MAC and DPS" after every restart caused by a voltage failure, for example.

If the DPC31 receives a Set\_Slave\_Address message, and if the SAP55 is enabled, the MAC first checks whether the indication buffer has the corresponding size. If not, the MAC responds with 'No Resource'. Otherwise, it sends a short acknowledgement and after the send process transfers this buffer to the DPS module. The MAC has already accepted the new station address, however.

7	6	5	4	3	2	1	0	Byte	Name
								0-5:	Buffer Header
								6:	New_Slave_Address
								7:	Ident_Number_High
								8:	Ident_Number_Low
								9:	No_Add_Change
								10-249:	Rem_Slave-Data

**Figure 5.2-2: Assignment in the Data Field of the SSA Indication Buffer**

In the following states, the DPS module ignores the SSA indication:

- DP\_SM mode 'Wait\_Cfg', 'Data\_Exchange'
- Net data length less than 4 bytes
- Parameter 'Real\_No\_Add\_Change' is 'True' (FFh)
- New station address is larger than 125
- Ident No. is wrong

**User\_SSA\_OK Cmd** (Read Operation):

0	0	0	0	0	0	User_Ack <sub>1</sub>	User_Ack <sub>0</sub>
---	---	---	---	---	---	-----------------------	-----------------------

- User\_Ack<sub>1,0</sub> = 00 ⇒ User\_SSA\_Finished
- User\_Ack<sub>1,0</sub> = 01 ⇒ SSA\_Conflict
- User\_Ack<sub>1,0</sub> = 11 ⇒ Not\_Allowed
- User\_Ack<sub>1,0</sub> = 10 ⇒ not possible

**Table 5.2-1: Coding of User\_SSA\_OK Cmd**

The acknowledgement 'User\_SSA\_OK Cmd' is a read access to a register cell with the corresponding codes 'Not\_Allowed', 'User\_SSA\_Finished', or 'SSA\_Conflict'.

The SSA\_State\_Machine is reset also when the DPS is powered up -that is, after the user has transferred 'MAC\_Start' in the request list- or the watchdog has expired in the mode 'DP\_Control'. If the SSA message is repeated because the short acknowledgement was faulty on the bus, the MAC ignores it because it has already accepted the new station address.

### 5.2.2 Set\_Param, Prm (SAP61)

For this SAP, two exchange buffers of the same length are available. One buffer is integrated as the indication buffer (MAC\_Prm buffer) and the other is located as the User\_Prm buffer in DPS buffer management. The indication is always transferred to the user in the User\_Prm buffer.

The DPS module accepts this request in any DPS mode (Wait\_Prm, Wait\_Cfg, Data\_Ex). However, the message has to have at least a length of  $\geq 7$  bytes; otherwise, it is ignored.

7	6	5	4	3	2	1	0	Byte	Name
								0-5:	Buffer Header
Lock_Req	Unlock_Req	Sync_Req	Freeze_Req	WD_On	Res.	Res.	Res.	6:	Station Status
								7:	WD_Fact_1
								8:	WD_Fact_2
								9:	MinTSDR
								10:	Ident_Number_High
								11:	Ident_Number_Low
								12:	Group_Ident
DPV1_Enable	Failsafe	EN_D P_DXB	res	res.	WD_Base	res	res	13:	DPV1_Status_1 (Spec_User_Prm_Byte)
								14:	DPV1_Status_2
EN_Prm_Cmd								15:	DPV1_Status_3
								16-249:	Rem_Slave-Data

**Figure 5.2-3: Assignment in the Data Field of the PRM Indication Buffer**

Byte 13 is permanently reserved for the DPC31 and **must not** be used for User Prm data. The bytes 13 to 15 are reserved according to DPV1 and **should not** be used for User Prm data in order to make a compatible change to DPV1 possible.

Byte 13, Bit 2: WD\_Base:

This bit specifies the time base of the watchdog in DP\_Control; the following assignment applies.

WD\_Base = 0: time base of 10 ms

WD\_Base = 1: time base of 1 ms

Default state: WD\_Base= 0.

Byte 13, Bit 4: Must be assigned log. '0'!

Byte 13, Bit 5: EN\_DP\_DXB:

The bit specifies whether DDB\_Request and DDB\_Response frames filter or pass through on the default SAP.

EN\_DP\_DXB=0: Only DXB\_Request frames are filtered

EN\_DP\_DXB=1: Error-free DDB\_Request and DXB\_Response frames are are passed through and processed by the MAC.

Default state: EN\_DP\_DXB = 0.

Byte 13, Bit 7: DPV1\_Enable:

This bit enables the coupling of the MSAC1S-SAPs to the cyclic machine if 'Check\_DPV1\_Enable=1' (Param register) and the parameter assignment frame contains a data length  $\geq 10$  bytes.

DPV1\_Enable=0: No coupling of MSAC1S-SAPs.

DPV1\_Enable=1: Coupling of MSAC1S-SAPs.

Byte 15, Bit 7: 'Check\_En\_Prm\_Cmd=1' in the Param register enables 'Enable\_Prm\_Cmd' in the Set\_Prm frame (DPV1\_Status\_3, Bit 7) to be checked. However, this occurs only in the 'Data\_Ex' and 'Operation-Mode=V1' state when a new Prm frame (V1) is received from the same master. If 'Enable\_Prm\_Cmd=1', DPS remains in 'Data\_Ex' state. When 'Enable\_Prm\_Cmd=0', a Leave\_Master is executed. In the case of this Leave\_Master, the diagnostic bit 'Diag\_Prm\_fault' is set.

DPS evaluates the first 7 bytes or the first 10 bytes for longer Prm messages (refer to Figure 5.2-3). The evaluation is performed according to EN 50 170 Volume 2 and will not be discussed in more detail in this description.

In the case of negative validation, DPS sets corresponding diagnostic bits and branches into the 'Wait\_Prm mode'. If the master requests 'Sync\_Req' or 'Freeze\_Req' and the application does not support 'Sync' or 'Freeze' (Sync\_Supported=0, Freeze\_Supported=0 in the param register), the Prm message is not accepted and the diagnostic flag 'Diag.Not\_Supported = 1' is set. In case of positive validation (new, valid message), DPS makes the transition to 'Wait\_Cfg', and executes the following responses, depending on the data length:

- If 'Lock\_Req = 0' and 'Unlock\_Req = 0', only the parameter 'MinTSDR' is accepted internally (S/R unit) and no response is initiated to the user. If 'MinTSDR = 00H', the old value is saved. The S/R unit waits at least  $11 T_{\text{Bit}}$  prior to sending its response messages. If a MinTSDR < 11 is parameterized, the time is set to 11 by the ASIC.
- If 'Lock\_Req = 1' and 'Unlock\_Req = 0', the DPS accepts the following values: Flag: WD\_ON; watchdog factors: WD\_FACT1/2; the min station delay response: MinTSDR (if it differs from 0 and >10); group generation: Group\_Ident; the master address: Master\_Add. For messages that are longer than 7 net parameter data bytes, the bits from the Spec\_User\_Prm\_Byte are also accepted; otherwise, these bits are assigned default values. The user indication New\_Prm\_Data is then triggered.

The acknowledgements 'User\_Prm\_OK cmd/User\_Prm\_Not\_OK cmd' are read accesses to defined register cells with the corresponding messages 'Not\_Allowed', 'User\_Prm\_Finished', or 'Prm\_Conflict' (refer to Table 5.2-2).

**User\_Prm\_OK Cmd** (Read Operation):

0	0	0	0	0	0	User_Ack <sub>1</sub>	User_Ack <sub>0</sub>
---	---	---	---	---	---	-----------------------	-----------------------

User\_Ack<sub>1..0</sub> = 00 ⇒ User\_Prm\_Finished  
 User\_Ack<sub>1..0</sub> = 01 ⇒ Prm\_Conflict  
 User\_Ack<sub>1..0</sub> = 11 ⇒ Not\_Allowed  
 User\_Ack<sub>1..0</sub> = 10 ⇒ not possible

**User\_Prm\_Not\_OK Cmd** (Read Operation):

0	0	0	0	0	0	User_Ack <sub>1</sub>	User_Ack <sub>0</sub>
---	---	---	---	---	---	-----------------------	-----------------------

User\_Ack<sub>1..0</sub> = 00 ⇒ User\_Prm\_Finished  
 User\_Ack<sub>1..0</sub> = 01 ⇒ Prm\_Conflict  
 User\_Ack<sub>1..0</sub> = 11 ⇒ Not\_Allowed  
 User\_Ack<sub>1..0</sub> = 10 ⇒ not possible

**Table 5.2-2:** Coding of User\_Prm\_(Not)\_OK Cmd

### 5.2.3 Check\_Config, Chk\_Cfg (SAP62)

For this SAP, two exchange buffers of the same length are allocated. One buffer is integrated as the indication buffer (MAC\_Cfg buffer) and the other is included as the User\_Cfg buffer in DPS buffer management. The indication is always transferred to the user in the User\_Cfg buffer. This service is accepted by DPS in any DP mode. If the Check\_Config message does not come from 'Master\_Add' i.e., the locking master, DPS ignores this message. The user evaluates the configuration data. After DPS has received a plausible Cfg message, there will be an indication. That is, DPS exchanges the indication buffer in the Cfg SAP for the User\_Cfg buffer from DPS buffer management and generates the 'New\_Cfg\_Data interrupt'. There is no response at this time in the DP\_SM. The user must then check the 'User\_Config\_Data' and acknowledge either positively or negatively (see below).

**User\_Cfg\_Ok Cmd** (Read Operation):

0	0	0	0	0	0	User_Ack <sub>1</sub>	User_Ack <sub>0</sub>
---	---	---	---	---	---	-----------------------	-----------------------

User\_Ack<sub>1..0</sub> = 00 ⇒ User\_Cfg\_Finished  
 User\_Ack<sub>1..0</sub> = 01 ⇒ Cfg\_Conflict  
 User\_Ack<sub>1..0</sub> = 11 ⇒ Not\_Allowed  
 User\_Ack<sub>1..0</sub> = 10 ⇒ not possible

**User\_Cfg\_Not\_Ok Cmd** (Read Operation):

0	0	0	0	0	0	User_Ack <sub>1</sub>	User_Ack <sub>0</sub>
---	---	---	---	---	---	-----------------------	-----------------------

User\_Ack<sub>1..0</sub> = 00 ⇒ User\_Cfg\_Finished  
 User\_Ack<sub>1..0</sub> = 01 ⇒ Cfg\_Conflict  
 User\_Ack<sub>1..0</sub> = 11 ⇒ Not\_Allowed  
 User\_Ack<sub>1..0</sub> = 10 ⇒ not possible

**Table 5.2-3:** Coding of User\_Cfg\_(Not)\_OK Cmd

During operation, if the interrupts 'New\_Prm\_Data', New-Ext\_User\_Prm\_Data and 'New\_Cfg\_Data' are pending at the user at the same time, the user must follow the sequence Set\_Param, Set\_Ext\_User\_Param and then Check\_Config acknowledgement.

### 5.2.4 Slave\_Diagnosis (SAP60)

The diagnostic data of DPS in the DPC31 can be fetched by the master any time.

When the buffers are exchanged by the user, the internal 'Diag\_Flag' is set in. Furthermore, the Diag\_Flag is entered in the status register. If 'Diag\_Flag' is activated, the MAC responds at the next Write\_Read\_Data message with high priority response data. This signals to the associated master that new diagnostic data is present at the slave. If DPS does not have any input data, it responds with a high-priority SD2 message with a dummy net byte (00h). After this high priority reply, the master fetches the new diagnostic data with a Slave\_Diagnosis message. The 'Diag\_Flag' is then reset and the user 'Diag\_Fetched interrupt' is generated. However, if the user signals 'Diag.Stat\_Diag = 1" (static diagnosis; refer to structure of the Diagnosis\_Reply buffers), the 'Diag\_Flag' remains activated even after the associated master has fetched the diagnosis. The user can poll the 'Diag\_Flag' in the status register. DPS sets 'Diag\_Flag=0' for 'Power\_On', caused by a reset or the startup of the watchdog timer in the 'DP\_Control mode'; or 'Diag\_Flag = 1' when entering 'Data\_Exchange'. The Diag\_Buffer\_SM is also reset when DPS is powering up. That is, after the user has transferred 'MAC\_Start' in the request list or the watchdog has expired in the 'DP\_Control' mode. A 'Diag\_Buffer\_Changed' interrupt, if present, is not deleted.

### Structure of the Diagnosis\_Reply Buffers:

The user transfers the Diagnostic\_Reply buffer shown in Figure 5.2-4. The buffer control area is located in the first 6 bytes. In the 7<sup>th</sup> byte, the user only enters the bit 'Diag.Ext\_Diag' and in the 8<sup>th</sup> byte the bit 'Diag.Stat\_Diag'. The remaining bits in these two bytes can be assigned as required. The user sets up Byte 9 (StationStatus\_3), Byte 11,12 (Ident\_Number) and Byte 13..250 (Ext\_Diag data) completely. Byte 10 is used as wildcard for 'Master\_Add' and can be assigned as required. During buffer exchange, DPS enters the internal Diagnosis\_Flags in Bytes 7 and 8 and also enters the 'Master\_Add' in Byte 10 (refer to Figure 5.2-5).

7	6	5	4	3	2	1	0	Byte	Name
-	-	-	-	Diag.Ext_Diag	-	-	-	0-5:	Buffer Header
-	-	-	-	-	-	Diag.Stat_Diag	-	6:	StationStatus_1
Diag.Ext_Diag_Overflow	0	0	0	0	0	0	0	7:	StationStatus_2
-	-	-	-	-	-	-	-	8:	StationStatus_3
-	-	-	-	-	-	-	-	9:	Wildcard
-	-	-	-	-	-	-	-	10:	Ident_Number_High
-	-	-	-	-	-	-	-	11:	Ident_Number_Low
-	-	-	-	-	-	-	-	12-249:	Ext_Diag-Data

Figure 5.2-4: Structure of the User\_Diag\_Reply Buffer

7	6	5	4	3	2	1	0	Byte	Name
0	Diag.Prm_Fault	0	Diag.Not_Supported	Diag.Ext_Diag	Diag.Cfg_Fault	Diag.Station_Not_Rdy	0	0-5:	Buffer Header
0	0	Diag.Sync_Mode	Diag.Freez_Mode	Diag.WD_On	1	Diag.Stat_Diag	Diag.Prm_Req	6:	StationStatus_1
Diag.Ext_Diag_Overflow	0	0	0	0	0	0	0	7:	StationStatus_2
-	-	-	-	-	-	-	-	8:	StationStatus_3
-	-	-	-	-	-	-	-	9:	Master_Address
-	-	-	-	-	-	-	-	10:	Ident_Number_High
-	-	-	-	-	-	-	-	11:	Ident_Number_Low
-	-	-	-	-	-	-	-	12-249:	Ext_Diag-Data

Figure 5.2-5: Structure of the MAC\_Diag\_Reply Buffer

### 5.2.5 Write\_Read\_Data (Default SAP)

The MAC accepts the Write\_Read\_Data message only in the 'Data\_Exchange' mode and only from the 'Master\_Add' i.e., the locking master; otherwise, a negative acknowledgement 'RS' is generated. If the received net data (output data) does not fit into indication buffer 'D', the service is ignored and the response is 'no resource'.

The length of the indication buffer 'D' corresponds exactly to the data output configuration of the respective slave. If the received output data is less than the length of the indication buffer, there is a configuration error. In this case, DPS does the following: it sets 'Diag.Cfg\_Fault =1' (refer to diagnostic data), executes the 'Leave\_Master macro' transitioning to 'Wait\_Prm') and transmits the input data from the response buffer. Otherwise, the received net data is written to the assigned indication buffer and the net data that is to be sent is fetched from the assigned response buffer. For the output data, 4 exchange buffers are available and for the input data, 3 exchange buffers.

Two operating modes must be able to be taken into account:

- **Operation without Clear Buffer**

This operating mode occurs if bit Disable\_C\_Buffer=1 in the parameter register. In this case, only three alternating buffers are operated and the Clear buffer is blocked from use. The clear buffer is no longer automatically injected. A check by adding substitute values at the outputs is now done with the user software.

- **Operation with Clear Buffer**

This operating mode occurs if bit Disable\_C\_Buffer=0 in the parameter register. In this case, four alternating buffers are operated. The Clear buffer is now automatically injected and discharged in the buffer circulation. A check by adding the C buffer and, thus, the substitute values, occurs automatically.

Operation with Clear buffer is always preferred over operation without Clear buffer. If operation without Clear buffer is employed, the Clear buffer is not used. Rather, the current 'D' is transferred to the 'N' buffer with 'DPS-Dout-Change' and an 'DX\_Out\_Interrupt' is initiated. However, in comparison with a 'normal' D→N buffer changeover, the data in the buffer are now invalid. In place of the invalid data, the substitute values must now be output by the user.

In this operating mode, the user must determine for each 'User\_New\_Dout-Cmd' whether the data from the buffer or the substitute values are to be output. Two registers are available to the user for evaluation:

- The 'U\_Buffer\_State' register on address 834h
- The 'req\_ssap' byte in the buffer header of the current U buffer (applies only to the buffer of the default SAP).

In these two registers, only Bit #3 has a function, all other bits are log. '0'.

Bit #3 = 1: The data transferred in the current U-buffer to the user are invalid. The substitute values must be output in place of the data from the U-buffer.

Bit #3 = 0: The data transferred to the user in the current U-buffer are valid and can be output by the user.

Additional particularities in conjunction with this operating mode:

- In this operating mode, bit 'U\_Buffer\_Cleared' in the 'User-New\_Dout-Cmd' register is not updated and can therefore also not be used for evaluation
- If Bit #3 in the 'req\_ssap' byte of the current U-buffer is set and this buffer is entered by the MAC as a reply buffer for the read output SAP (SAP 57), the data length is always set to the value from the 'Dout\_Buffer-Length' register in this buffer header in byte 'Resp\_Data\_Length'. The data sent as a response for a Read Output SAP are invalid/incidental.
- Under certain circumstances, the content of the 'U\_Buffer\_State' register can be inconsistent with the content of the user's buffer.  
For this reason, the 'req\_ssap' byte from the buffer header should be used.

- The 'req\_ssap' byte in the buffer header is only updated if an output buffer is present, i.e., Dout\_Buffer-Length is >0.

### 5.2.5.1 DX-Entered-Interrupt

If the MAC\_State\_Machine changes from 'passive-Idle' state to a Set\_Param\_Frame (SAP61) and the Check\_Config-Frame (SAP62) either through a Write\_Read\_Data-Frame (Default\_SAP), a fail-safe frame (Default-SAP), or a Global\_Control\_Clear-Frame(SAP58) in 'Data\_Exchange' state, the 'DX-Out-Interrupt' is triggered after each of these frames (provided it is enabled).

In addition, the 'DX-Entered' interrupt is triggered with the first of these frames, if diagnostic messages are no longer present.

This interrupt is only triggered again if the 'Data\_Exchange' state is left and then reached.

With the read operation 'User\_Dout\_Buffer state', the user receives the current buffer assignment without initiating a buffer exchange!

**User\_New\_Dout Cmd** (Read Operation):

0	0	0	0	U_Buffer_Cleared	U_Buffer_State	U_Buffer <sub>1</sub>	U_Buffer <sub>0</sub>
---	---	---	---	------------------	----------------	-----------------------	-----------------------

U\_Buffer\_State = 0 ⇒ no new U\_Buffer

U\_Buffer\_State = 1 ⇒ new U\_Buffer

U\_Buffer\_Cleared = 0 ⇒ received data

U\_Buffer\_Cleared = 1 ⇒ substitute values

U\_Buffer<sub>1..0</sub>=00 ⇒ Buffer4

U\_Buffer<sub>1..0</sub>=01 ⇒ Buffer1

U\_Buffer<sub>1..0</sub>=10 ⇒ Buffer2

U\_Buffer<sub>1..0</sub>=11 ⇒ Buffer3

**User\_Dout\_Buffer State** (Read Operation):

F_Buffer <sub>1</sub>	F_Buffer <sub>0</sub>	U_Buffer <sub>1</sub>	U_Buffer <sub>0</sub>	N_Buffer <sub>1</sub>	N_Buffer <sub>0</sub>	D_Buffer <sub>1</sub>	D_Buffer <sub>0</sub>
-----------------------	-----------------------	-----------------------	-----------------------	-----------------------	-----------------------	-----------------------	-----------------------

F/N-Buffer<sub>1..0</sub> = 00 ⇒ Nil

F/N-Buffer<sub>1..0</sub> = 01 ⇒ Buffer1

F/N-Buffer<sub>1..0</sub> = 10 ⇒ Buffer2

F/N-Buffer<sub>1..0</sub> = 11 ⇒ Buffer3

U-Buffer<sub>1..0</sub> = 00 ⇒ Buffer4

U-Buffer<sub>1..0</sub> = 01 ⇒ Buffer1

U-Buffer<sub>1..0</sub> = 10 ⇒ Buffer2

U-Buffer<sub>1..0</sub> = 11 ⇒ Buffer3

D-Buffer<sub>1..0</sub>=00⇒not possible

D-Buffer<sub>1..0</sub>= 01 ⇒ Buffer1

D-Buffer<sub>1..0</sub>= 10 ⇒ Buffer2

D-Buffer<sub>1..0</sub>= 11 ⇒ Buffer3

**Table 5.2-4:** Coding of User\_New\_Dout Cmd, User\_Dout\_Buffer State

With the read operation 'User\_Din\_Buffer State', the user receives the current buffer assignment without the buffer being exchanged!

**User\_New\_Din Cmd** (Read Operation):

0	0	0	0	0	0	U_Buffer <sub>1</sub>	U_Buffer <sub>0</sub>
---	---	---	---	---	---	-----------------------	-----------------------

U\_Buffer<sub>1..0</sub>=00⇒not possible

U\_Buffer<sub>1..0</sub> = 01 ⇒ Buffer1

U\_Buffer<sub>1..0</sub> = 10 ⇒ Buffer2

U\_Buffer<sub>1..0</sub> = 11 ⇒ Buffer3

**User\_Din\_Buffer State** (Read Operation):

F_Buffer <sub>1</sub>	F_Buffer <sub>0</sub>	U_Buffer <sub>1</sub>	U_Buffer <sub>0</sub>	N_Buffer <sub>1</sub>	N_Buffer <sub>0</sub>	D_Buffer <sub>1</sub>	D_Buffer <sub>0</sub>
-----------------------	-----------------------	-----------------------	-----------------------	-----------------------	-----------------------	-----------------------	-----------------------

F/N-Buffer<sub>1..0</sub> = 00 ⇒ Nil

F/N-Buffer<sub>1..0</sub> = 01 ⇒ Buffer1

F/N-Buffer<sub>1..0</sub> = 10 ⇒ Buffer2

F/N-Buffer<sub>1..0</sub> = 11 ⇒ Buffer3

U/D-Buffer<sub>1..0</sub>=00 ⇒ not possible

U/D-Buffer<sub>1..0</sub> = 01 ⇒ Buffer1

U/D-Buffer<sub>1..0</sub> = 10 ⇒ Buffer2

U/D-Buffer<sub>1..0</sub> = 11 ⇒ Buffer3

**Table 5.2-5:** Coding of User\_New\_Din Cmd and User\_Din\_Buffer State

At startup, the DP\_SM goes to 'Data\_Exchange' only after a positive user acknowledgement of User\_Cfg\_OK cmd' has followed a Check\_Config message, and additionally, the first valid Din buffer was made available in 'N' with the 'User\_New\_Din cmd'.

**DPS\_User Watchdog:**

After power-up ('Data\_Exchange' mode), it is possible that the DPC31 continuously replies to Write\_Read\_Data messages without the user fetching the received Dout buffers or making new Din buffers available. If the user processor should "hang", the master would not notice it. For that reason, a 'DPS\_User watchdog' is implemented in DPS. This timer can be enabled or disabled any time via the request interface (DPS\_User WD, Enable; or DPS\_User WD, Disable).

Note: In the case of the SPC3, the processor is monitored via a counter.

The DPS\_User\_Watchdog is an internal 16bit RAM cell that is started by a user-parameterized value 'DPS\_User WD Value<sub>15.0</sub>', and is decremented every 10 msec. If the timer reaches the value '0000h', DPS does the following: it executes 'Leave\_Master', locks the DPS\_User WD, and enters the event 'DPS\_User\_WD Expired' in the Indication\_Queue.

The user has to cyclically set this timer to its initial value. To do this, the user must transfer 'DPS\_User WD, Reset' via the request interface. DPS then reloads the timer to the parameterized value 'DPS\_User WD Value<sub>15.0</sub>'.

With 'DPS\_USER WD, Enable' request, the DPS\_User WD is automatically set to its initial value and started.

**5.2.6 Global\_Control (SAP58)**

The MAC accepts the Global\_Control message only in the 'Data\_Exchange' mode and only from 'Master\_Add'. Under all other instances, the service is ignored. If more than two net data bytes (Control\_Command, Group\_Select) are received (refer to Table 5.2-6) or if there is no indication buffer, DPS also does not accept this service.

7	6	5	4	3	2	1	0	Byte	Name
								0-5:	Buffer Header
Res.	Res.	Sync	Unsync	Freeze	Unfreeze	Clear_ Data	Res.	6:	Control_Command
								7:	Group_Select

**Table 5.2-6: Data Format of the Global\_Control Message**

The parameter Group\_Select establishes which group(s) is(are) to be addressed. The Global\_Control message becomes effective if the bit by bit AND operation of the Group\_Ident, transferred in the Set\_Parameter message, with the Group\_Select parameter supplies a value unequal to 0 on at least one bit position. If Group\_Select is equal to 0, all slaves are addressed.

#### Byte Control\_Command:

Bit 7, 6, 0: Reserved

The designation "Reserved" indicates that these bits are reserved for future function expansions. If such a bit is set, DPS sets 'Diag.Not\_Supported=1', and the "Leave\_Master macro" is executed. However, if the user parameterizes 'Check\_No\_GC\_Reserved=1' in the param register, the Reserved bits are not checked.

Bit 5: Sync

The output data transferred with a Write\_Read\_Data message is changed from 'D' to 'N' (DX\_OUT interrupt is generated). The subsequently transferred output data is kept in 'D' until the next 'Sync' command is made. The same reaction occurs for 'Sync\_Supported=0' as does for a set Reserved bit.

Bit 4: Unsync

The command 'Unsync' cancels the 'Sync' command. In addition, as in the case of 'Sync', the previously transferred output data is changed from 'D' to 'N'.

Bit 3: Freeze

The input data is fetched from 'N' to 'D', and "frozen". New input data will be fetched only if the master sends the next 'Freeze' command. The same reaction occurs for 'Freeze\_Supported=0' as does for a set Reserved bit.

Bit 2: Unfreeze

With 'Unfreeze', freezing the input data is cancelled. In addition, as in the case of 'Freeze', new input data that was made available is fetched from 'N' to 'D'.

Bit 1: Clear\_Data

In operation with Clear buffer, the Dout buffer is not deleted and it is not changed; rather, the mode 'N\_CI=1' is set in the Dout\_Buffer\_SM, and the user interrupt 'DX\_OUT' is generated. If the user then fetches his new Dout data, the C and U buffers are exchanged and the user gets the message 'U\_Buffer\_Cleared'.

In operation without Clear buffer, this command causes a completely normal buffer exchange and interrupt 'DX\_OUT' is generated. If the user fetches his new Dout data from it, he must use the 'req-ssap' byte in the buffer header or the 'U\_Buffer\_State' register to determine whether the data are to be output from the buffer or if the substitute values are to be output instead.

The DDB/DXBout data are not subject to Clear control. The 'Clear' has a higher priority than 'Sync.'

With 'sync', data buffers are made available synchronously. However, this does not provide for synchronous mapping directly to the I/O as is the case with the LSPM2. Although the application is interrupted via the 'DX\_OUT interrupt', the transfer time from the buffer that was made available to the I/O is subject to interrupt latency. To bypass it, the interrupt 'DX\_OUT' can directly be applied to the port PB3 if a global control message is received with 'Sync', provided 'Enable DX\_OUT\_Port=1' was parameterized in the C31\_Control register beforehand. Thus, external HW support, or separate interrupt processing could bring about the transfer from the buffer to the I/O in a fixed time reference.

With 'Freeze', the available Din buffer in 'N' is frozen to 'D'. Thus, in distinction to the LSPM2, no updating is provided at this time from the I/O. To circumvent this, the user would have to make the input data, if it changes, available immediately in the N buffer (high processor capacity required).

For each valid Global\_Control message, the Control\_Command byte is stored in the RAM cell 'GC\_Command'. At initialization, DPS preassigns FFh (not a valid value) to the RAM cell 'GC\_Command'. The user can read and interpret this cell. Depending on the setting of 'New\_GC\_Int mode' (refer to Param Register), the interrupt 'New\_GC\_Command' is generated. With 'New\_GC\_Int mode = 0', the interrupt is generated only if the Control\_Command byte for the last received Global\_Control message has changed. With 'New\_GC\_Int mode=1', the interrupt is generated after each receipt of a GC message.

#### Equidistant Cycle, Sync Clock:

The ASPC2 Step E supports an equidistant cycle, i.e., it begins with the DP cycle exactly on the clock pulse (resolution; approx. 1  $\mu$ s). For closed-loop control systems such as an 'electronic shaft', the DPC31 can generate a Sync clock (clock pulse beater) after receipt of the last bit of the first DP request. This request must be a 'global control frame' in its own group. The Sync clock (low active pulse of 2-3  $\mu$ sec length), is output on Port PB<sub>2</sub> and interrupt 'GC-Clock' is also generated. So that Port PB<sub>2</sub> is isolated for this function, bit 'Enable GC\_Clock=1' must be set in the C31\_Control-Register.

The special group for the Sync clock must be parameterized for the DPC31 by the user ('group clock'). During receipt of **any GC frame** by the DP master (Master\_Add), the DPC31 makes a bit-by-bit AND operation of the second received data byte (Group Select) with 'Group\_Clock'. If a hit occurs, the Sync clock is initiated at the end of the GC frame. The values for the jitter and the delay of the Sync clock are presented in Table 5.2-7 for asynchronous operating mode of the DPC31 and a load of 80 pF at Port PB<sub>2</sub>.

Baud Rate	Delay (after ED of GC frame.)	Jitter (max.)
12 MBd	$T_{\text{BIT}} + 7 T_{48} + 20 \text{ ns}$	$\frac{1}{4} T_{\text{BIT}} + T_{48}$
$\leq 6 \text{ MBd}$	$T_{\text{BIT}} + 7 T_{48} + 20 \text{ ns}$	$\frac{1}{4} T_{\text{BIT}}$

**Table 5.2-7:** Timing Values for Sync Clock

#### Failsafe Mode:

To support the failsafe mode, a 'Spec\_Clear\_Mode' is implemented in the DPC31. The master generates such a Clear mode by sending a Global\_Control message with 'Clear\_Data=1'. The Din data has to continue to be fetched during this Clear\_Mode. For this, the master has to send the Write\_Read\_Data message with the parameterized number of Dout data bytes=00h. However, in the case of many slave applications, the value 00h does not correspond to the Clear mode (for example, substitute values for analog modules). Here, the user generates the corresponding substitute values. If the Global\_Control message was not received because of a bus fault, this slave does not know that it should be in the Clear mode; therefore, the subsequently received Dout data bytes with the value 00h can't be replaced with the substitute values.

To support the failsafe mode, the DPC31 also accepts Write\_Read\_Data messages without output data even though the parameterized Dout length 'Dout\_Buffer length # 0' is set. After the receipt of this message, the C buffer where the substitute values are stored, is then included in the buffer circulation. If the user fetches this buffer in operation with Clear buffer, the display 'U\_Buffer\_Cleared' is set with 'User\_New\_Dout-Cmd' (refer to Table 5.2-4) and the user receives the information that it is cleared data (substitute values).

If the user fetches this buffer in operation without Clear buffer, he receives information with 'req\_ssap' byte in the buffer header or with the 'U\_Buffer\_State' register on whether the data from the data buffer or substitute values are to be output.

### 5.2.7 Read\_Inputs (SAP56)

The Read\_Input message is accepted by the MAC only with request data length = 0, in the mode 'Data\_Exchange', from any master. For this, DPS enters the corresponding validation values in

'SAP56 of the SAP\_SCB'. In the other modes, the DPC31 responds with 'no service activated' (modes 'Wait\_Prm, Wait\_Cfg') or 'no resource' (request data length # 0). The exchange of the Read\_Input buffer has been described previously. Between the initial call and the repetition if there is a buffer change from 'U' -> 'N' -> 'D' (through User\_New\_Din command), the new input data is sent at the repetition.

### 5.2.8 Read\_Outputs (SAP57)

The Read\_Output message is accepted by the MAC only with request data length = 0, in the mode 'Data\_Exchange', from any master. For this, DPS enters the corresponding validation values in 'SAP57 of the SAP\_SCB'. In the other modes, the DPC31 responds with 'no service activated/no resource'.

The exchange of the Read\_Output buffer has been described previously. Between the initial call and the repetition if there is a buffer change from 'N' -> 'U' (through User\_New\_Dout command), the new output data is sent at the repetition.

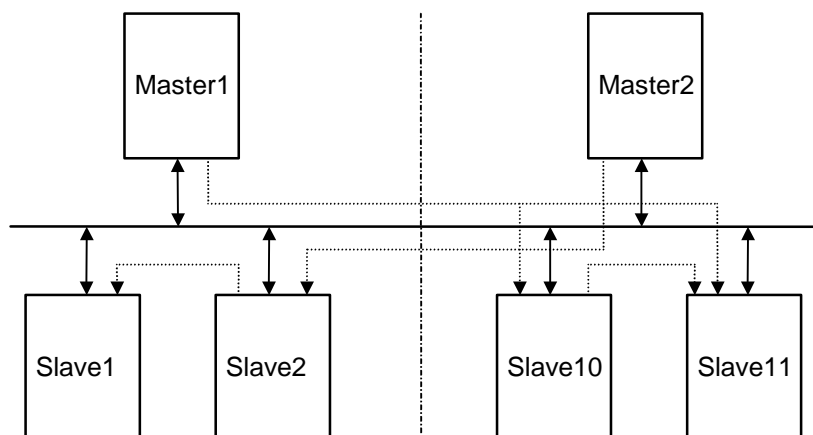
### 5.2.9 Get\_Config (SAP59)

The Get\_Config message is accepted in all modes. If the call message contains request data, the MAC acknowledges with 'no resource'.

### 5.2.10 Data Exchange Broadcast (DXB)

**Requirements for Data Exchange Broadcast:**

- Every DP slave must be able to receive data from another DP node (master or slave) without going through its own master (see Figure 5.2.5).
- Each DP node (master or slave) must be able to send data to another DP slave without going through its own master.
- Up to 8 links per DP slave are possible.



**Figure 5.2-6:** Data Exchange Broadcast between different Masters and Slaves  
**Implementation of Internode Data Communication:**

- The DXB service (Write\_Read\_Data with DXB) enables internode communication from one DP slave (publisher) to several other DP slaves (subscriber) using a single data frame. The master addresses the publisher with a special call service (DXB request) and the publisher responds under the broadcast address. All slaves are thus in the position to receive this response frame.
- Configuration is used to inform each DP slave of which information is defined for it (link buffer). This configuration information is communicated to the node.
- The address relationship under which a DP slave is to receive data from one of these frames is entered in the respective link. In addition, the link includes the data range (continuous) that is to be tapped from the frame.
- The Publisher function applies to all DPS SAPs and the subscriber function applies only to the **I/O data (Default\_SAP under DXB response frame)**

- Publisher mode is enabled with 'EN\_DXB\_Publisher=1' in the Param register. Subscriber mode is only enabled in DPC31 if 'EN\_DP\_DXB=1' was communicated in the Set\_Param frame) or the command 'EN\_DP\_DXB' is issued via the request interface.
- When the DPC31 is used as a Publisher, the master must parameterize a  $T_{\text{MinTSDR}} \geq 35 T_{\text{Bit}}$ .

### Configuration of the Data Exchange Broadcast:

The configuration for the DXB tap can occur with the Set\_User\_Prm-Frame (SAP53) or via DPV1.

#### a) Ext\_User\_Prm-Frame (SAP53):

The Set\_DDB\_Prm-Frame can be locked by the user by setting the 'MAC\_Ext\_User\_Prm-Buffer=00h' during startup. This SAP is then available as a FDL\_SAP. Otherwise, two alternating buffers of the same length are available for this SAP. One buffer is integrated in the SAP-SCB as an Indication buffer (MAC\_Ext\_User\_Prm-Buffer) and the other is attached in the DPS buffer management as a Ext\_User\_Prm-Buffer. The Indication is always transferred to the user in the Ext\_User\_Prm-Buffer.

This service is accepted by DPS in 'Wait\_Cfg' state only and by 'Master\_Add'. In the 'Data\_Exchange' state, this frame involves a 'Leave\_Master'. DPS ignores this call under all other faulty constellations.

The user takes on the evaluation of the Set\_Ext\_User\_Prm-Frame. Once DPS has received a plausible frame, an Indication occurs, i.e., DPS swaps the Indication buffer in the Ext\_User\_Prm-SAP for the Ext\_User\_Prm-Buffer from the DPS buffer management, and a 'New\_Ext\_User\_Prm\_Data-Interrupt' is generated. No reaction occurs at this time during the sequence. The user must check the 'Ext\_User\_Prm\_Daten' and provide either a positive or negative acknowledgement (see above).

The 'Ext\_User\_Prm\_Ok-Cmd/Ext\_User\_Prm\_Not\_Ok-Cmd' acknowledgements are read accesses to defined register cells with the corresponding messages 'Not\_Allowed', 'Ext\_User\_Prm\_Finished', or 'Ext\_User\_Prm\_Conflict'.

#### Ext\_User\_Prm\_Ok-Cmd (Read Operation):

0	0	0	0	0	0	User_Ack <sub>1</sub>	User_Ack <sub>0</sub>
---	---	---	---	---	---	-----------------------	-----------------------

User\_Ack<sub>1..0</sub> = 00 ⇒ Ext\_User\_Prm\_Finished

User\_Ack<sub>1..0</sub> = 01 ⇒ Ext\_User\_Prm\_Conflict

User\_Ack<sub>1..0</sub> = 11 ⇒ Not\_Allowed

User\_Ack<sub>1..0</sub> = 10 ⇒ not possible

#### Ext\_User\_Prm\_Not\_Ok-Cmd (Read Operation):

0	0	0	0	0	0	User_Ack <sub>1</sub>	User_Ack <sub>0</sub>
---	---	---	---	---	---	-----------------------	-----------------------

User\_Ack<sub>1..0</sub> = 00 ⇒ Ext\_User\_Prm\_Finished

User\_Ack<sub>1..0</sub> = 01 ⇒ Ext\_User\_Prm\_Conflict

User\_Ack<sub>1..0</sub> = 11 ⇒ Not\_Allowed

User\_Ack<sub>1..0</sub> = 10 ⇒ not possible

#### Table 5.2-8: Coding of Ext\_User\_Prm\_(Not)\_Ok-Cmd

DPS memorizes the current 'Master\_Add' at each trigger. When the DPS interrupts are processed, DPS first checks to determine whether the current 'Master\_Add' still corresponds to the previously stored 'Master\_Add'. If not, a different master has taken over in the meantime and, thus, events are rejected. Otherwise, DPS evaluates the acknowledgements.

**If during operation the 'New\_Prm\_Data', 'New\_Ext\_User\_Prm\_Data' und 'New\_Cfg\_Data' interrupts are pending simultaneously for the user, he must adhere to the following acknowledgement order: Set\_Param, Set\_Ext\_User\_Param, and then Check\_Config.**

A reset is also performed during startup of DPS, thus once the user has transferred 'MAC\_Start' to the request list, or the watchdog has expired in 'DP\_Control' state.

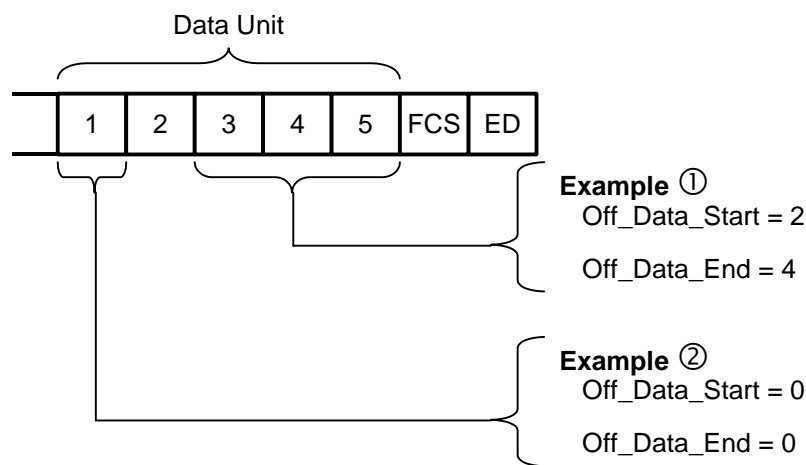
#### b) DPV1 Configuration:

The Set\_Ext\_User\_Prm frame **must** be locked by the user by setting the 'MAC\_Ext\_User\_Prm buffer=00h' during startup. The user communicates the links (taps) to DPS in the Ext\_User\_Prm

buffer. In addition, the MAC must still be enabled for the internode communication. This occurs via the request 'DXB\_Enabled/Disabled' in the Instruction\_Queue. This same request can be used at any time to end internode communication. In addition, internode communication mode is tied to DPS. This mode is permitted only in 'Data\_Exchange' state. If DPS leaves 'Data\_Exchange' state, internode communication will be disabled simultaneously.

**Link Buffer (DXB\_Buffer):**

The link buffer is the Ext\_User\_Prm buffer that was received and checked by the user. One to eight entries can be made in this buffer. The buffer end is defined based on an 'End Character = FFh'. When a DXB response frame is received, the MAC makes this frame plausible with the link entries. In so doing, the received DA and SA addresses are compared to the corresponding entries. In addition, the received net data length must correspond to the value assigned in 'DXB\_Data length'. Because each DP master can also be a DP slave, the frame type (Request or Response) must also still be made plausible. If the received frame is now plausible with a link entry, the preset data area (tap), which is specified by two offset pointers (start: Off\_Data\_Start, End: Off\_Data\_End), is tapped and placed in the appropriate D buffer. Three alternating buffers apiece are available for each configured link. Of these, the D buffer is integrated in the DXB\_SCB as a DXBn\_D-buffer.



**Figure 5.2-7:** Programming of 'Off\_Data\_Start' and 'Off\_Data\_End'

	Entries	Coding	Comments
<b>Cntl</b>	Header (6 Bytes)		see Section 5.1
<b>Link0:</b>	Destination Address Source Address DXB_Data Length  Check byte  Off_Data_Start Off_Data_End	01111111b 0xxxxxxb xxxxxxb  0000000b (Response) 0100000b (Request) xxxxxxb xxxxxxb	Compared to the received DA Compared to the received SA Must correspond to the net data length of the tapped frame! Compared to the masked frame type bit of the received FC Points to the first byte that is to be tapped Points to the last byte that is to be tapped
.....			
<b>Link7:</b>	Destination Address Source Address DXB_Data Length Check byte  Off_Data_Start Off_Data_End	01111111b 0xxxxxxb xxxxxxb  0000000b (Response) 0100000b (Request) xxxxxxb xxxxxxb	

<b>End</b>	End Character	11111111b	End coding of link buffer
------------	---------------	-----------	---------------------------

Table 5.2-9: Structure of Link Buffer

**User\_New\_DXB0-7 Cmd** (Read Operation):

0	0	0	0	0	U_Buffer_State	U_Buffer <sub>1</sub>	U_Buffer <sub>0</sub>
---	---	---	---	---	----------------	-----------------------	-----------------------

U\_Buffer\_State = 0 ⇒ No new U\_Buffer    U\_Buffer<sub>1..0</sub> = 00 ⇒ Buffer1

U\_Buffer\_State = 1 ⇒ New U\_Buffer    U\_Buffer<sub>1..0</sub> = 01 ⇒ Buffer2

U\_Buffer<sub>1..0</sub> = 10 ⇒ Buffer3

Table 5.2-10: Coding of User\_New\_DXB-Cmd

### Timeout Monitoring of DXB Connections:

For timeout monitoring of the DXB connections, the watchdog must be operated in 'DP\_Control' state. For this purpose, the master must transfer a valid watchdog value with 'WD\_On=1' in the Set\_Param frame.

All configured links must have sent data at least once in a watchdog cycle. The links are hereby registered in a 'Score\_Actual\_Register'. When DXB mode is disconnected, DPS deletes this Score\_Actual\_Register. Each signaled link is arranged in the appropriate bit position in the Score\_Actual\_Register (Bit 0 to 7) with a "log. 1" according to its entry (0 to 7) in the link buffer. If the watchdog has reached its parameterized value, DPS compares the Score\_Actual\_Register to the 'Score\_Register', which contains the received links of the last turn. If no difference is detected, all of the links arrive and the Score\_Actual\_Register is deleted for the next pass.

If a difference is detected, DPS transfers the Score\_Actual\_Register to the Score\_Register, then deletes the Score\_Actual\_Register, and generates the 'DXB\_Error-Interrupt'. As a result, the user receives information on failed or returned links in the new Score\_Register. This state is now taken as the basis for the subsequent watchdog cycle. If, for example, a failed link comes up again, a difference between the Score\_Actual\_Register and the Score\_Register arises again at the end of the cycle, and the 'DXB\_Error' interrupt is generated again.

If the user preassigns the Score\_Register with configured DXB connections for each disconnection of DXB mode, monitoring for 'failed links' starts. When '00h' is preassigned, all received links are first registered one time in the Score-Register following a turn, and consulted as monitoring for the next turn.

## 5.3 Additional FDL Functions

### 5.3.1 FDL Monitoring Timer

For monitoring a connection, a 16-bit monitoring timer (FDL\_Timer\_Count-H/L) is available for each FDL\_SAP. This timer is operated with a cycle clock of 10 ms (1 ms for WD\_Test=1). This enables implementation of monitoring times up to 655 sec. Each timer can be individually controlled via the FDL\_Timer control.

Table 5.3-1 gives the assignment FDL\_Timer control. With 'Timer\_Enable=1', the FDL\_Timer is enabled. After each elapsed time interval (10 ms), the timer (FDL\_Timer\_Count H/L) is decremented. If it reaches a value of '0000h', the reaction assigned in 'Timer\_Action<sub>1..0</sub>' is executed and the timer is disabled. The following reactions are defined:

- No reaction
- Lock SAP 'req\_sa = 7Fh'
- Remove access protection 'req\_sa = FFh (all)'

Timer_State <sub>3</sub>	Timer_State <sub>2</sub>	Timer_State <sub>1</sub>	Timer_State <sub>0</sub>	0	Timer_Action <sub>1</sub>	Timer_Action <sub>0</sub>	Timer_Enable
7							0

Timer\_State<sub>3..0</sub> = User defined

Timer\_Action<sub>1..0</sub> = 00 ⇒ nothing

Timer\_Enable = 0 ⇒ disable

Timer\_Action<sub>1..0</sub> = 01 ⇒ req\_sa = '7Fh' Timer\_Enable = 1 ⇒ enable  
 Timer\_Action<sub>1..0</sub> = 10 ⇒ req\_sa = 'FFh'  
 Timer\_Action<sub>1..0</sub> = 11 ⇒ Reserved

**Table 5.3-1:** Assignment of FDL\_Timer Control

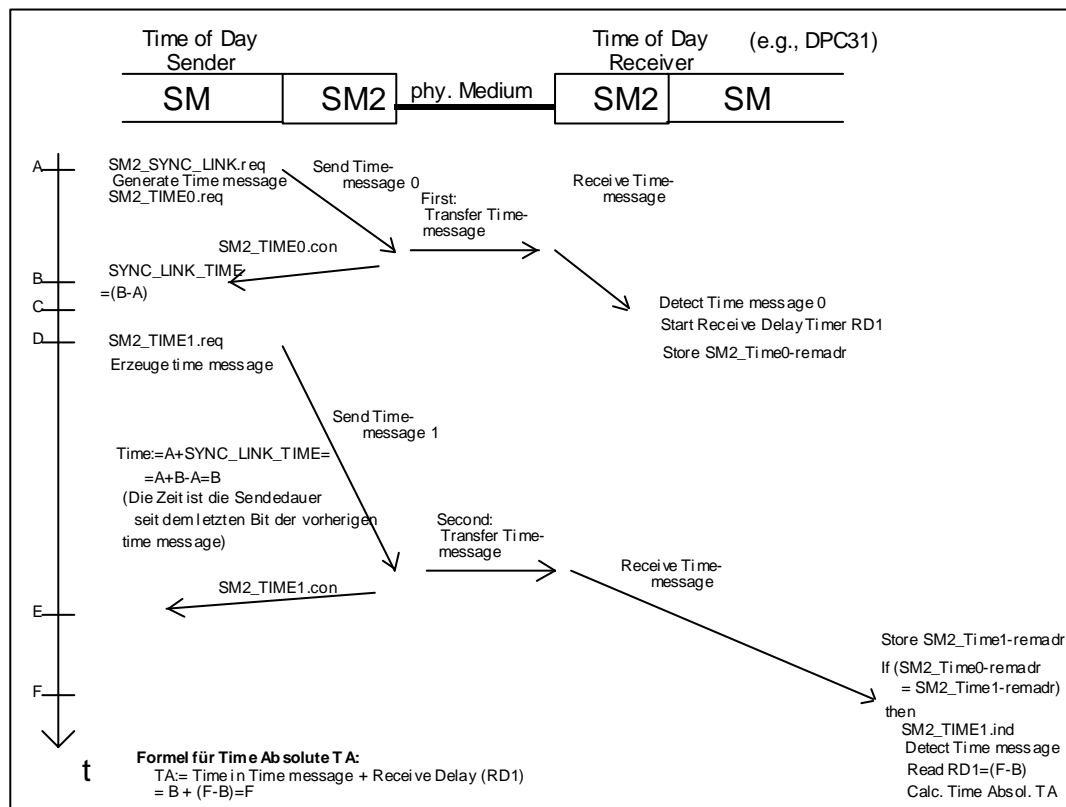
In addition, when an FDL\_Timer elapses, an entry is made in the Indication queue. The SAP\_No. in which the FDL\_Timer elapses and the FDL\_Timer control is thereby transferred to the user. The user recognizes the state of the connection monitoring in which the FDL\_Timer elapses via the 'Timer\_State<sub>1..0</sub>' (user defined, not modified by DPC31).

The FDL\_Timer can be started, stopped, or reloaded on a sap-specific basis using the request 'FDL\_Timer\_Change'. The new 'FDL\_Timer Control' and the Reload value 'FDL\_Timer\_Count-H/L' is always transferred with this request.

The FDL\_Timebase Timer is an internal free-running counter that generates an interrupt after every time interval (10 ms). All FDL-SAPs are then scanned and the enabled FDL\_Monitoring timer is decremented. **Only SAPs 0 - 54d are considered.**

### 5.3.2 Time of Day Synchronization

Figure 5.3-1 presents the main sequence. However, this service is only enabled with 'En\_Clock\_Sync=1' in the Param Register. Otherwise, the SM\_Time0/1 frames are filtered by the MAC.



**Figure 5.3-1:** Sequence of Time of Day Synchronization

A time of day master first sends the SM2\_Time0 frame (broadcast, FC=40h). If the SM\_Time-SAP is enabled in the SAP\_SCB, the delay timer is reloaded and started in the DPC31 directly after the receipt of SM2\_Time0<sup>4</sup>, and the receipt is indexed. In addition, the MAC saves the Remote-Adr of this Time of Day\_Master. With the SM2\_Time1 frame (Broadcast, FC=C0h), the Time of Day\_Master now communicates the time at which it sent the SM2\_Time0 frame (time of send completion) and, thus, the time at which the DPC31 started its delay timer. Upon receipt, the MAC checks to determine whether SM2\_Time1 has come from the same Time of Day\_Master as SM2\_Time0. If so, the MAC enters this Indication in the Indication queue and goes to 'Wait\_New\_Enable' state. Otherwise, this frame is ignored.

<sup>4</sup> max. 280ns nach ED-Ende

If the user now processes the indication, he must stop and read out the delay timer (see Mode Register1). The real time of day is now calculated by adding the received time of day to the value in the delay timer. Then, the user must enable another time of day synchronization. For this purpose, the command 'En\_New\_SM\_Time-Sync' must be transferred via the request interface.

The delay timer is a 16-bit timer with a resolution of 1 $\mu$ s. This enables implementation of a time of 65.5 ms. If an overflow occurs, the 'Delay\_Timer-Overrun' interrupt is generated and the timer starts over.

If the MAC receives another SM2\_Time0 frame from the same time of day master, a fault occurs. For example, it is possible that the SM2\_Time1 frame has been lost due to a bus malfunction. The 'Ignore\_Delay\_Timer\_Overrun' interrupt is thereby generated to the user. The user must then stop the delay timer and enable a new synchronization with the 'En\_New\_SM\_Time-Sync' request.

### 5.3.2.1 Support of MSACS1 SAPs

To support Master Slave Acyclic Class 1, opening and closing of SAPs 50, 51, and 52d can be coupled to the cyclic machine. To do so, the user must set Bit 'Check\_DPV1\_Enable=1' in the Param Register. If the DPC31 is now assigned in DPV1 mode (DPV1-Enable=1 in the Set\_Param frame, DPV1\_Status\_1, Bit 7) and this frame contains at least 10 data bytes, DPS opens these SAPs for the DP master (Master\_Add) upon entrance in 'Data\_Exchange'. No message is sent to the user. When the 'Data\_Ex' state is left, these SAPs are again locked and any buffered Indications are deleted (Buffer\_In\_Use=0). However, the buffer control (Buffer\_State, Response in Buffer\_Control Byte) is not changed. In addition, an Indication is made to the user, 'Stop\_C1-Ind' in the Indication\_Queue. The user must then readjust the buffers of SAPs 50-52 and confirm this to the DPC31. 'Stop\_C1-Con' in the Instruction\_Queue. Return to 'Data\_Ex' is only possible after this confirmation!

If the acyclic machine does not require all of the SAPs (50-52), the user must lock them in the SAP\_List (SAPx\_Ptr=FFh). This means they will not be controlled by the DPC31 either.!

'Check\_En\_Prm\_Cmd=1' in the Param Register enables the check of 'Enable\_Prm\_Cmd' in the Set\_Prm frame (DPV1\_Status\_3, Bit 7). However, this only occurs in the 'Data\_Ex' state and 'Operation-Mode=V1' when a new Prm frame (V1) is received from the same master. If 'Enable\_Prm\_Cmd=1', DPS remains in the 'Data\_Ex' state. When 'Enable\_Prm\_Cmd=0' a Leave\_Master is executed.

## 6 User Functions on the C31 Controller

The DPC31 contains an integrated C31 core that is available entirely for user functions. One of the two external interrupts (XINT0) is already being used for interfacing the communication component and is therefore no longer available to the application.

## 7 Description of the Hardware Blocks

### 7.1 Universal Processor Interface

The DPC31 has a parallel 8-bit interface with a 13-bit address bus. It supports all 8-bit processors and micro-controllers as follows: 80C31/32 by Intel and the Motorola HC11 family. It also supports the 8/16 bit processors and micro-controllers of the 80C166 family by Siemens, X86 by Intel and the HC16/HC916 family by Motorola.

In addition, a clock pulse scaler is integrated which makes the internal work clock pulse (divided by 2 (pin CLKOUT1X2) or 4 (pin CLKOUT1X4) available as system clocks in order to be able to connect a slower controller without additional effort in a lowcost application (refer to Chapter 7.8.1). Both clock outputs can be switched off separately via Mode Register1. For asynchronous physics, the DPC31 is wired to a quartz of 12MHz (XTAL1\_CLK, XTAL2). An integrated PLL generates the internally needed work clock pulse (48MHz: refer to Chapter 7.8.1). In the case of synchronous physics, the DPC31 can be operated in a mode that is particularly low in power loss. This can be achieved only

for low clock pulse rates. The PLL is switched off in this case ( $XPLLEN = VDD$ ) and the variable supply clock pulse of (2), 4, 8, or 16 MHz is applied directly to XTAL1\_CLK.

### 7.1.1 Bus Interface Unit (BIU)

The BIU is the interface to the connected processor/microcontroller. It allows the CPU accesses to the internal 5.5kByte dual port RAM and the registers. It is a synchronous or asynchronous 8-Bit interface with a 13-Bit address bus. The interface can be configured via 3 bus type pins (BusType2..0) (refer to Table 7.1-1). With it, the connected processor family (Intel/Motorola bus control signals such as XWR, XRD, and R\_W, the – data format) and the synchronous (rigid) or asynchronous bus timing is specified.

Figure 7.1-1, Figure 7.1-2, Figure 7.1-3, and Figure 7.1-4 show different Intel and Motorola system configurations. In the C31 mode, the internal address latch and the integrated decoder must be used. In Figure 7.1-1, the minimum configuration of a system with external  $\mu P$  and DPC31 is shown; the chip is connected to an EPROM version of the controller. In terms of additional components, only a quartz crystal is needed in this configuration. If a controller is to be used without integrated program memory, the addresses have to be latched additionally for the external memory (refer to Figure 7.1-2). The connection diagram in Figure 7.1-3 applies to all Intel/Siemens processors that offer asynchronous bus timing and interpret the Ready signal.

#### Notes:

If the **DPC31 is connected to an 80286** or something similar, it is to be taken into account that the processor accesses words; that is, either a swapper is needed that switches, during reading, the corresponding characters from the DPC31 to the corresponding byte position of the 16-Bit data bus. Otherwise the least significant address bit is not connected and the 80286 must make word accesses and correspondingly only interpret the lower byte as shown in Figure 7.1-3.

#### Wiring of the C31-Core Port-Pins for Unused C31-Core:

Set the pin DBX to VDD, then the internal C31 is in permanent reset.  
Ports A, B, C and D except for port D(2) must each be wired with a pull-up resistor.

BusType <sub>2..0</sub>	The DPC31 Processor Interface supports the following micro-controllers:
<p>0 1 1 (synchronous Motorola)</p>	<p>MOTOROLA micro-controller with the following features:</p> <ul style="list-style-type: none"> <li>- Synchronous (rigid) bus; timing without evaluation of XDSACK (PH<sub>2</sub>)</li> <li>- 8-Bit non-multiplexed bus: DB<sub>7..0</sub> (PE<sub>7..0</sub>), AB<sub>12..0</sub> (PG<sub>4..0</sub>, PF<sub>7..0</sub>)</li> </ul> <p>The following can be connected :</p> <ul style="list-style-type: none"> <li>- HC11- types: K, N, M and F1</li> <li>- HC16- and HC916- types with programmable ECLK timing</li> <li>- For all other HC11-types with a multiplexed bus, the addresses AB<sub>7..0</sub> have to be selected externally from the data DB<sub>7..0</sub>.</li> </ul> <p>Address decoder is switched off in the DPC31; CS-signal is supplied from the outside:</p> <ul style="list-style-type: none"> <li>- For micro-controllers with chip select logic: K, F1, HC16, HC916, the chip selection signals can be programmed regarding the address area, priority, polarity, and the window width in the write and read cycle.</li> <li>- For micro-controllers without chip selection logic: N, M and others, an external chip select logic is needed. This means additional HW effort and fixed assignments.</li> </ul> <p>Condition:</p> <ul style="list-style-type: none"> <li>- The DPC31 output clock (CLKOUT1X2/4) has to be at least four times larger than the E Clock. The DPC31 clock (48MHz) has to be at least ten times larger than the desired system clock (E Clock). Pin CLKOUT1X4 is to be wired with this (E_Clock = 3MHz at 48MHz DPC31 clock).</li> </ul>
<p>0 1 0 (asynchronous Motorola)</p>	<p>MOTOROLA micro-controller with the following features:</p> <ul style="list-style-type: none"> <li>- Asynchronous bus; timing with evaluation of XDSACK (PH<sub>2</sub>)</li> <li>- 8-Bit non-multiplexed bus: DB<sub>7..0</sub> (PE<sub>7..0</sub>); AB<sub>12..0</sub> (PG<sub>4..0</sub>, PF<sub>7..0</sub>)</li> </ul> <p>The following can be connected:</p> <ul style="list-style-type: none"> <li>- HC16 and HC916 types</li> </ul> <p>Address decoder in the DPC31 is switched off; CS signal is applied from the outside</p> <ul style="list-style-type: none"> <li>- Chip-select signals are present in all micro-controllers and can be programmed.</li> </ul>
<p>0 0 1 (synchronous Intel)</p>	<p>INTEL, CPU Basis 80C31/32, micro-controllers of various manufacturers:</p> <ul style="list-style-type: none"> <li>- Synchronous (rigid) bus timing without XRDY (PH<sub>2</sub>) evaluation</li> <li>- 8-Bit multiplexed bus ADB<sub>7..0</sub> (PE<sub>7..0</sub>),</li> </ul> <p>The following can be connected:</p> <ul style="list-style-type: none"> <li>- Micro-controller families, such as INTEL, SIEMENS, PHILIPS ...</li> </ul> <p>Address decoder is switched on in the DPC31; CS signal is generated internally:</p> <ul style="list-style-type: none"> <li>- The lower address bits AB<sub>7..0</sub> are stored with the ALE signal in an internal address latch. In the DPC31, the internal CS decoder is activated that generates its own signal from the addresses AB<sub>12..0</sub>.</li> <li>- The integrated address decoder is permanently wired, so that the DPC31 Always has to be addressed under the fixed addresses AB<sub>7..0</sub>=000xxxxb, Whereby the DPC31 selects the corresponding address window from the Signals AB<sub>4..0</sub>.</li> <li>- In this mode, the CS pin (PG<sub>6</sub>) has to be on VDD (high potential)</li> </ul> <p>Wiring: refer to Figure 7.1-1, Figure 7.1-2. Apply ADB<sub>7..0</sub> to DPC31-Pin PE<sub>7..0</sub>, AB<sub>15-8</sub> to DPC31-Pin PF<sub>7..0</sub>, and the DPC31-Pin PG<sub>4..0</sub> to VSS.</p>
<p>0 0 0 (asynchronous Intel)</p>	<p>INTEL and SIEMENS 16/8-Bit micro-controller families</p> <ul style="list-style-type: none"> <li>- Asynchronous bus; timing with evaluation of XRDY (PH<sub>2</sub>)</li> <li>- 8-Bit non-multiplexed bus: DB<sub>7..0</sub> (PE<sub>7..0</sub>); AB<sub>12..0</sub> (PG<sub>4..0</sub>, PF<sub>7..0</sub>)</li> </ul> <p>The following can be connected:</p> <ul style="list-style-type: none"> <li>- Micro-controller families; for example, SIEMENS, 80C16x and INTEL X86</li> </ul> <p>Address decoder in DPC31 is switched off; CS signal is applied from the outside</p> <ul style="list-style-type: none"> <li>- External address decoding is always required</li> <li>- External chip selection logic, if not available in micro-controller.</li> </ul>

**Table 7.1-1**The Different Configurations of the Processor Interface

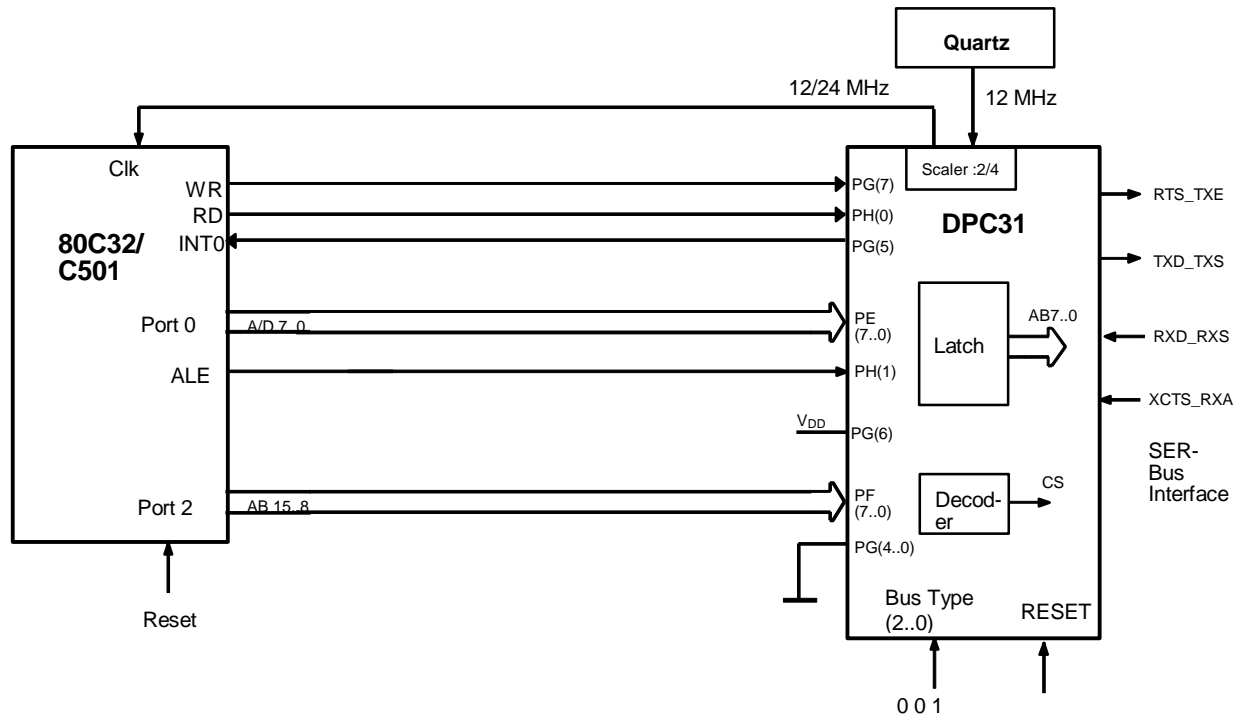


Figure 7.1-1: Low Cost System (C31 Mode)

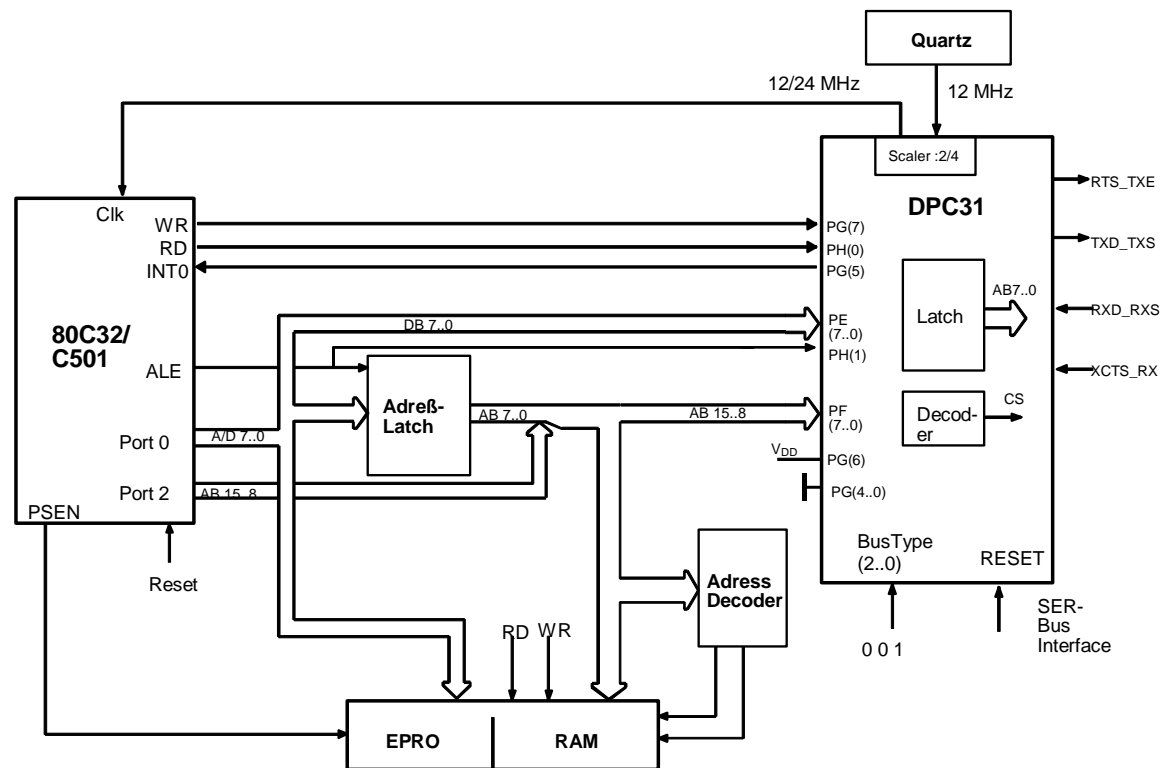


Figure 7.1-2: C31 System with External Memory (C31 Mode)

CS

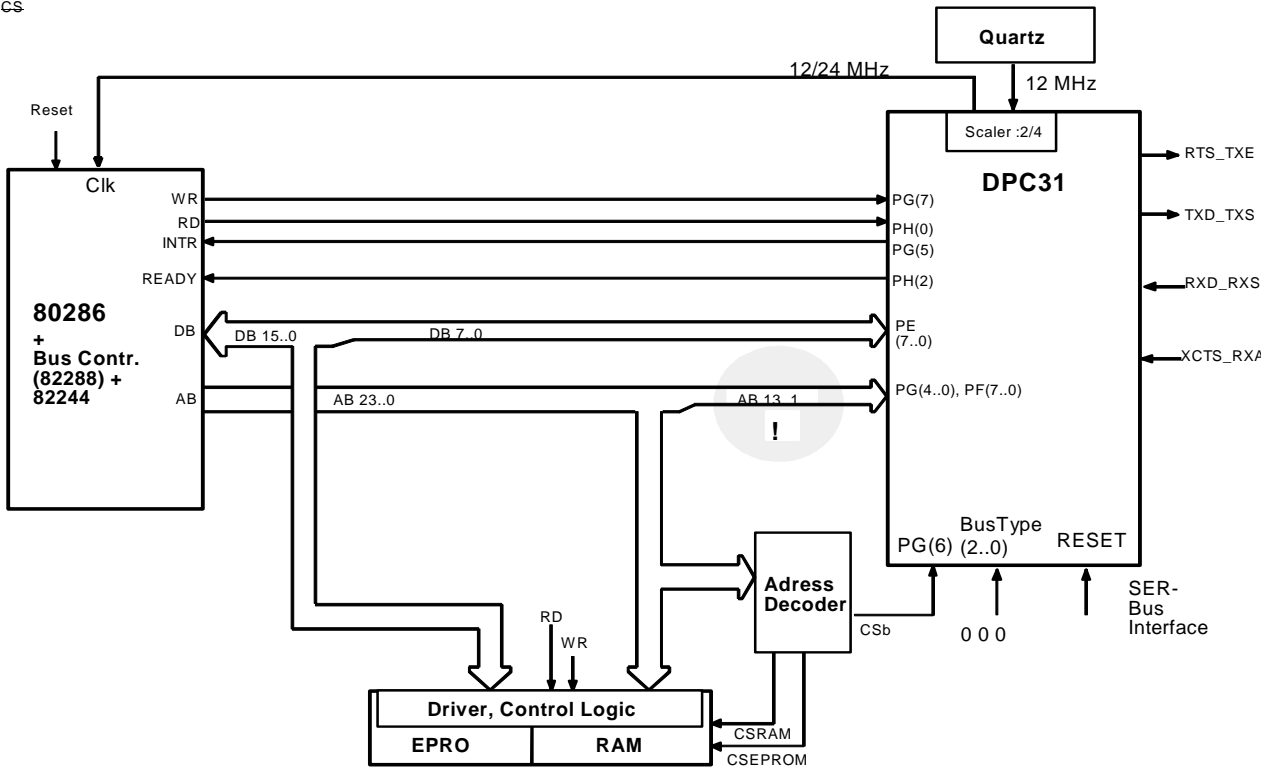


Figure 7.1-3: 80286 System as an Example for Mode X86

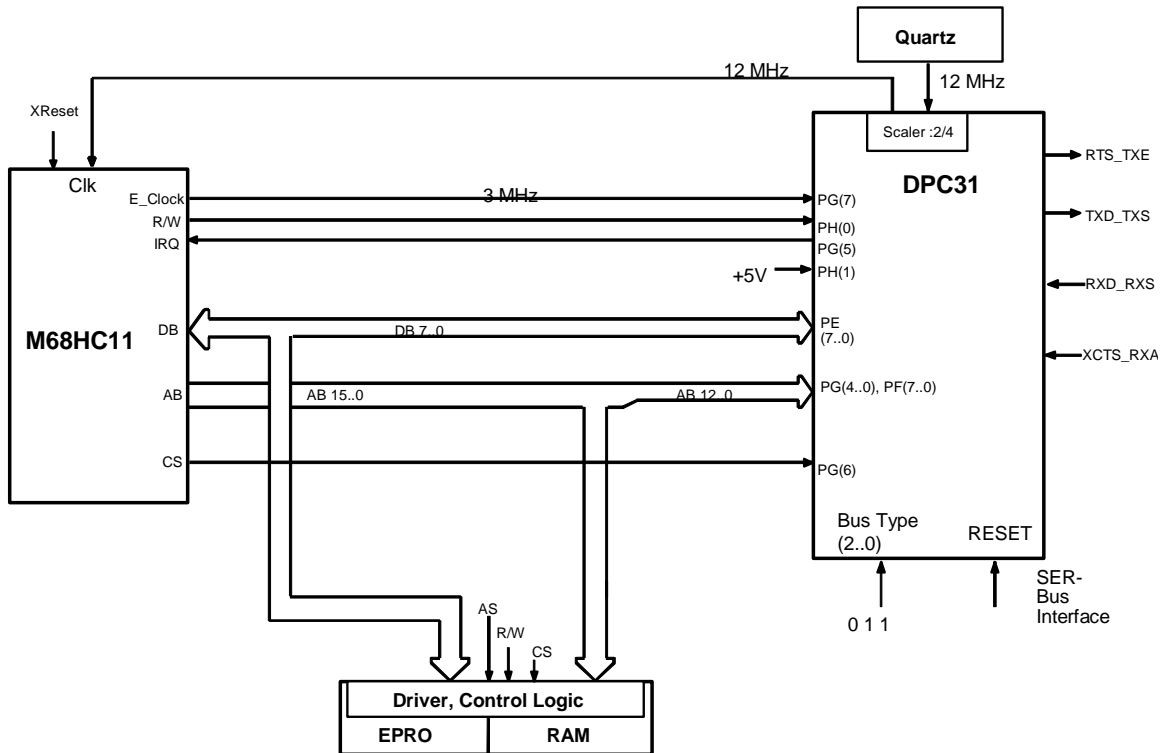


Figure 7.1-4: M68HC11 System as an Example for Synchronous Motorola Mode

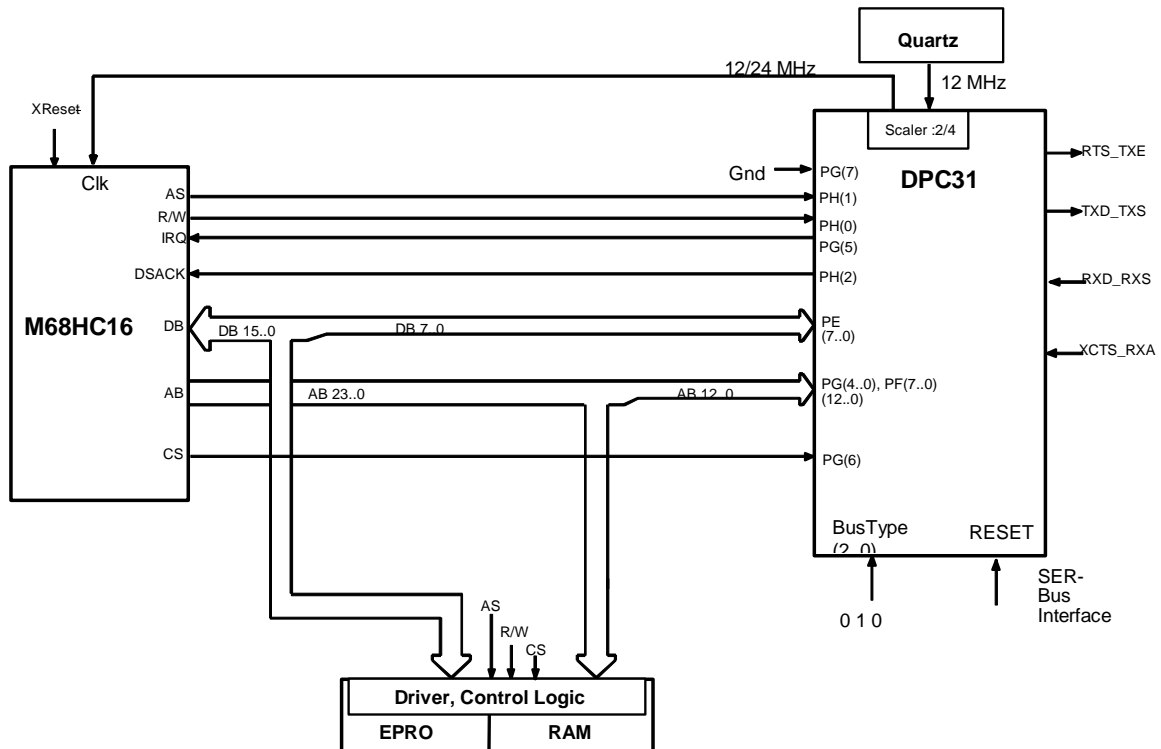


Figure 7.1-5: M68HC16 System as an Example for Asynchronous Motorola Mode

### 7.1.2 IO Interface

If the DPC31 is to be operated without external processor, an I/O interface is available instead of the processor interface (can be set via the bus type pins). This I/O interface consists of four ports (PE<sub>7..0</sub>, PF<sub>7..0</sub>, PG<sub>7..0</sub>, PH<sub>2..0</sub>). Each port bit can be configured as input or output by the internal application (C31). The outputs can be addressed bit by bit as well as byte by byte. Reading is always byte by byte. To configure the I/O bits, each port has a Direction Register (Dir\_Reg). The output status is kept in a register bit (refer to Table 7.1-2). After reset, all ports are switched to input. The addressing of these I/O ports is provided in Chapter 7.3.2.

BusType <sub>2..0</sub>	PH <sub>2..0</sub>	PG <sub>7..0</sub>	PF <sub>7..0</sub>	PE <sub>7..0</sub>
1 - - (I/O Interface)	<b>Dir_Reg_H<sub>2..0</sub></b> (0=Out;1=In) <b>Addresses:</b> Adr_H <sub>2..0</sub> =ByteAddress Adr_H <sub>0</sub> =BitAddress Adr_H <sub>1</sub> =BitAddress Adr_H <sub>2</sub> =BitAddress	<b>Dir_Reg_G<sub>7..0</sub></b> (0=Out;1=In) <b>Addresses:</b> Adr_G <sub>7..0</sub> =ByteAddress Adr_G <sub>0</sub> =BitAddress Adr_G <sub>1</sub> =BitAddress Adr_G <sub>2</sub> =BitAddress Adr_G <sub>3</sub> =BitAddress Adr_G <sub>4</sub> =BitAddress Adr_G <sub>5</sub> =BitAddress Adr_G <sub>6</sub> =BitAddress Adr_G <sub>7</sub> =BitAddress	<b>Dir_Reg_F<sub>7..0</sub></b> (0=Out;1=In) <b>Addresses:</b> Adr_F <sub>7..0</sub> =ByteAddress Adr_F <sub>0</sub> =BitAddress Adr_F <sub>1</sub> =BitAddress Adr_F <sub>2</sub> =BitAddress Adr_F <sub>3</sub> =BitAddress Adr_F <sub>4</sub> =BitAddress Adr_F <sub>5</sub> =BitAddress Adr_F <sub>6</sub> =BitAddress Adr_F <sub>7</sub> =BitAddress	<b>Dir_Reg_E<sub>7..0</sub></b> (0=Out;1=In) <b>Addresses:</b> Adr_E <sub>7..0</sub> =ByteAddress Adr_E <sub>0</sub> =BitAddress Adr_E <sub>1</sub> =BitAddress Adr_E <sub>2</sub> =BitAddress Adr_E <sub>3</sub> =BitAddress Adr_E <sub>4</sub> =BitAddress Adr_E <sub>5</sub> =BitAddress Adr_E <sub>6</sub> =BitAddress Adr_E <sub>7</sub> =BitAddress

Table 7.1-2: IO Interface on the Processor Interface

### 7.1.3 Interface Signals

Pin Name	Signal Names						IO Interface		Comment
	µP Interface				I/O				
	Intel sync.	Intel async.	Motorol. sync.	Motorol. async.					
PE <sub>7..0</sub>	DB <sub>7..0</sub> / AB <sub>7..0</sub>	DB <sub>7..0</sub>	DB <sub>7..0</sub>	DB <sub>7..0</sub>	I/O	PE <sub>7..0</sub>	I/O	high-resistance at reset	
PF <sub>7..0</sub>	AB <sub>15..8</sub>	AB <sub>8..1</sub>	AB <sub>7..0</sub>	AB <sub>7..0</sub>	I	PF <sub>7..0</sub>	I/O		
PG <sub>4..0</sub>	GND	AB <sub>13..9</sub>	AB <sub>12..8</sub>	AB <sub>12..8</sub>	I	PG <sub>4..0</sub>	I/O		
PG <sub>5</sub>	X/INT	X/INT	X/INT	X/INT	O	PG <sub>5</sub>	I/O	Interrupt, polarity can be parameterized	
PG <sub>6</sub>	V <sub>DD</sub>	XCS	XCS	XCS	I	PG <sub>6</sub>	I/O	Chip-select	
PG <sub>7</sub>	XWR	XWR	E-Clock	GND	I	PG <sub>7</sub>	I/O	Intel: Write / Motorola: E-Clock	
PH <sub>0</sub>	XRD	XRD	R_W	R_W	I	PH <sub>0</sub>	I/O	Intel: Read / Motorola: Read/Write	
PH <sub>1</sub>	ALE	V <sub>DD</sub>	V <sub>DD</sub>	AS	I	PH <sub>1</sub>	I/O	Address Latch Enable	
PH <sub>2</sub>	-	XRDY	-	XDSACK	O	PH <sub>2</sub>	I/O	Ready Signal	
BUSTYP <sub>2..0</sub>	"001"	"000"	"011"	"010"	I	"1 - -"	I	Setting of the interface	
RESET	RESET	RESET	RESET	RESET	I	RESET	I	Reset input	

**Table 7.1-3:** Interface Signals for µP and IO Interface

The data bus outputs are high-resistance during the reset phase. In the test mode, all outputs are switched to high resistance.

#### 7.1.4 Interrupt Controller of the µP Interface in the DPC31

Via the interrupt controller, the processor is informed of various events. These consist primarily of indication messages and different error events. The controller has no prioritization level and does not provide an interrupt vector (not compatible with 8259A).

It consists of the following: an interrupt request register (IRR), interrupt mask register (IMR), interrupt register (IR) and an interrupt acknowledge register (IAR). The structure is shown in Figure 7.1-6.

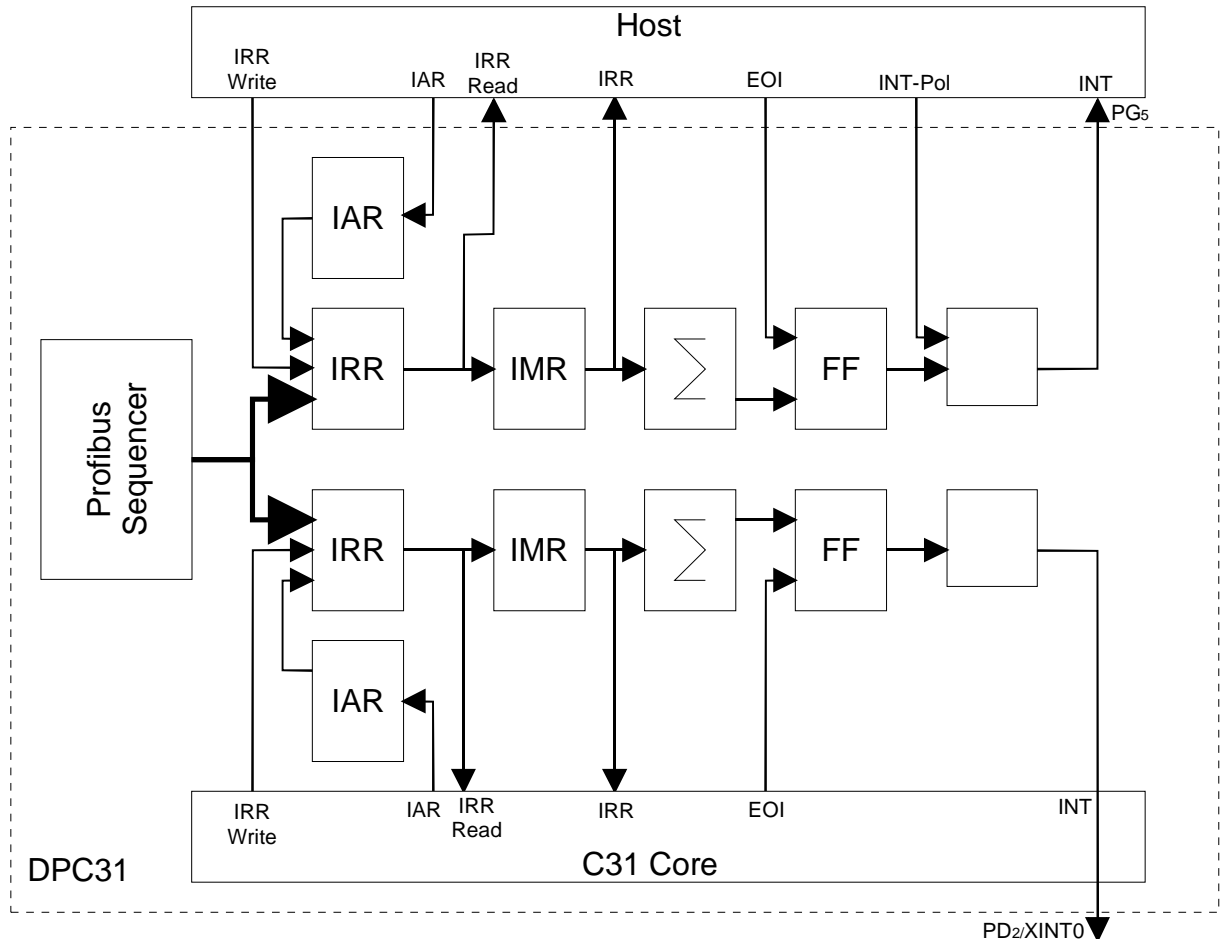
In the IRR, every event is stored. Via the IMR, individual events can be suppressed. If, for example, the DPS indications are evaluated only by the internal C31, the corresponding masks have to be set here and enabled for the C31 in the interrupt controller. The entry in the IRR is independent of the interrupt mask. Events that are not masked out in the IMR generate the **X/INT Interrupt** (Pin PG<sub>5</sub>) via a cumulative network.

For debugging, the user can set every event in the IRR (only those bits are activated that are to be set).

Each interrupt event that was processed by the processor has to be cleared via the IAR (except for New\_Prm\_Data, New\_DDB\_Prm\_Data, New\_Cfg\_Data). A log '1' is to be written to the corresponding bit position. If a new event and an acknowledgement of the previous event are pending at the same time at the IRR, the event remains stored. If the processor subsequently enables a mask, it has to be ensured that there is no past entry in the IRR. To make sure, the position must be cleared in the IRR prior to the mask enable.

Prior to exiting the interrupt routine, the processor has to set the **"End of Interrupt Signal (EOI) = 1"** in the EOI register (see below). With this edge change, the interrupt line is switched

inactive. If an event should still be stored, the interrupt output becomes active again only after an interrupt inactive time of at least 1µs or 1ms, or at most 2µs or 2ms (refer to Chapter 9.6.2.2). Via 'EOI\_Timebase' (Param Register, refer to Chapter 3.3), this interrupt inactive time can be set (EOI\_Timebase=0 -> 1µs; EOI\_Timebase=1 -> 1ms). This makes it possible to reenter the interrupt routine when using an edge-triggered interrupt input.



**Figure 7.1-6: Interrupt Controller of the µP Interface and C31 Core in the DPC31**

The polarity of the interrupt input can be parameterized (Mode Register1; refer to Chapter 3.3: INT\_Pol). After the HW reset, the output is low-active.

**Interrupt Request Register, IRR (writable, readable):**

New_GC_Command	Go/Leave_Data_Exchange	IndQ_Full	IndQ_Entry	Ignore_Del_Timer_Overrun	Delay_Time_r_Overrun	Diag_Fetched	WD_State_Changed
7							0
DX_OUT_Overflow	DX_OUT	Diag_Buffer_Changed	Get_Cfg_Buffer_Changed	New_Cfg_Data	New_DDB_Prm_Data	New_Prm_Data	New_SSA_Data
15							8
DXB7_OUT	DXB6_OUT	DXB5_OUT	DXB4_OUT	DXB3_OUT	DXB2_OUT	DXB1_OUT	DXB0_OUT
23							16

0	0	DX-Entered	GC_Clock	0	SSC_Interface	RAM_Access_Violation	DXB_Error
---	---	------------	----------	---	---------------	----------------------	-----------

28

24

WD_State_Changed:	The state of the WD_SM has changed (change between 'Baud_Search', 'Baud_Control' or 'DP_Control').
Diag_Fetched:	The master fetched the diagnostic buffer
Delay_Timer Overrun	The delay timer has expired and has been started over
Ignore_Del_Timer Overrun	The time of day synchronization has been aborted. External Del_Timer_Overrun Timer must be deleted.
IndQ_Entry:	An entry was made in the indication queue
IndQ_Full:	The Indication_Queue is full. The pending indication could not be transferred
Go/Leave_Data_Exchange:	DPS has entered the 'Data_Exchange' mode or has exited it
New_GC_Command:	DPS has received a Global_Control message with a modified 'GC_Command byte' (New_GC_Int_Mode=0) and has stored this byte in the RAM cell 'GC_Command'. If 'New_GC_Int_Mode=1', this interrupt is set for every received Global_Control message.
New_SSA_Data:	DPS has received a 'Set_Slave_Address message' and has made the data available in the User_SSA buffer.
New_Prm_Data:	DPS has received a 'Set_Param message' and has made the data available in the User_Prm buffer.
New_DDB_Prm_Data	DPS has received a 'Set_DDB_Param Frame' and provided the data in the User_DDB_Prm Buffer.
New_Cfg_Data:	DPS has received a 'Check_Cfg message' and has made the data available in the User_Cfg buffer.
Get_Cfg_Buffer_Changed:	Upon request by 'User_New_Get_Cfg_Buf', DPS has exchanged the Get_Config buffers and has made the old buffer available again to the user.
Diag_Buffer_Changed:	Upon request by "User_New_Diag_Buf", DPS has exchanged the diagnostic buffers and has made the old buffer available again to the user.
DX_OUT:	DPS has received a 'Write_Read_Data/GC message' and made the new output data available in the N buffer. In the case of 'Power_On', 'Clear', or 'Leave_Master', the DPS_SM makes a cleared C buffer available and generates this interrupt also. By parameterizing 'Enable_DX_Out_Port=1' in the C31_Control register, the interrupt 'DX_OUT' can be applied directly to Port PB <sub>3</sub> .
DX_OUT_Overflow:	DPS has received a 'Write_Read_Data/GC message' and has made the new output data available in the N buffer. However, the old data wasn't fetched and is no longer available. In the sync mode, the frozen output data in the D buffer was overwritten because there was no GC message.
DXB0-7_OUT:	DPS has tapped data from a configured connection and provided the new output data in the N buffer in the appropriate DXB_Buffer_SMO-7.
DXB_Error:	During the last watchdog cycle, at least one connection failed or was returned.
RAM_Access_Violation:	The memory was accessed outside the communication memory.
SSC_Interface:	The SSC interface generated an interrupt.
GC_Clock	The Sync_Clock (clock pulse beater) has been triggered at Port PB <sub>2</sub>
DX-Entered	Comes after the transition to Data-Exchange synchronously with the first DX_OUT interrupt in which diagnostics are not pending

After reset, the IR is cleared.

**Interrupt Register, IR (readable only):**

For bit assignment, refer to Interrupt Request Register.

**Interrupt Mask Register, IMR (writable, can be changed during operation):**

For bit assignment, refer to Interrupt Request Register.

Bit = 1: Mask is set and the interrupt is disabled

Bit = 0: Mask is cleared and the interrupt is enabled.

After reset, all bits are set.

**Interrupt Acknowledge Register, IAR (writable, can be changed during operation):**

For bit assignment, refer to Interrupt Request Register.

Bit = 1: The IRR bit is cleared.

Bit = 0: The IRR bit remains unchanged.

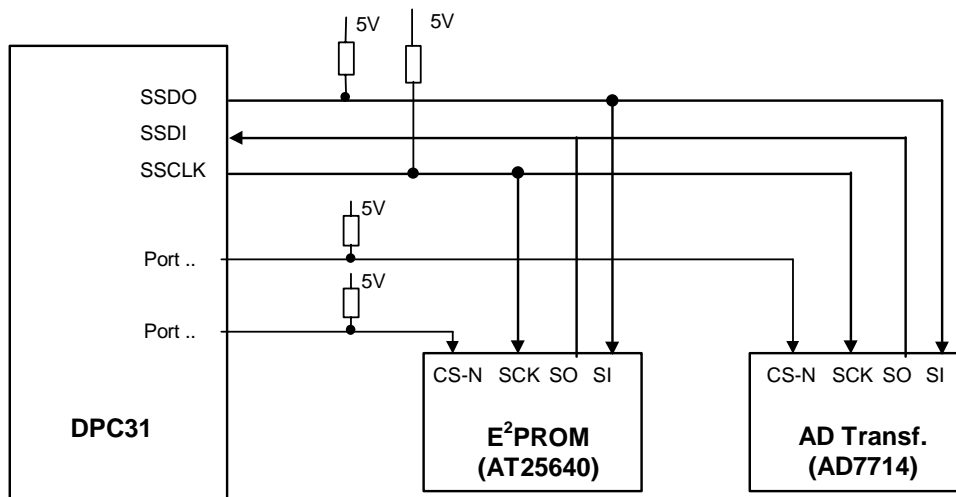
After reset, all bits are cleared.

**Interrupt EOI Register, EOI (writable, can be changed during operation)**

EOI is triggered based on the write operation to the register cell 'Interrupt EOI Register'. The write data is don't care.

**7.2 Synchronous Serial Interface (SSC Interface)**

In the DPC31, a universal synchronous serial interface is integrated. In addition, several SPI slave blocks (ser. E<sup>2</sup>PROMs or AD transformers) can be connected to this interface (Figure 7.2-1). This SSC interface has full duplex capability, and only supports the master mode.



**Figure 7.2-1: SPI Interface at the DPC31**

To connect **SPI devices** (ser. E<sup>2</sup>PROM, AD transformer), an output port is needed per SPI device, in addition to the SSC channel, in order to generate the chip select signal.

**Description of the SSC Module:**

The SSC module consists of a transmit channel and a receive channel. Each channel contains a 9-bit shift register, and an 8-bit buffer. Character widths of 1 to 8-bit are supported.

The user operates the transmit buffer. If the transmit buffer is empty, the transmitter generates the Transmit Buffer Empty which can be polled via the status register, or which, with a corresponding enable in the Interrupt Enable Register, activates the SSC interrupt. After loading the transmit buffer, Transmit Buffer Empty enters inactive. As soon as the transmit shift register is free, the data byte is

transferred there and shifted out. The clock (SSCLK) is generated only as long as the shift process is running. During continuous sending, the user always writes the next data byte to the transmit buffer while one is being shifted out.

In the receiver, the arriving bits are shifted to the Receiver Shift Register. After 8 data bits have been received, or 9 bits with enabled parity, this data byte is accepted in the receive buffer and Receive Buffer Full is generated. This state can be polled via the status register or it can be activated as SSC\_Interface interrupt if there is a corresponding enable in the Interrupt Enable Register.

If there is continuous receiving, the user reads a data byte from the receive buffer while the next one is arriving at the receiver shift register. Error states (Receive Buffer Overflow, RECERR; or Parity Error, PERR) can be polled in the status register or can be generated as SSC\_Interface interrupt (enable in the Interrupt Enable Register).

Because of the full duplex channel in the SSC module, it can receive while it is sending. However, the protocols process only half-duplex (SPI E<sup>2</sup>PROM, etc.). For that reason, the received data is to be ignored (disable the corresponding interrupts). The last received data byte is always in the receive buffer. To receive user data, dummy data bytes have to be sent so that the SSC module generates a clock pulse.

### Register Assignment of the SSC Module:

The user (external  $\mu$ P or C31) addresses the SSC module in the address range from 0020h to 0025h. It can be polled or operated with interrupt output. The interrupt runs to the two interrupt controllers (refer to Chapter 7.1.4).

### Control1 Register:

Bit Position	7	6	5	4	3	2	1	0
	BREN	-	PODD	PPOS	PEN	HCB	CPOL	CPHA
Default	0	0	0	0	0	0	1	1
	rw	r	rw	rw	rw	rw	rw	rw

CPHA:	Clock Phase Control Bit =0 Acceptance of the receive data at the leading clock edge; sending at the back clock edge. =1 Shifting the send data at the leading clock edge; receiving at the back clock edge.
CPOL:	Clock Polarity Control Bit =0 Clock idle state is 'low'; leading clock edge is a low-to-high edge. =1 Clock idle state is 'high'; leading clock edge is high-to-low edge.
HCB:	Heading Control Bit =0 Send/receive LSB first. =1 Send/receive MSB first.
PEN:	Parity Control Bit =0 Generating/checking parity disabled. =1 Generating/checking parity enabled.
PPOS:	Parity Position Control Bit =0 Send/receive parity bit last. =1 Send/receive parity bit first.
PODD:	Parity Selection Bit =0 Even parity bit (parity bit generates in the data byte an even number of '1's). =1 Uneven parity bit (parity bit generates in the data byte an uneven number of '1's).

**BREN:** Baud rate Enable Bit  
 =0 Baud rate generator is disabled (power save).  
 =1 Baud rate generator is enabled.

**Control2 Register:**

Bit Position	7	6	5	4	3	2	1	0
	-	-	-	-	-	DW <sub>2</sub>	DW <sub>1</sub>	DW <sub>0</sub>
Default	0	0	0	0	0	0	0	0
	r	r	r	r	r	r/w	r/w	r/w

Bit	Function
DW <sub>2..0</sub>	Data Width Selection 000: Transfer data with 8 bit length 001: Transfer data with 1 bit length 010: Transfer data with 2 bit length 011: Transfer data with 3 bit length 100: Transfer data with 4 bit length 101: Transfer data with 5 bit length 110: Transfer data with 6 bit length 111: Transfer data with 7 bit length

**Status Register:**

Bit Position	7	6	5	4	3	2	1	0
	BUSY	-	-	-	REC-ERR	PERR	RBFU	TBEM
Default	0	0	0	0	0	0	0	1
	r	r	r	r	r/w	r/w	r	r

**TBEM:** Transmit Buffer Empty Flag  
 =0 Transmit buffer is full.  
 =1 Transmit buffer is empty.

**RBFU:** Receive Buffer Full Flag  
 =0 Receive buffer is empty.  
 =1 Receive buffer is full.

**PERR:** Parity Error Flag  
 =0 No parity error in data byte.  
 =1 Parity error in data byte; has to be reset by the user.

**RECERR:** Receive Error Flag  
 =0 No receive buffer overflow.  
 =1 Receive buffer overflow; has to be reset by the user.

**BUSY:** Busy Flag  
 =0 No action; SSC module can be reparameterized.  
 =1 Action on the bus; reparameterization not permitted.

These bits are ORed to the interrupt 'SSC\_Interface'. They must have been enabled in the Interrupt Enable Register.

**Interrupt Enable Register:**

Bit Position	7	6	5	4	3	2	1	0
	-	-	-	-	EN-RECERR	EN-PERR	EN-RBFU	EN-TBEM
Default					0	0	0	0
					w	w	w	w

ENTBEM:	Enable Transmit Buffer Empty Interrupt =0 Transmit Buffer Empty Interrupt is disabled. =1 Transmit Buffer Empty Interrupt is enabled.
ENRBFU:	Enable Receive Buffer Full Interrupt =0 Receive Buffer Full Interrupt is disabled. =1 Receive Buffer Full Interrupt is enabled.
ENPERR:	Enable Parity Error Interrupt =0 Parity Error Interrupt is disabled. =1 Parity Error Interrupt is enabled.
ENRECERR:	Enable Receive Error Interrupt =0 Receive Error Interrupt is disabled. =1 Receive Error Interrupt is enabled.

**Baud rate Register:**

An 8-bit division factor (G) is loaded in the baud register. This value specifies the baud rate according to the following formula: ( $f_{\text{sys}}$  = internal system clock). At 48 MHz, synchronous transmission of 12 MBaud maximum is possible.

$$BR = f_{\text{sys}} / 4(G+1)$$

**7.3 80C31 Core and Interface**

The internal C31 core is SW-compatible with Industrial Standard 8031 (including command execution times). In addition, it has Timer2 from the 80C32 and the internal work memory consisting of 256 bytes. *Below, this internal processor is called "C31".* All functions of the controller can be used by the user except port PD2, where the interrupt of the sequential control system is located. The C31 runs with half of the input frequency (for asynchronous with 24MHz, for synchronous with 2, 4, or 8 MHz).

In order to get the original performance of the C31, Ports A, B, and D must be wired with external pull-up resistors. Address Port C is always on Output and thus does not have to be wired with pull-up resistors. The same applies to Port D2 (XINT0), Port D6 (XWR) and Port D7 (XRD).

**Notes:**

The ports E, F, G and H are configured as input or output channels by the user program if the interface is set to I/O (BUSTYPE<sub>2..0</sub> = "1 - -").

### 7.3.1 Reset Phase of the C31

The reset phase of the C31 needs a minimum time span of 30 elementary periods. The build-up time of the PLL is at 200 µs after the supply voltage and the external quartz have stabilized.

#### 7.3.1.1 Boot Type Setting

In order to start the DPC31, the boot type has to be set. **Presently, only Boot Type 2 is permissible.**

BOOT TYPE		
Bit 1	Bit 0	
0	0	Type 1a
0	1	Type 1b
<b>1</b>	<b>0</b>	<b>Type 2</b>
1	1	Type 3

**Table 7.3-1:** Boot Type Settings

#### 7.3.1.2 Boot Type 2

Two variants are possible for Boot Type 2:

1. The internal C31 core processes the program that is stored in the externally connected EPROM (Port A .. D). Ports E .. H are free and can be used for I/O.
2. The µP/I/O interface (ports E .. H) can be used for connection to an external µP system (with EPROM) or as I/O channels. Via the SPI interface, an A/D transformer and/or an EPROM can be connected in addition.

**Figure 7.3-1: Operation in Boot Type 2**

## 7.3.2 80C31 Core

### 7.3.2.1 Data Area

The processor has an "internal" work memory consisting of 256 bytes.

The data area of the processor is broken down into different blocks (Figure 7.3-2):

The register cells (interrupt controller, DPS control units, etc.) are located from Address 000h to 004Fh. From Address 0050h to 008Fh, the I/O ports E, F, G, and H can be addressed. From 0090h to 07FFh is an unused area. The internal RAM follows starting with address 0800h broken down into the block: work cells, parameter cells, and buffer management, which consists of approx. 0.5 kByte, and the communication area, which consists of 5.5 kByte.

Starting with 2000h, the external RAM is accessed (signal pin: XCSDATA = low).

External RAM (Data Memory)	FFFFh
	2000h
int. CRAM (5.5 kByte)	1FFFh
S/R_UnitTemp Buffer Internal Stack (for sequential control system) Buffer Management Parameter Cells Internal Variables	approx. 09FFh
<b>Area that can't be used</b>	Start internal RAM 07FFh
	0090h
Port H Direction Register Port H (1 = Input, 0 = Output)	0089h
ByteAddress H <sub>2..0</sub>	0088h
BitAddress H <sub>2</sub>	0082h
BitAddress H <sub>1</sub>	...
BitAddress H <sub>0</sub>	0080h
Port G Direction Register Port G (1 = Input, 0 = Output)	0079h
ByteAddress G <sub>7..0</sub>	0078h
BitAddress G <sub>7</sub>	0077h
...	...
BitAddress G <sub>0</sub>	0070h
Port F Direction Register Port F (1 = Input, 0 = Output)	0069h
ByteAddress F <sub>7..0</sub>	0068h
BitAddress F <sub>7</sub>	0067h
...	...
BitAddress F <sub>0</sub>	0060h
Port E Direction Register Port E (1 = Input, 0 = Output)	0059h
ByteAddress E <sub>7..0</sub>	0058h
BitAddress E <sub>7</sub>	0057h
BitAddress E <sub>6</sub>	0056h
BitAddress E <sub>5</sub>	0055h
BitAddress E <sub>4</sub>	0054h
BitAddress E <sub>3</sub>	0053h
BitAddress E <sub>2</sub>	0052h
BitAddress E <sub>1</sub>	0051h
BitAddress E <sub>0</sub>	0050h
DPS Control Units SSC Interface Parameter Register / Delay Timer Interrupt Controller	004Fh   0000h

**Figure 7.3-2:** X Data Area of the Internal Processor

### 7.3.2.2 Code Area

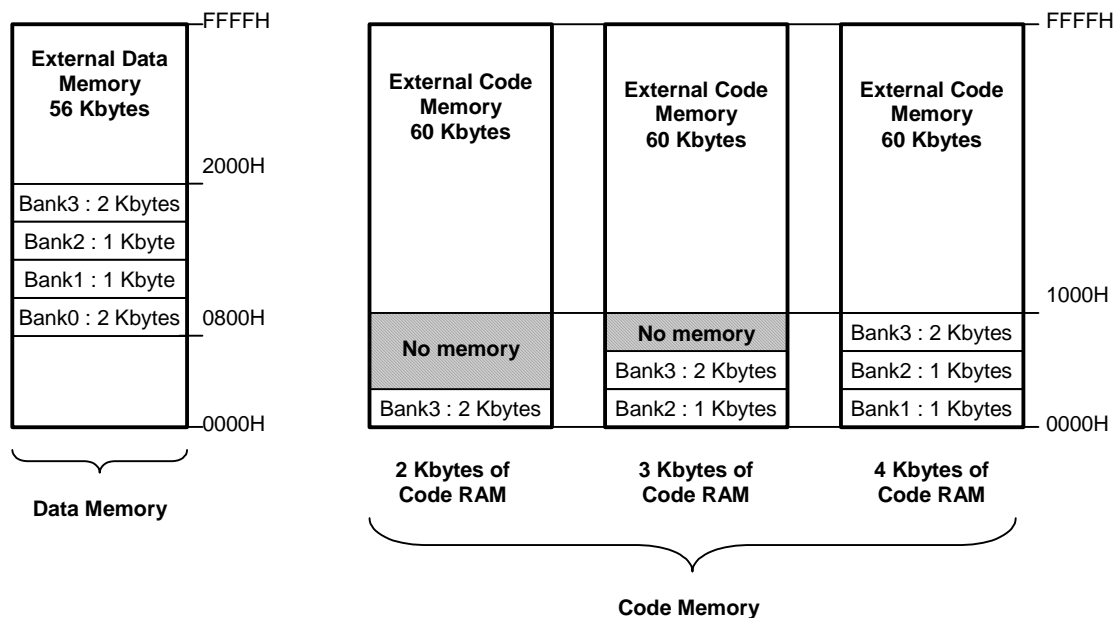
The code area normally includes all 64 Kbytes of the address space. The XCS\_CODE and XPSEN signals are active when the code memory is accessed.

If a code RAM is programmed by means of the C31 control register, the address area 0000H – 1000H is reserved for the code RAM and the external code memory is no longer addressed in address area 0000H – 1000H.

### 7.3.3 Code RAM

During normal operation (no code RAM), all accesses go in the code area on the external program memory (XCSCODE = low).

However, the 'C31 Control Register' can be used to show a portion of the internal RAM in the code area as so-called 'Code RAM'. The C31 then no longer obtains the program code from the external code memory but rather from the internal code RAM shown on the same address.



**Table 7.3-1: Allocation Between KRAM and Code RAM**

When Code RAM is used, the following must be observed:

- Code RAM is composed of the individual blocks of the multiport RAM.
- Prior to programming as Code RAM, the program code must be landed in the RAM through accesses on the data memory.
- As long as a RAM bank is used as code RAM, access to the relevant RAM bank via the data area is no longer permitted.
- If a code RAM is used, the lowest 4 Kbytes can no longer be addressed by the external code memory.
- A reset resets the register settings for the code RAM. The content of the internal RAM can also be affected by a reset.

### 7.3.4 Expansion Interface to the 80C31 Core

Via the ports A, B, C, and D, the ALE and XPSEN signal, all signals of the C31 are taken outside. The C31 must always be operated with address and data bus because the external memory of the DPC31 is connected to it. The exact assignment is provided in Table 7.3-2 (function/alternative function). PD2 is not to be used; here, the interrupt of the sequential control system is located that is always taken permanently to the outside. In addition, the following signals are generated: "XCSDATA" (chip select external data memory (RAM)) and "XCSCODE" (chip select external program memory (ROM)). XCSDATA = low if the access is made to the external data area (starting with address 2000h). XSCODE = low, if the external code area is accessed (if code RAM is not used, starting at address 0000h to the end; otherwise, starting at address 1000h). These signals are always to be connected so that there will not be driver conflicts when connecting an In Circuit Emulator (ICE).

This makes connecting a standard In Circuit Emulator for an 8052 controller (24 MHz) possible. For this, the pin has to be wired DBX = high.

### 7.3.5 Interface Signals

Pin Name	Function		Alternative Fct.		DebugMode (ICE) DBX = '1'		Comment	
	Type	Signal Name	Type	Signal Name	Type	Signal Name		
PA <sub>7..0</sub>	I/O	AB <sub>7..0</sub> / DB <sub>7..0</sub>	I/O	-		AB <sub>7..0</sub> / DB <sub>7..0</sub>	I/O	Multiplexed address/data bus
PB <sub>0</sub>	I/O	P1.0	I/O	T2	I	-	I	
PB <sub>1</sub>	I/O	P1.1	I/O	T2EX	I	-	I	
PB <sub>2..7</sub>	I/O	P1.2 ... P1.7	I/O	-		-	I	
PC <sub>7..0</sub>	I/O	AB <sub>15..8</sub>	O	-		AB <sub>15..8</sub>	I	Address bus more significant byte
PD <sub>0</sub>	I/O	P3.0	I/O	RXD	I	-	I	
PD <sub>1</sub>	I/O	P3.1	I/O	TXD	O	-	I	
PD <sub>2</sub>	I/O	XINT0	O	-		XINT0	O	Interrupt of the seq. ctrl. syst.
PD <sub>3</sub>	I/O	P3.3	I/O	XINT1	I	-	I	Ext. interrupt
PD <sub>4</sub>	I/O	P3.4	I/O	T0	I		I	
PD <sub>5</sub>	I/O	P3.5	I/O	T1	I		I	
PD <sub>6</sub>	I/O	XWR	O	-		XWR	I	
PD <sub>7</sub>	I/O	XRD	O	-		XRD	I	
ALE	I/O	ALE	O	-		ALE	I	Address Latch Enable
XPSEN	I/O	XPSEN	O	-		XPSEN	I	Output Enable for Code-Memory
XCSDATA	O	XCSDATA	O	-		XCSDATA	O	Chip Select for Data Memory
XCSCODE	O	XCSCODE	O	-		XCSCODE	O	Chip Select for Code Memory
DBX	I	DBX	I	-		DBX	I	In Circuit Emulator debug mode

**Table 7.3-2: Interface Signals of the C31**

## 7.4 C31 Interrupt Controller in the DPC31

Via this interrupt controller, the C31 can be provided with the same interrupt events as the external  $\mu$ P (refer to Chapter 7.1.4).

It is structured exactly as the other interrupt controller. Each event is stored in the IRR. Via the IMR, individual events can be suppressed. If, for instance, the DPS indications are to be evaluated by the external processor, the corresponding masks have to be set here and be enabled in the interrupt controller for the external processor. The entry in the IRR is independent of the interrupt mask. The event signals that are not masked out in the IMR generate the **C31 interrupt** via a summation network.

For debugging, the user can set any event in the IRR (activate only the bits that are to be reset).

Before leaving the interrupt routine, the C31 has to set the “**End of Interrupt signal (EOI) = 1**” in the EOI register. With this edge change, the interrupt line is switched inactive. If an event should still be stored, the interrupt output becomes active again only after an interrupt inactive time of at least 1  $\mu$ s but no more than 2  $\mu$ s (refer to Section 9.6.2.2).

The interrupt registers IRR, IR, IMR, IAR, and the EOI register are described in Chapter 7.1.4.

The PD<sub>2</sub>/XINT0 pin is always an output (negative active). The state of the C31 interrupt controller can be observed externally at this output. In contrast to a normal C31, this pin can be used neither as an IO port nor as an interrupt input.

**These interrupt registers -assigned only to the C31- can be accessed by the C31 under the same addresses as the interrupt registers assigned to the host interface. Only the interrupt outputs (ports PG5 and PD2) are different.**

## 7.5 Serial PROFIBUS Interface

### 7.5.1 Asynchronous Physics Unit (NRZ)

#### 7.5.1.1 Transmitter

The transmitter converts the parallel data structure into a serial data stream. The asynchronous UART process processes with a start bit and a stop bit that frame 9 information bits (8 data bits; 1 even parity bit). The start bit is always log '0', and the stop bit as well as the idle state are always log '1'. The least significant bit is transmitted first.

The transmitter switches the request to send (RTS) active first. After a minimum waiting time of 4 elementary periods (at XCTS active), it then starts the transmission process. (To connect a modem, the XCTS input is available. After RTS is active, the transmitter must hold back the first message character until the modem activates XCTS. During message transmission, the transmitter no longer queries the XCTS.)

When closing transmission, the transmitter deactivates the RTS.

#### 7.5.1.2 Receiver

The receiver converts the serial data stream into the parallel data structure. It scans the serial data stream with the 4-fold transmission speed. One requirement of the PROFIBUS protocol is that no idle states are permitted between the message characters. The DPC31 transmitter ensures that this specification is adhered to. In order to check outside systems (for example, S/W solutions) with respect to this point, supplementary logic is implemented in the DPC31 receiver. The receiver checks whether start bit synchronization takes place (not at the ED character of a message) after the stop bit. By parameterizing “DIS\_START\_CONTROL=1” (in the param register, or 'Set\_Param message' for DP), this subsequent start bit check is switched off.

Due to the 4-fold scan, a maximum distortion of the serial input signal of  $X = -47\%$  to  $y = +22\%$  in reference to the falling start bit edge is permissible.

### 7.5.1.3 Interface Signals

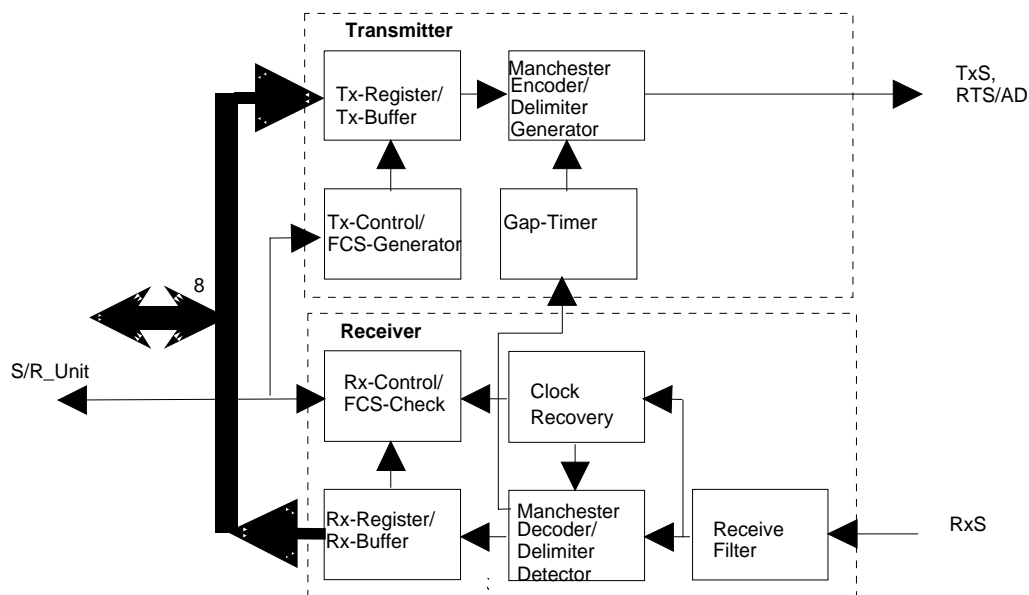
Pin Name	Signal Name	Input=I Output=O	Comment
TXD_TXS	TxD	O	Send Data
RXD_RXS	RxD	I	Receive Data
RTS_TXE	RTS	O	Enable of the send drivers
XCTS_RXA	XCTS	I	Sender Enable

**Table7.5-1: Asynchronous PROFIBUS Interface of the DPC31**

In the test mode, all outputs are switched to high resistance.

### 7.5.2 Synchronous Physics Unit (Manchester)

The synchronous interface makes data transmission according to IEC 1158-2 possible. It includes services of the interface -defined in this standard- between the following: data link layer and physical layer ((**FDL Ph layer interface**), the sublayers Ph DIS (**DCE independent sublayer**) and Ph MDS (**medium dependent sublayer**) for wire media and the corresponding **MDS-MAU interface**. In addition, the **station management physical layer interface** is implemented (parts of the service primitives, optionally defined in Standard IEC 1158-2). The so-called "medium access unit (MAU)" is not implemented, which includes the following: the initial pulse shaper, the line driver, the receive amplifier, the receive filters and the line coupling (if needed, with remote supply setup). The MAU can be set up with little effort with the SIM1 Analog ASIC.

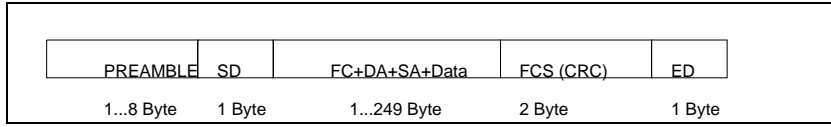


**Table7.5-2: Block Diagram of the Synchronous Interface.**

#### 7.5.2.1 Transmitter

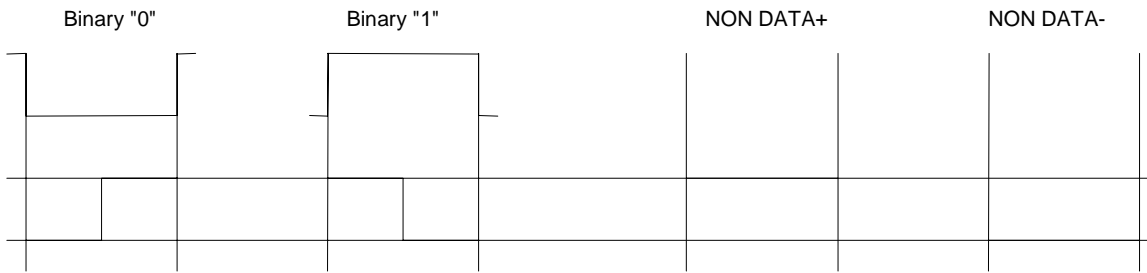
The transmitter converts the parallel data structure into a serial data stream. The synchronous transmission procedure according to IEC 1158-2 processes with Manchester coding and start and end delimiters. Each message is preceded by a preamble. The length of the preamble is stored in the preamble register (refer to 3.3). **In contrast to the asynchronous interface, the most**

significant data bit is transmitted first<sup>5</sup>. The transmitter generates a 16-bit CRC field and attaches it to the data field.

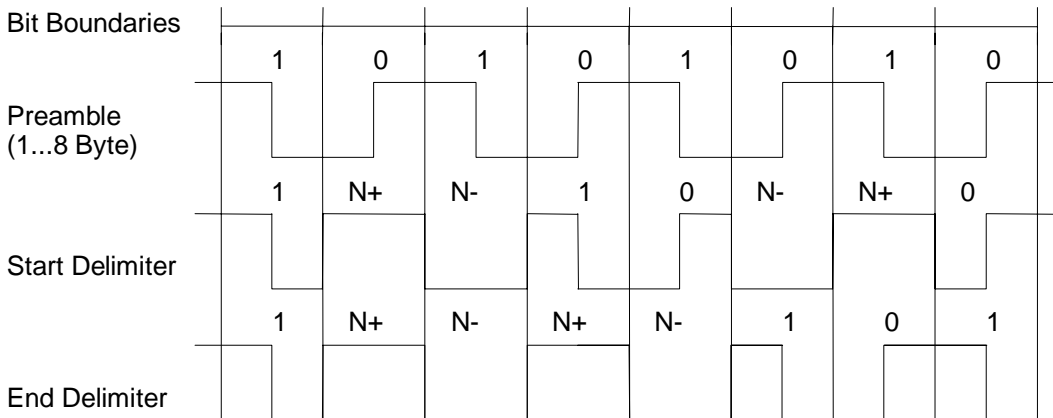


**Table7.5-3: Frame Structure of the Serial Interface**

Figure 7.5-5 shows the coding rules . Figure 7.5-6 shows the structure of the preamble and of the delimiters. These figures show that the elementary characters (= smallest quantization unit) at the transmitter output have the length of half a bit period. Their generation requires the double bit clock.



**Table7.5-4: Bit Coding of the Synchronous Interface**



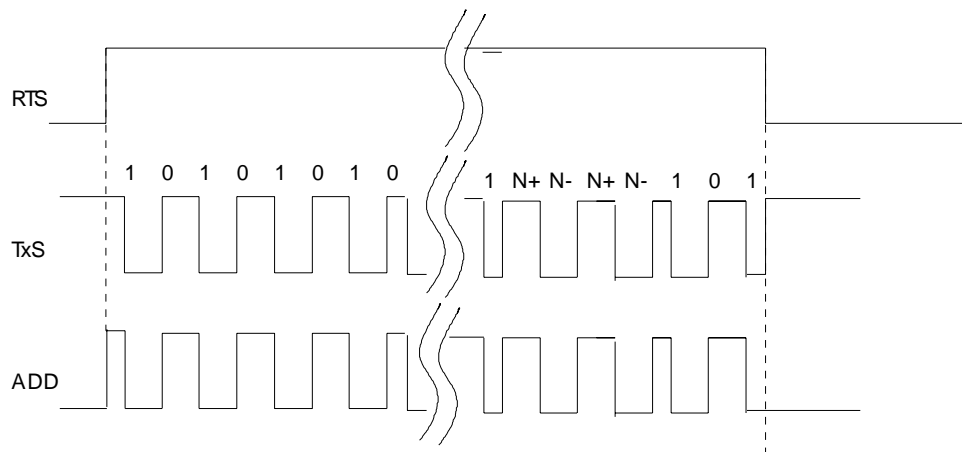
**Table7.5-5: Preamble and Delimiters**

The transmitter makes different output signals available (Figure 7.5-6). In addition to the signals RTS (enable of the send driver) and TxS (send signal), the signal ADD can be utilized. With the combination of TxS and ADD, an adder circuit for activating a current control unit can easily be established as it is used for the interface of an intrinsically safe bus station. The combination RxS/TxS is an advantage when activating a transformer.

It is useful to make the signals RTS and ADD available at a joint output (RTS/ADD). Switching between the two modes can be parameterized (Param Register; refer to 3.3).

In order to ensure the minimum gap between two messages, the transmitter is disabled at the end of a message for the duration of a minimum interframe gap time. The gap timer is loaded with the current value for the interframe gap time from the interframe GAP\_Time register (Chapter 3.3).

<sup>5</sup> according to IEC 1158-2, Chapter 7.



**Table 7.5-6: Output Signals of the Synchronous Transmitter**

## 7.5.2.2 Receiver

### Receive Filter

The receive filter conditions the receive signal RxS for clock recovery and for decoding.

### Manchester Decoder and Clock Recovery

This unit includes all the resources that are needed to decode the data from the filtered receive signal.

The **Clock Recovery** recovers the clock CLK1 from the filtered receive signal and the system clock CLK16 (whose nominal frequency corresponds to the 16-fold data rate).

Because of the ambiguity of the zero crossings<sup>6</sup> and because of the normally relatively long “catch time” of a phase control loop, it is necessary to provide the clock recovery with a quick synchronization setup (quick synchronizer) which, at the beginning of each receive process, quickly synchronizes the recovered clock with the receive signal.

The signal RxA, generated by the line activity detector, switches the synchronizer into a “quick synchronization mode” at the beginning of a message. In this mode, **the fourth zero crossing** (or the first four zero crossings)<sup>7</sup> of the signal supplied by the preamble filter leads to resynchronization (Zero\_Phase=transition to the initial state) respectively. After the quick synchronization phase, the receive clock is corrected only with  $\pm 1/16$  clock period regarding phase deviation from the signal FRxS<sup>8</sup>. This state is retained until the next falling edge of the signal RxA.

The DPC31 has an improved quick synchronizer. To activate it, the user must set the bit ‘Quick\_Sync\_New=1’ in the param register (refer to Chapter 3.3). In this mode, the DPC31 attempts to more accurately determine the bit center during the preamble phase by recording the duration of the last high and low phase before the 4<sup>th</sup> edge. From the average of these two numbers, it calculates a correction value which is taken into account when the bit center is specified.

The **data decoder** scans the filtered receive signal with the recovered receive clock (positive edge), and passes on the scan value, weighted with the polarity information (POL=1, or POL=0) that was transferred by the decoder state machine as receive signal RxD.

<sup>6</sup> Only the zero crossings in bit center can be utilized for clock recovery.

<sup>7</sup> According to IEC 1158-2 (Chapter 9.6), at least four bits are available to the preamble for synchronization. Multiple synchronization during this phase does not provide advantages. A decrease in the error frequency would be attainable through notification via several bits (three maximum). The estimated improvement, however, does not justify the probably much greater effort. In the example shown, it is assumed that the spike that possibly occurs at the start of the preamble is to be counted with two transitions (worst case). The signal QSync indicates the duration of the synchronization process.

<sup>8</sup> Through this rigid phase control loop, the required detection according to IEC 1158-2 (Chapter 9.7) of half-bit slip errors is ensured.

### 7.5.2.3 Power-Saving Serial Interface

Figure 7.5-8 shows three different interfaces of the SIM1 at the DPC31.

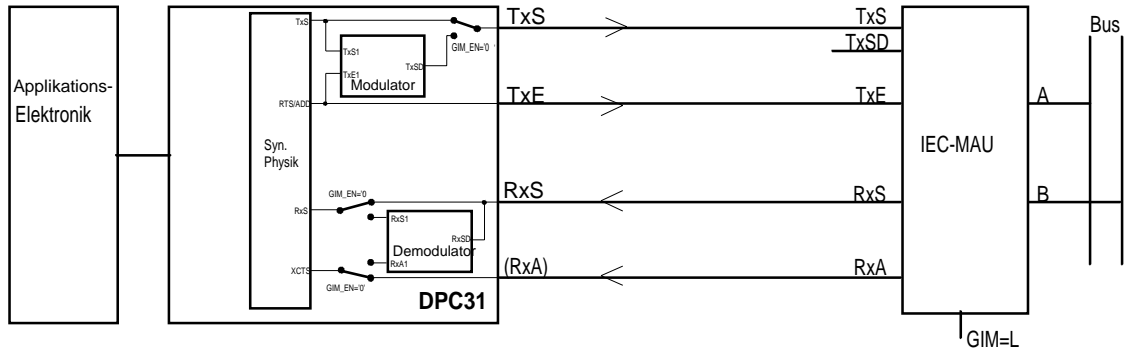
If no galvanic isolation of the bus interface (SIM1) is required by the application-specific electronics, the send signals (TxS, TxE) and receive signals (RxS, RxA) are passed on without processing in the DPC31 to the synchronous bus physics unit (Figure 7.5-8a) with the parameter assignment `GIM_EN='0'` (Galvanic Isolation Mode, refer to Param Register, Chapter 3.3) in the interface of the power-saving serial interface. The output levels RxA and RxS are adjusted via the supply input  $V_{IF}$  (SIM1).

To galvanically isolate the lines for the data- and auxiliary signals, different isolated components and circuits can be used (Figure 7.5-7b and c). The conventional type provides for an optocoupler each for the signals TxS, TxE, RxS (and RxA). Otherwise, processing the send and receive signals in the interface of the power-saving serial interface is as shown in Figure 7.5-7a.

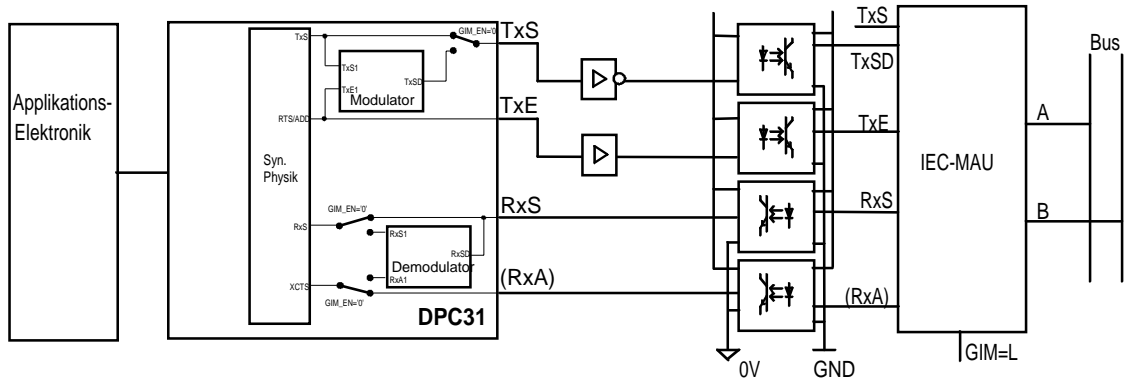
To implement a power-saving method of working with optocouplers, an interface logic was conceived (Figure 7.5-7c) which is to be activated via the parameter assignment `GIM_EN='1'`.

This circuit generates short pulse-width modulated transmission pulses only in the case of edge transitions of the data stream from which the data signal is recovered in the secondary circuit. The mean power input can thus be reduced to low values. The following are pointed out as special features:

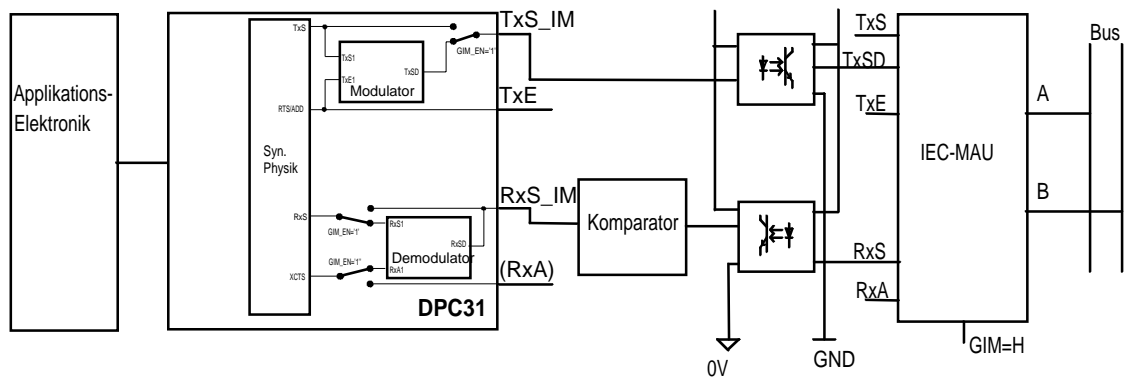
- Combination of the control and data signals in a transmission channel (TxSD, RxSD); thus, reducing the interface width for send and receive direction from 4 to 2 optocoupler channels.
- Suitable for 5V and 3V engineering
- Use of conventional optocoupler blocks with simple selection at the manufacturer; can also be used for optocouplers with higher power requirements and approval for intrinsically safe circuits.
- The power-saving interface can be used only for a transmission rate of 31.25kBd (refer to Param Register, Chapter 3.3).



a) Ohne galvanische Trennung



b) Herkömmliche Trennung mit Optokopplern



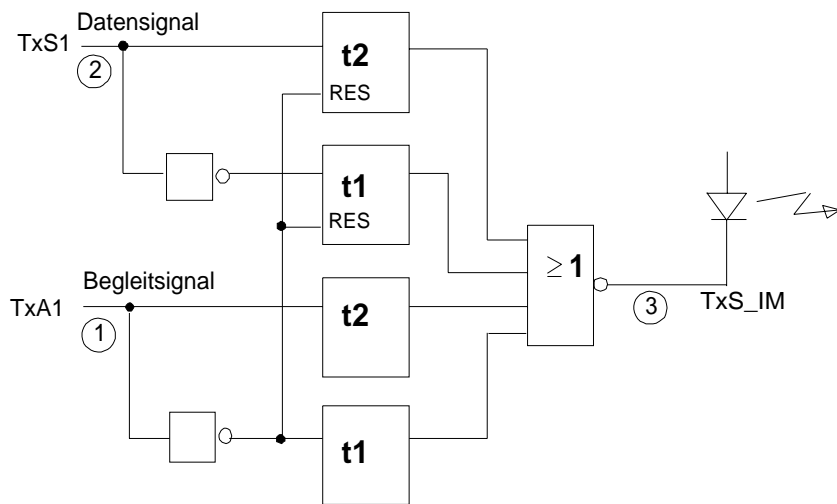
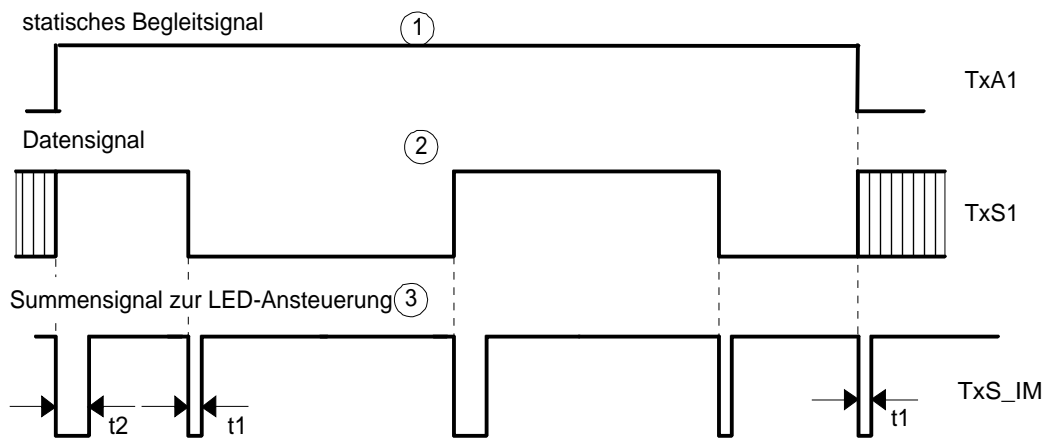
c) Stromsparende Trennung mit Optokopplern

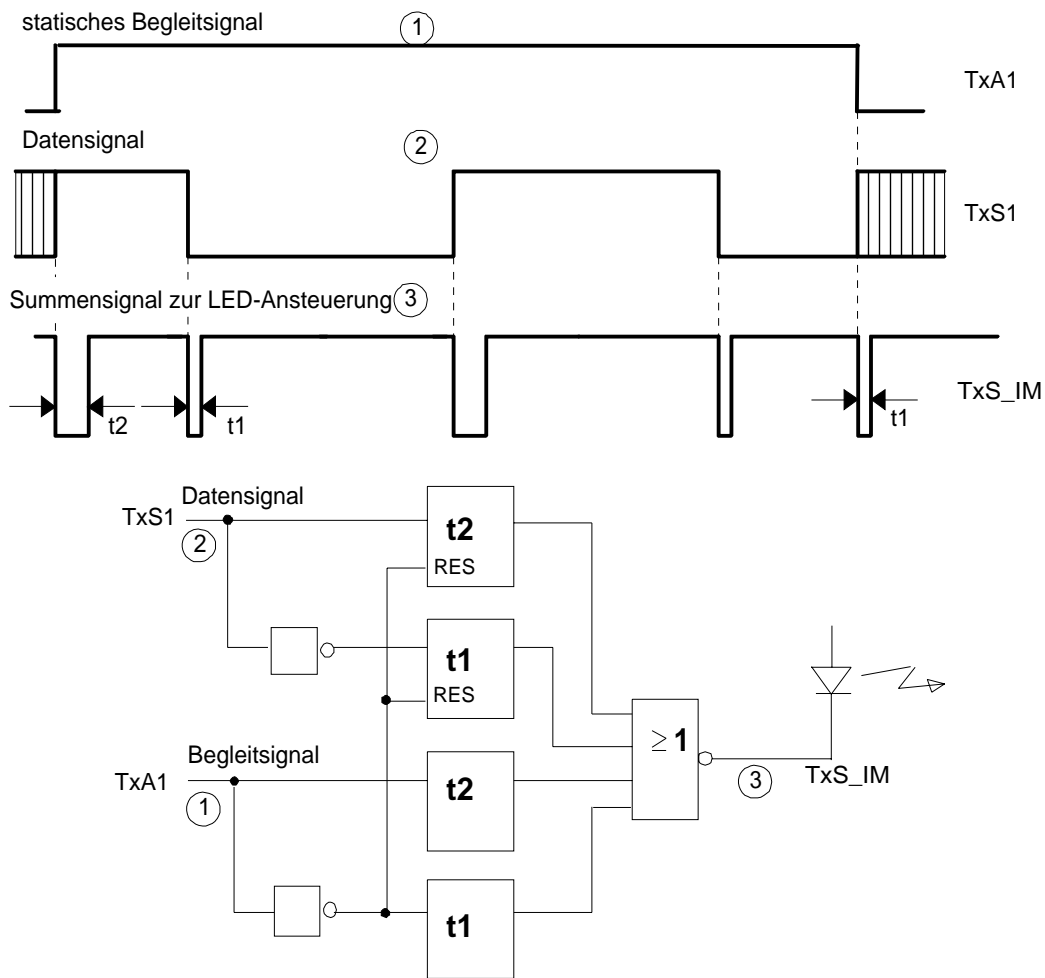
**Table 7.5-7: Interface to communication-controller DPC31**

The interface logic of the power-saving serial interface includes a pulse modulator and a pulse demodulator as in the SIM1. The comparator for regeneration of the analog receive signal behind the optocoupler is not integrated into the DPC31 but must be set up externally.

**Pulse Duration Modulator:**

In the galvanic isolation mode (GIM\_EN='1'), the PDM (Figure 7.5-8) converts the serial signal that is to be transmitted into a duration-modulated pulse sequence; the rising edge of the send signal (TxS1) is assigned a long pulse and the falling edge is assigned a short pulse. Likewise with the edges of the static auxiliary signal (TxE1 or RTS/ADD), a long and short pulse is generated which are added to the pulse sequence of the data signal. The summation signal thus generated (TxS\_IM) is used for sampling the LED of an optocoupler.





**Table 7.5-8: Signal Shaping in the Modulator**

**Pulse Duration Demodulator:**

In the galvanic isolation mode (GIM\_EN='1'), the useful signal for the PDM is recovered from the collector signal of the optocoupler transistors by using a comparator.

The following digital circuit component (integrated into the DPC31) evaluates the length of the output pulses of the comparator and recovers from it the data signal and the auxiliary signal. The circuit diagram of the demodulator is shown in Figure 7.5-9. The signal characteristic with respect to time is shown in Figure 7.5-10.

When using RxS\_IM, the pin RxA is to be applied to GND.

Table7.5-9: Circuit of the Demodulator in principle

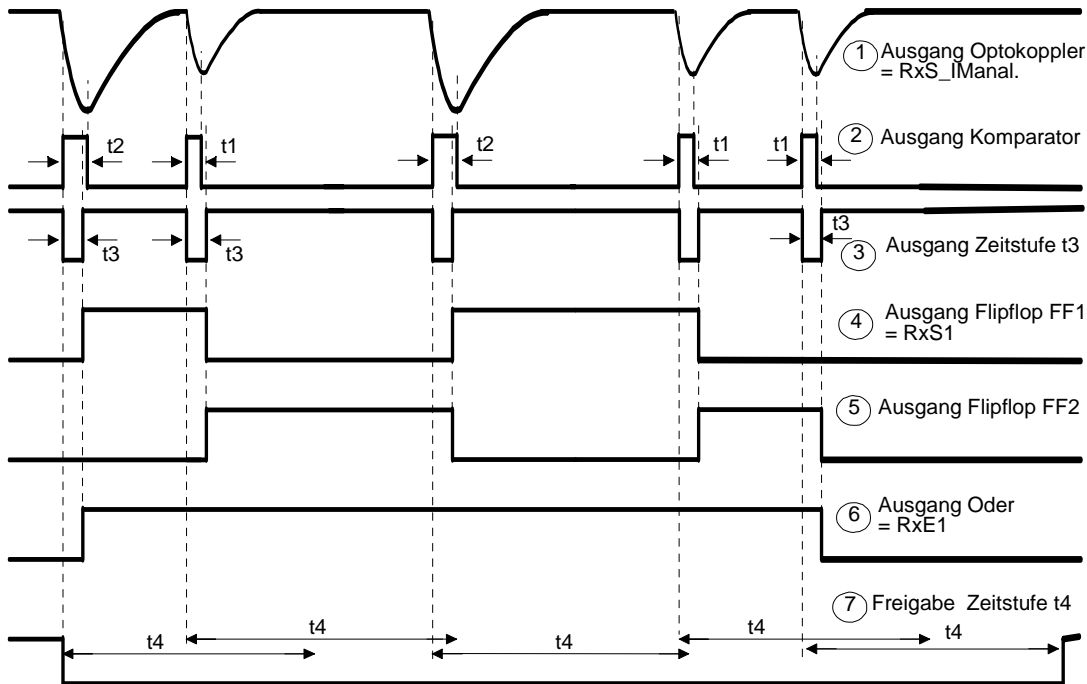


Table7.5-10: Signal Evaluation of the Demodulator

The rising edge of each incoming pulse (2) triggers a monoflop at runtime  $t_3$ . The following condition applies:  $t_1 < t_3 < t_2$ . When  $t_3$  elapses, a scan is made to see if the pulse is still pending, that is, the input signal still corresponds to a logical '1':

- If the pulse is no longer pending, a **short pulse** is detected ( $t_1$ )
- If the pulse is still pending, a **long pulse** is detected ( $t_2$ )

Depending on the detected pulse duration  $t_1$  (falling edge) or  $t_2$  (rising edge), the flip-flop FF 1 is set to L or H. The output of the flip-flop thus corresponds to the serial data signal RxS1 (4). The output signal (5) of an additional flip-flop FF 2 is logically connected to the signal (4) via an OR function.

When two short pulses in a row occur, both flip-flops are reset (frame end). The OR function yields an L which is recognized as the end of static signal RxE1 (6).

At scan time  $t_3$ , an uncertainty window is specified because a jitter in the amount of the pending clock pulse period always occurs in the switching system. In this case, the jitter is  $0.5 \mu\text{s}$ . Time  $t_3$  is permanently set in Step A/B of the DPC31. However, because the time durations  $t_1$  and  $t_2$  can change due to distortions caused by the optical coupler interface module, an assignable time window was implemented in Step C. This corresponds exactly to the values that were implemented in SPC4-2. (see Table 7-5-1.

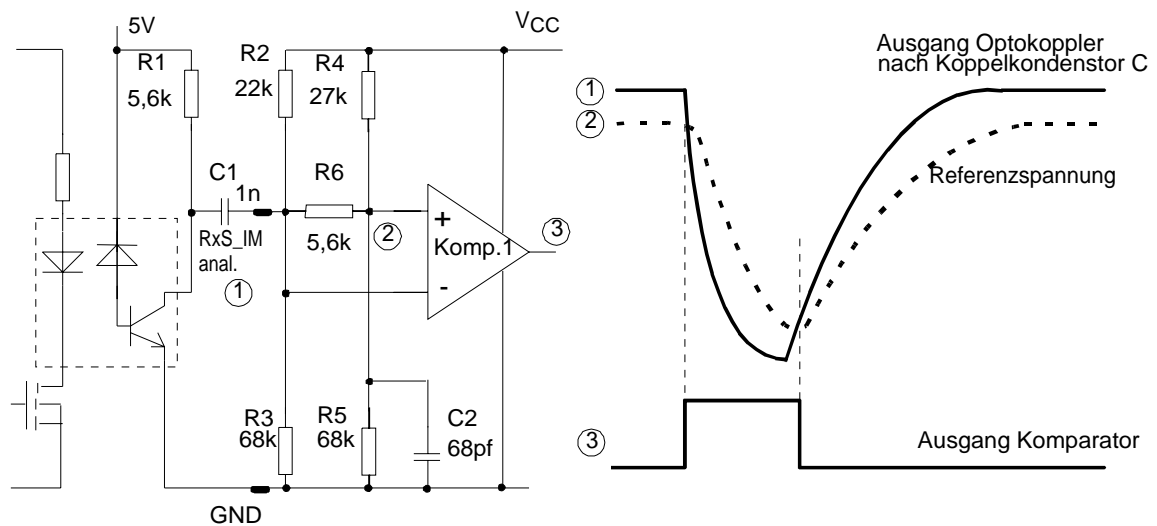
The various scan times  $t_3$  are set by the bits within the expanded parameter register designated for scan mode (see Section "Organizational Parameters – Expanded Parameter Register"). The scan mode set by default after reset deviates from the permanently set scan mode in Step A/B.

Scan Mode (1 ... 0)	Uncertainty Window	
	min	max
01 (after a Reset)	3.0 $\mu$ s	3.5 $\mu$ s
00	2.5 $\mu$ s	3.0 $\mu$ s
10	3.5 $\mu$ s	4.0 $\mu$ s
11	4.0 $\mu$ s	4.5 $\mu$ s

**Table 7.5-11:** Uncertainty Window

**Alternative Suggestion regarding Comparator Circuitry:**

The wiring of the comparator output described under Figure 7.5-? has the disadvantage that the comparator has to be supplied with the external voltage 5V via the input  $V_{IF}$ , and a level adaptation is necessary at the output. In addition, a control area up to the positive supply voltage has to be ensured. The circuit variant below (Figure 7.5-11) avoids these disadvantages. The two voltage dividers R2 / R3 and R4 / R5 move the work area of the comparator to the center of the internal supply voltage  $V_{CC}$ ; an offset results from the difference of the values R2 and R4 in the idle state; R6 causes a decrease in amplitude; C2 a delay of the reference voltage in the active circuit state. The capacitor C1 decouples the external voltage 5V and the internal  $V_{CC}$ . This comparator circuit is not integrated into the DPC31 and must be implemented externally.



**Table7.5-12:** Wiring of the Comparator with Bridge Network

### 7.5.2.4 Interface Signals

Pin Name	Signal Name	Input=I Output=O	Comment
TXD_TXS	TxS (TxS_IM)	O	Send signal (for asyn. physics TxD)
RXD_RXS	RxS (RxS_IM)	I	Receive signal (for asyn. physics RxD)
RTS_TXE	TxE	O	Enable of the send drivers/addition signal (for asyn. physics RTS)
XCTS_RXA	RxA	I	Auxiliary signal for receive (has not been needed so far in the syn. physics unit (apply to GND); for asyn. physics XCTS)

**Figure 7.5-13:** Synchronous PROFIBUS Interface of the DPC31

In the test mode, all outputs are switched to high resistance.

## 7.6 DPS Watchdog Timer

### 7.7 Watchdog Timer

#### 7.7.1 Automatic Baud Rate Detection

The DPC31 is able to recognize the baud rate automatically. The "Baud\_Search" mode is entered after each RESET as well as after the expiration of the Watchdog(WD) timer in the 'Baud\_Control' mode.

The DPC31 starts the search for the set baud rate always with the highest baud rate. If during the monitoring time no SD1, SD2, or SD3 message has been received completely and faultlessly, the search is continued with the next lower baud rate.

After detecting the correct baud rate, the DPC31 switches to the "Baud\_Control" mode and monitors the baud rate. The monitoring time can be parameterized (WD\_Baud\_Control\_Val). The watchdog processes in this case with a clock of 100 Hz (10 msec). Each faultlessly received message to its own station address resets the watchdog. If the timer expires, the DPC31 reswitches to the Baud\_Search mode.

#### 7.7.2 Baud rate Monitoring

In 'Baud\_Control', the baud rate that was found is monitored **continuously**. With each faultless address to the DPC31s own station address, the watchdog is reset. The monitoring time is the result of multiplying 'WD\_Baud\_Control\_Val' (to be parameterized by the user) by the time base (10 ms). If the monitoring time expires, the WD\_SM reenters 'Baud\_Search'. If the user handles the DP protocol with the DPC31 (DP\_Mode =1; refer to Mode Register 0), the watchdog is used for the 'DP\_Control' mode after a 'Set\_Param message' with enabled response monitoring 'WD\_On = 1' was received. If the master monitoring 'WD\_On = 0' is switched off, the watchdog timer remains in the baud rate monitoring mode. The PROFIBUS DP state machine is not reset if the timer expires; that is, the slave remains in the DATA\_Exchange mode.

### 7.7.3 Response Monitoring

The 'DP\_Control' mode is used for response monitoring of the DP master (Master\_Add). The set monitoring time is the result of multiplying both watchdog factors and then multiplying by the time base valid at the moment (1 ms or 10 ms):

$T_{WD} = (1 \text{ ms or } 10 \text{ ms}) * WD\_Fact\_1 * WD\_Fact\_2$  (refer to Byte 7 of the parameter assignment message).

The two watchdog factors (WD\_Fact\_1, WD\_Fact\_2) and the time base that represent a value for the monitoring time can be loaded by the user with the 'Set\_Param message' with any value between 1 and 255.

**Exception: the setting  $WD\_Fact\_1=WD\_Fact\_2=1$  is not permissible.** This setting is not checked by the circuit.

With the permissible watchdog factors, monitoring timing between 2 ms and 650s can thus be implemented regardless of the baud rate.

If the monitoring time expires, the DPC31 reenters 'Baud\_Control' and the DPC31 generates the 'WD\_DP\_Control\_Timeout interrupt'. In addition, the state machine is reset; that is, the reset modes of buffer management are generated.

If another master takes over the DPC31, it either switches to 'Baud\_Control' (WD\_On = 0) or it remains in 'DP\_Control' (WD\_On = 1) depending on the enabled response monitoring.

## 7.8 Clock Supply

### 7.8.1 PLL

In the asynchronous mode, the clock pulse is generated with an integrated oscillator and an analog-PLL in the DPC31. The oscillator pins (XTAL1\_CLK and XTAL2) are, as shown in Figure 7.8-1, wired with the values according to Table 7.8.2. The following PLL quadruples the input frequency of 12 MHz (pin XPLEN = low). The DPC31 now has the internal system frequency of  $f_{SYS} = 48\text{MHz}$ . It is not possible to connect the PLL with an external clock pulse generator. The internal system clock has an inaccuracy from the external quartz (here assumed to be  $\pm 150 \text{ ppm}$ ) plus the inaccuracy of the PLL ( $\pm 200 \text{ ppm}$ ). The rise time of the PLL is at 1 ms after the supply voltage and the external quartz have stabilized.

In the synchronous mode, the lower system frequency ( $f_{SYS} = 16/8/4/(/2)\text{MHz}$ ) is supplied via an external clock pulse generator directly at pin XTAL1\_CLK. The integrated oscillator and the PLL are switched off in that case (pin XPLEN = high, power-save mode). (2 MHz system frequency is not enabled.)

To connect an external  $\mu$ Processor, the output CLKOUT1X2 ( $f_{SYS} / 2$  or  $f_{sys}$ ) and/or CLKOUT1X4 ( $f_{SYS} / 4$ ) can be used. The outputs are active after being switched on -also during the reset phase- and can be switched off via Mode Register0.

The internal processing clock pulse is  $f_{SYS}/2$ . The bus physics unit is operated with the scanning frequency (4-fold for asynchronous, 16-fold for synchronous).

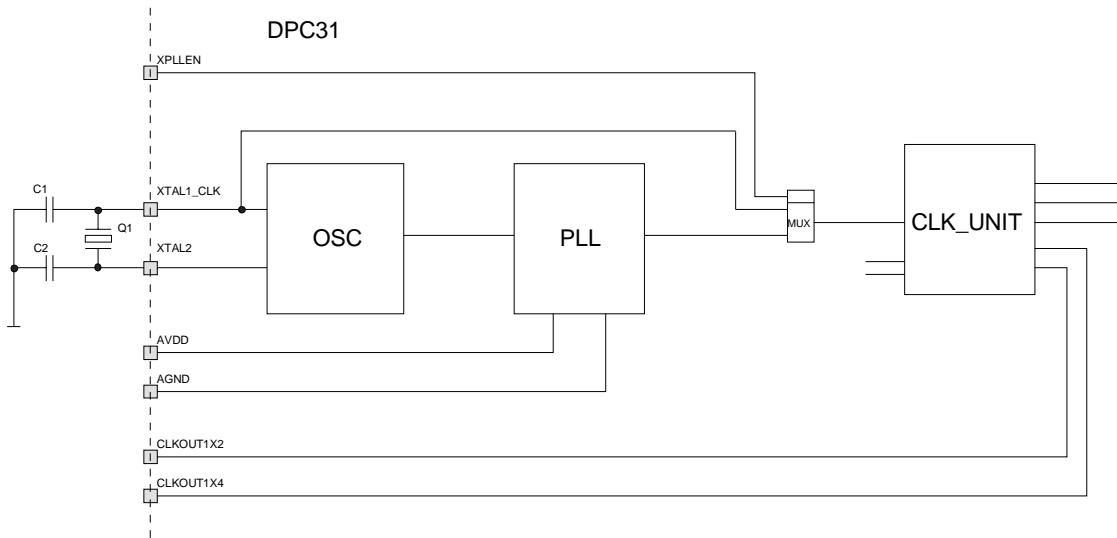
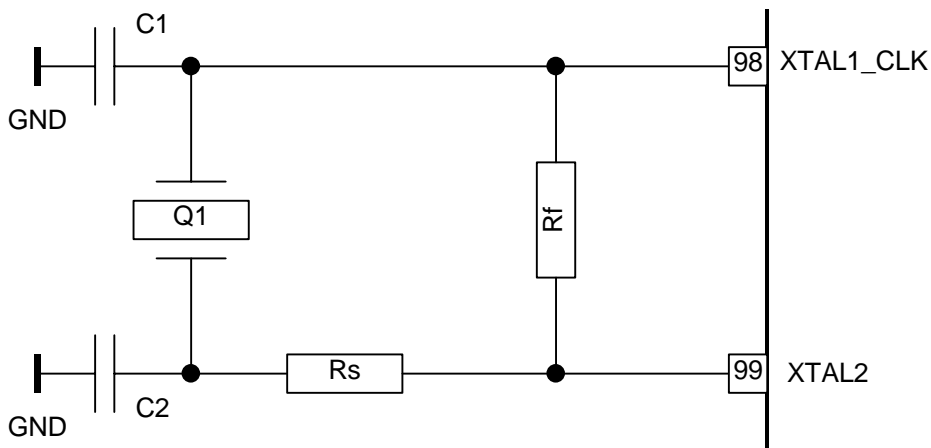


Figure 7.8-1: Block Diagram of Clock Supply

Pin Name	Pad	Comment
XTAL1_CLK	I	Quartz connection / direct clock input (for syn. mode)
XTAL2	O	Quartz connection
XPLEN	I	Selection PLL or clock input
CLKOUT1X2	O	Half of the internal clock (clock for In Circuit Emulator) or the internal clock
CLKOUT1X4	O	Quarter of the internal clock
AVDD	-	Supply pin of PLL
AGND	-	Supply pin of PLL

Table 7.8-1: Pins for the Clock Supply



Component	Value
Q1	12 MHz quartz; Epson type: MA505, MA506, MA406, MA406H
Rs	2,7 K $\Omega$
Rf	1,0 M $\Omega$
C1	22 pF
C2	22 pF

**Table 7.8-2:** Component Values of Oscillator Wiring

Notes on PCB layout:

- Place all components of the quartz interface connection as close as possible to Pins 98 and 99 .
- No other signals should cross the area of the quartz interface connection – nor the other signal positions.
- The CLKOUT1X2 signal (Pin 100) should be routed in the maximum possible clearance to the quartz interface connection.
- Shield the entire quartz interface connection and Pins 98 and 99 against possible interferences.
- The voltage supply of PLL should be as free of interference as possible. If required, provide external support or filtering.

## 8 Test Support

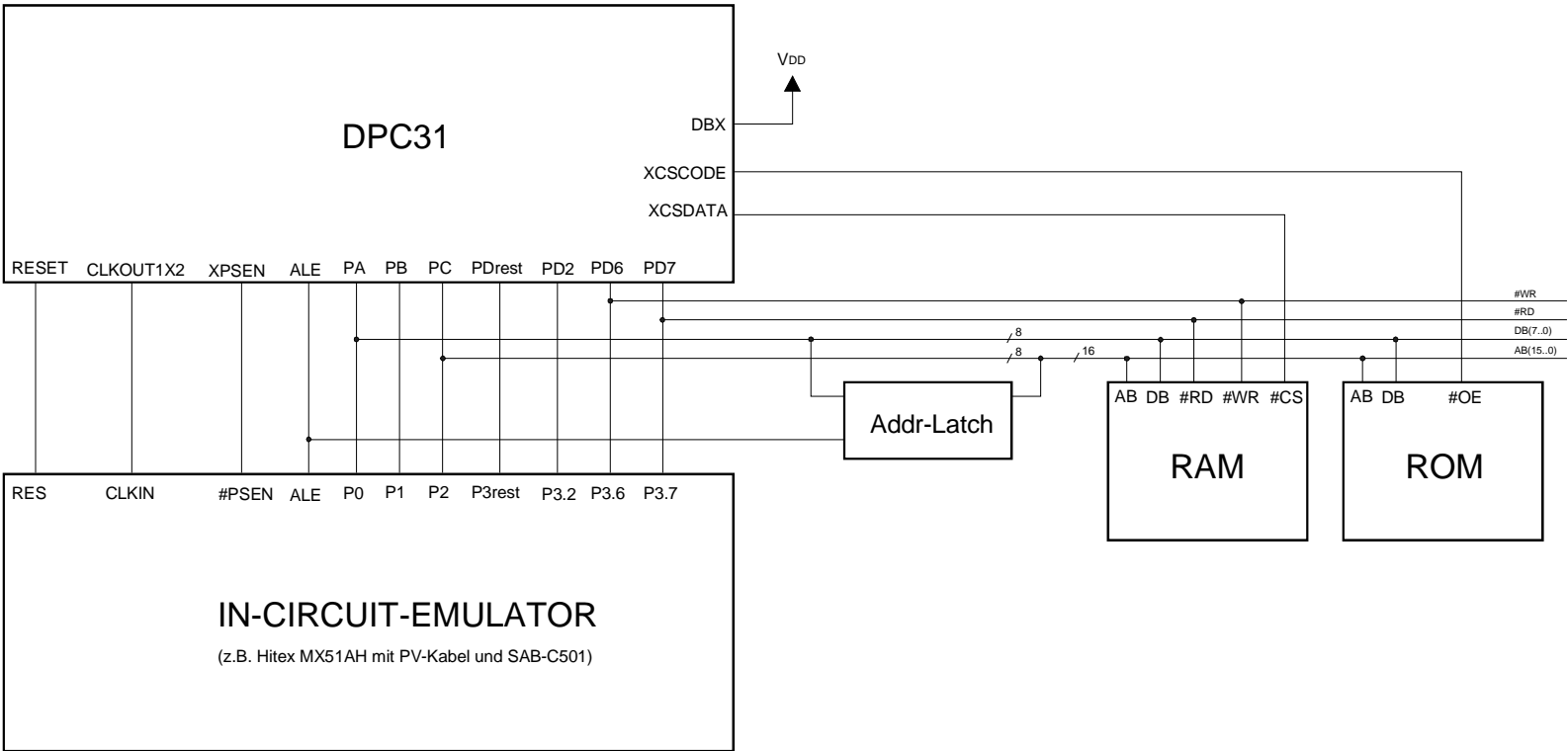
The DPC31 has three test pins (TST1,NTEST1, NTEST2). For operation, all pins are to be at 0 Volt. To switch the outputs to high-resistance (In Circuit Test), NTEST1/2 are to be at 1.

### 8.1 Emulator Connection for the C31

To emulate the C31 that is integrated in the DPC31, a standard emulator (such as Hitex MX51AH) can be connected. The interfacing is shown in Figure 8.1-1. The emulator must be used with the SAB-C501-40 or a type compatible with the timing, because of the more relaxed timing of the processor.

**Problem Case:** If the C165 (@20MHz) without tristate time waitstate for DPC31 accesses and the C31 emulator (@24MHz) are operated together, there may be access conflicts to the internal DPC31 RAM.

**Remedy:** For accesses by the C165 to the DPC31, the tristate time waitstate is to be set accordingly.



## 9 Electrical Specifications

### 9.1 Maximum Limits

Parameter	Name	Condition	Limits	Unit
DC voltage supply	$V_{DD}$		-0.3 to +4.0	V
Input voltage <sup>9</sup>				
• All inputs except Pin 98	$V_I$		-0.3 to +6.0	V
• Pin 98 (XTAL1_CLK)	$V_I$		-0.3 to $V_{DD}+0.3$	V
Output voltage <sup>10</sup>				
• All outputs except Pin 95, 96, 99	$V_O$		-0.3 to +6.0 <sup>11</sup>	V
• Pin 95, 96, 99	$V_O$		-0.3 to $V_{DD}+0.3$	V
DC output current (min)	$I_O$	$I_{OL} = 4.0\text{mA}$	-13	mA
( @ $V_O = 0\text{V} = \text{short circuit}$ )	$I_O$	$I_{OL} = 8.0\text{mA}$	-26	mA
Ambient temperature	$T_{opt}$		-40 to +85	°C
Storage temperature	$T_{stg}$		-55 to +125	°C
Power loss for PQFP-100	$P_{vmax}$		800	mW
Junction temperature	$\vartheta_{jmax}$		125	°C
$R_{th}$ junction case	$R_{thj \rightarrow c}$		---	K/W
$R_{th}$ case-ambient	$R_{thc \rightarrow a}$		---	K/W
$R_{th}$ junction-ambient	$R_{thj \rightarrow a}$		27	K/W

Table 9.1-1: Maximum Limits

### 9.2 Permitted Operating Values

<sup>9</sup> In DPC31 Step A/B, Pin 98 was not examined separately in this table.

<sup>10</sup> In DPC31 Step A/B, Pins 95, 96, and 99 were not examined separately in this table

<sup>11</sup> The output itself can supply a maximum voltage of  $V_{DD}+0.3$  V. Higher voltages at the output pins have an external cause, e.g., pull up resistors.

Parameter	Name.	Min.	Max.	Unit
DC supply voltage	$V_{DD}$	3.0	3.6	V
Input voltage (Low-Level) <sup>12</sup>	$V_{IL}$	0	0.8	V
Input voltage (High-Level) <sup>12</sup>	$V_{IH}$	2.0	5.5	V
Input voltage (Low-Level trigger) <sup>13</sup>	$V_{IL}$	0	0.7	V
Input voltage (High-Level trigger) <sup>13</sup>	$V_{IH}$	2.1	5.5	V
Hysteresis <sup>13</sup>	$V_H$	0.4	--	V
Input rise time	$t_r$	0	200	ns
Input dropout time	$t_f$	0	200	ns
Bus fight time	$t_{BF}$	0	20	ns
Schmitt trigger input rise time	$t_r$	0	10	ms
Schmitt trigger input dropout time	$t_f$	0	10	ms

**Table 9.2-1:** Permitted Operating Values

### 9.3 Guaranteed Operating Range for the Specified Parameters

Parameters	Name	Min.	Max.	Unit
DC Supply Voltage	$V_{DD}$	3.0	3.6	V
Operating Temperature	$T_{opt}$	-40	+85	°C

**Table 9.3-1:** Guaranteed Operating Range of the Specified Parameters

### 9.4 Power Loss

Power Loss: (all values worst case estimate)

Asynchronous: approx. 325 mW at 12 MBd

Synchronous: approx. 7 mW at 31.25 kBd and 2MHz clock (C31 switched off)  
 approx. 11 mW at 31.25 kBd and 2MHz clock (C31 core @ 1MHz)  
 approx. 35 mW at 31.25 kBd and 16MHz clock (C31 core @ 8MHz)

Power Loss: (all values measured typically)

Asynchronous: no values

Synchronous: no values

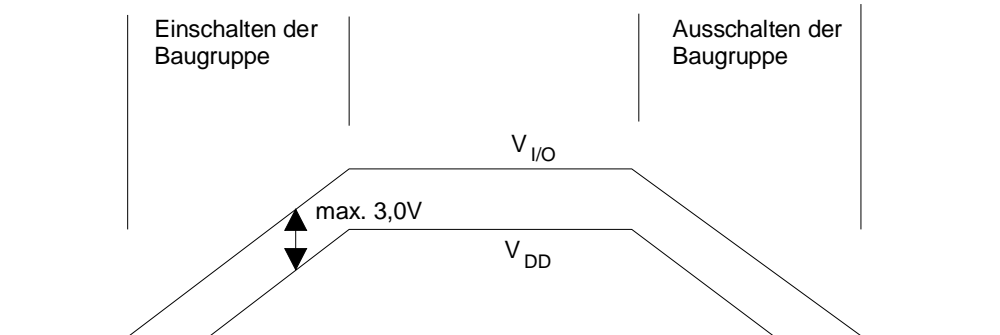
#### 9.4.1 Power-Up of the Supply Voltage

If the DPC31 is used in modules with mixed voltage supply (3.3V and 5V), the voltage difference between the supply pins ( $V_{DD} = 3.3V \pm 10\%$ ) and the signal pins ( $V_{I/O}$ ) is to be no larger than +3.0V at any time ( $V_{I/O} - V_{DD} < 3.0V$ ). If this value is exceeded, the DPC31 will be destroyed.

All pins except Pin 95, 96, 98, and 99:

<sup>12</sup> DPC31-Step C: does **not** apply to Input Pins 6, 87, 88, and 98.

<sup>13</sup> DPC31-Step C: **only** applies to Input Pins 6, 87, and 88.



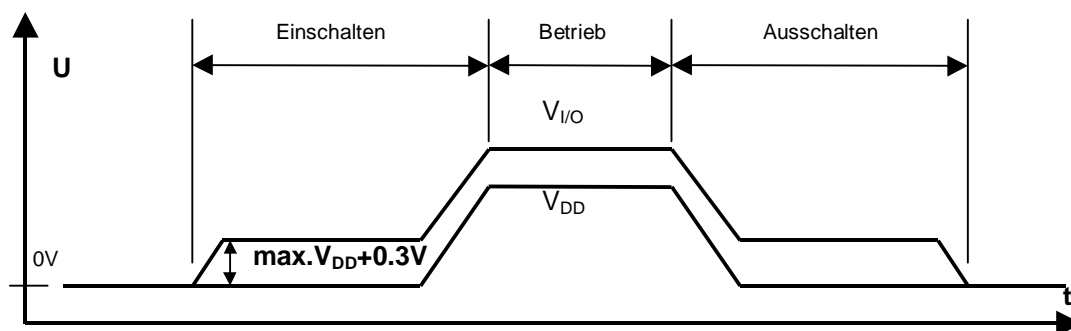
**Table 9.4-1: Voltage Ramp for 5 V-Tolerant I/O**

During switching on and off, the voltage at the 5 V-tolerant signal pins must not exceed a value of +6.0 V or fall below a value of -0.3 V.

The 5V-tolerant outputs can themselves only cause a voltage of -0.3 V up to  $V_{DD}+0.3$  V to flow. A higher voltage at the pin can only be brought about by external components, e.g., pull-up resistors.

These values must not be exceeded; otherwise the DPC31 may be destroyed.

Pins 95, 96, 98, and 99 only:



**Table 9.4-2: Voltage Ramp for LVTTTL I/O**

During switching on and switching off, the voltage at the indicated signal pins must not exceed a value of  $V_{DD}+0.3$  V or fall below a value of -0.3 V.

These values must not be exceeded; otherwise the DPC31 may be destroyed.

## 9.4.2 Structure of the Pad Cells with 5V Tolerance

The input pad cells used have a tolerance of 5V; that is, they are provided with a protective circuit. This means that, although they are supplied internally with only 3.3V, the input level may be 6 V maximum. Table 9.5-1 shows the operating points.

The 5V-tolerant output pad cells are also provided with a special protective circuit. When driving the 0-level, there is no difference with respect to the conventional pad cells. The 1-level is driven actively up to  $V_{DD} - 0.3V$ . *Starting with this voltage, the external pull-up resistor pulls the level to  $V_{DD2}$  (5V). This pull-up is needed only if a 5V-CMOS input is to be driven. For reasons of interference immunity, TTL-level is recommended.*

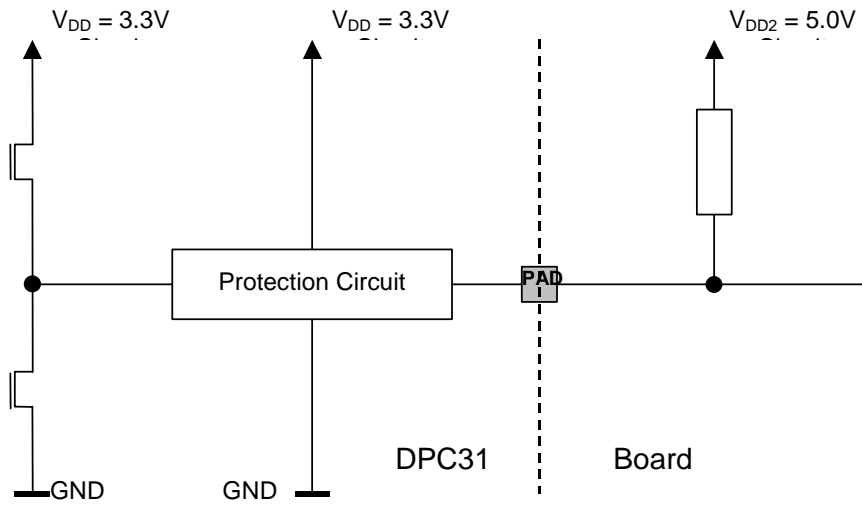


Figure 9.5-2: Wiring of an Output Pad Cell with 5V Tolerance

## 9.4.3 DC Specification of the Pad Cells

Parameter	Na me.	Condition	Min.	Type	Max.	Unit
Input voltage 0-level <sup>14</sup>	V <sub>IL</sub>		0		0.8	V
Input voltage 1-level <sup>14</sup>	V <sub>IH</sub>		2.0		5.5	V
Output voltage 0-level <sup>15</sup>	V <sub>OL</sub>	I <sub>OL</sub> = 0 mA			0.1	V
Output voltage 1-level	V <sub>OH</sub>	I <sub>OH</sub> = 0 mA	V <sub>DD</sub> - 0.2		5.5 <sup>16</sup>	V
Output voltage 1-level <sup>17</sup>	V <sub>OH</sub>	I <sub>OH</sub> = 0 mA	V <sub>DD</sub> - 0.2		V <sub>DD</sub>	V
Schmitt trigger. +ve threshold <sup>18</sup>	V <sub>P</sub>				2.1	V
Schmitt Trig. -ve threshold <sup>18</sup>	V <sub>N</sub>		0.7			V
Schmitt Trig. hysteresis <sup>18</sup>	V <sub>H</sub>		0.4			V
Schmitt Trig. input voltage <sup>18</sup>	V <sub>IH</sub>				5.5	V
Input leakage current	I <sub>I</sub>	V <sub>I</sub> = V <sub>DD</sub> or GND			±1	µA
Output current 0-level 4 mA cell/ 5 V-tolerant	I <sub>OL</sub>	V <sub>OL</sub> = 0.4 V; T <sub>J</sub> =95°C V <sub>DD</sub> = 3V	3.84			mA
Output current 1-level 4 mA cell / 5 V-tolerant	I <sub>OH</sub>	V <sub>OH</sub> = 2.4 V; T <sub>J</sub> =95°C V <sub>DD</sub> = 3V	-3.84			mA
Output current 0-level 8 mA cell / 5 V-tolerant	I <sub>OL</sub>	V <sub>OL</sub> = 0.4 V; T <sub>J</sub> =95°C V <sub>DD</sub> = 3V	7.68			mA
Output current 1-level 8 mA cell / 5 V-tolerant	I <sub>OH</sub>	V <sub>OH</sub> = 2.4 V; T <sub>J</sub> =95°C V <sub>DD</sub> = 3V	-7.68			mA
Output current 0-level 8 mA cell/ 3.3 V	I <sub>OL</sub>	V <sub>OL</sub> = 0.4 V; T <sub>J</sub> =95°C V <sub>DD</sub> = 3V	7.68			mA
Output current 1-level 8 mA cell / 3.3 V	I <sub>OH</sub>	V <sub>OH</sub> = 2.4 V; T <sub>J</sub> =95°C V <sub>DD</sub> = 3V	-7.68			mA
Output current 0-level XTAL2 pin / 3.3 V	I <sub>OL</sub>	V <sub>OL</sub> = 0.4 V; T <sub>J</sub> =95°C V <sub>DD</sub> = 3.3V	1.92			mA
Output current 1-level XTAL2 pin / 3.3 V	I <sub>OH</sub>	V <sub>OH</sub> = 2.4 V; T <sub>J</sub> =95°C V <sub>DD</sub> = 3.3V	-1.92			mA
Tristate output leakage current	I <sub>OZ</sub>	V <sub>O</sub> = V <sub>DD</sub> or GND			±10	µA
Short circuit current	I <sub>OS</sub>	V <sub>O</sub> = 0 V V <sub>DD</sub> =3.6V T=-40°C			-145	mA
Input capacity	C <sub>IN</sub>	@ f = 1 MHz		10	20	pF
Output capacity	C <sub>OUT</sub>	@ f = 1 MHz		10	20	pF
I/O capacity	C <sub>I/O</sub>	@ f = 1 MHz		10	20	pF

Table 9.5-1: DC Specification of the Pad Cells

<sup>14</sup> All inputs except Pins 6, 87, 88, and 98<sup>15</sup> All outputs except Pins 95, 96, and 99<sup>16</sup> Each output can cause a maximum level of V<sub>DD</sub> to flow; if a higher level exists at an output, this is dependent on an external component, e.g., pull-up resistors<sup>17</sup> Applies to Pins 95, 96, and 99<sup>18</sup> Applies to Pins 6, 87, and 88

## 9.5 AC Specification

### 9.5.1 Driver Capability

The run times at the chip outputs always depend on the driver capacity of the pad cells as well as on the assumed capacitive load. The capacitive load that was used as a basis for the following timing specifications is shown in Table 9.6-1. To specify the maximum and minimum runtimes, the variations of temperature range and supply voltage range shown in Table 9.3-1 were included also.

Signal Name	Direction	Driver Type	Voltage	Capacity	Load
PA	In/Out	Tristate	5V tolerant	4 mA	120 pF **
PB	In/Out	Tristate	5V tolerant	4 mA	80 pF
PC	In/Out	Tristate	5V tolerant	4 mA	80 pF
PD	In/Out	Tristate	5V tolerant	4 mA	80 pF
ALE	In/Out	Tristate	5V tolerant	4 mA	80 pF
XPSEN ***	In/Out	Tristate	5V tolerant	4 mA	10 pF
XCSDATA	Out	Tristate	5V tolerant	4 mA	80 pF
XCSCODE	Out	Tristate	5V tolerant	4 mA	80 pF
PE	In/Out	Tristate	5V tolerant	4 mA	100 pF
PF	In/Out	Tristate	5V tolerant	4 mA	100 pF
PG	In/Out	Tristate	5V tolerant	4 mA	100 pF
PH	In/Out	Tristate	5V tolerant	4 mA	100 pF
SSCLK	Out	Tristate	5V tolerant	8 mA	100 pF
SSDO	Out	Tristate	5V tolerant	8 mA	100 pF
CLKOUT1X2	Out	Tristate	5V tolerant	8 mA	50 pF
CLKOUT1X4	Out	Tristate	5V tolerant	8 mA	50 pF
RTS_TXE	Out	Tristate	3.3V *	8 mA	50 pF
TXD_TXS	Out	Tristate	3.3V *	8 mA	50 pF

\*) No pull-up resistors!      \*\*) including the capacity of the emulation connection (70 pF)

\*\*\*) XPSEN to be used only for activating the emulator; otherwise, XCSCODE is to be used

**Table 9.5-1:** ID Data of the Outputs

If, in reality, the capacitive load deviates from the assumed values, the result will be a change of 0.7 ns maximum per 10pF.

### 9.5.2 Timing Diagrams, Signal Run Times

In general, the following applies: all signals that start with 'X' are 'low active'.

The signal runtimes are based on the capacitive loads shown in Table 9.6-1.

All timing that refers to the elementary period "T" is defined according to Table 9.6-2.

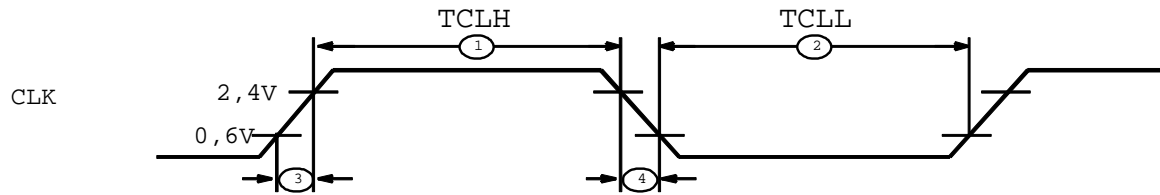
XP LLEN	Comment	T
1	Direct Clock Supply	1/CLK
0	Quartz Connection (12 MHz) ⇒ Internal Clock: 48 MHz	20.83 ns

**Table 9.5-2:** Definition of the Elementary Period T

### 9.5.2.1 Clock Supply (XPLLEN = '1')

No.	Parameters	Min	Max	Unit
1	Clock High Time	7.5		ns
2	Clock Low Time	9.8		ns
3	Rise Time		1	ns
4	Fall Time		1	ns
	Max. frequency		48	MHz

**Table 9.5-3: Input Clock**



**Figure 9.5-1: Clock Timing**

### 9.5.2.2 Clock Outputs

The clock outputs (CLKOUT1X2 and CLKOUT1X4) are active during the RESET also.

For switched on PLL (XPLLEN = '0'):

In this mode, the two clock outputs are derived from the output clock of the PLL. The clock outputs thus have the inaccuracy of the PLL (frequency stability +/- 200 ppm; phase jitter: maximum of 3 ns). The electrical characteristics are shown in Table 9.6-4.

Parameter	Min	Max	Unit
Duty factor CLKOUT1X2 (Hi:Low)	35:65	65:35	
Duty factor CLKOUT1X4 (Hi:Low)	35:65	65:35	
Response time (of PLL)		1	ms
Frequency stability (of PLL)		±200	ppm

**Table 9.5-4: Clock Outputs**

For switched off PLL (XPLLEN = '1'):

In this mode, both clock outputs are derived from the input clock at the XTAL1\_CLK pin. The clock outputs thus have the inaccuracy of the pending clock. The data for the duty factor is shown in Table 9.6-4.

### 9.5.2.3 Reset

For switched on PLL (XPLLEN = '0'):

The reset does not reset the PLL. After voltage is restored, the reset duration should be a multiple of the build-up duration for safety reasons (see Table 9.6-4).

For switched off PLL (XPLLEN = '1'):

The reset duration should be greater than 40 clock cycles at the XTAL1\_CLK pin.

### 9.5.2.4 Interrupt

After acknowledging an interrupt with EOI, there is at least a 1us or 1 ms wait in the DPC31 prior to a new interrupt being read out.

No.	Parameters	Min	Max	Unit
1	Interrupt Inactive Time (if EOI_Timebase = 0)	1	2	μs
	Interrupt Inactive Time (if EOI_Timebase = 1)	1	2	ms

Table 9.5-5: Interrupt Inactive Time after EOI

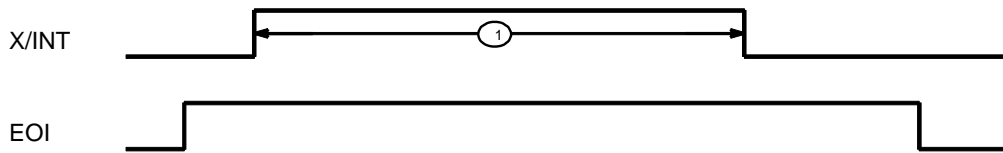


Figure 9.5-2: Peripheral Mode, Interrupt EOI Timing

Profibus Interface

No.	Parameters	Min	Max	Unit	
1	RTS ↑ to TXD Setup Time	XAsyn/Syn = low	7T	7T + T <sub>BIT</sub>	ns
		XAsyn/Syn = high	0		ns
2	RTS ↓ to TXD Hold Time	XAsyn/Syn = low	5T	6T	ns
		XAsyn/Syn = high	0		ns

T:= elementary period

T<sub>BIT</sub>: elementary period of the transition clock pulse of the Profibus Interface

XCTS\_RXA = '0'!

Table 9.5-6: Specification of the Profibus Interface



Figure 9.5-3: Transmit Timing, XCTS constant log. '0'

9.5.2.5 μP Interface

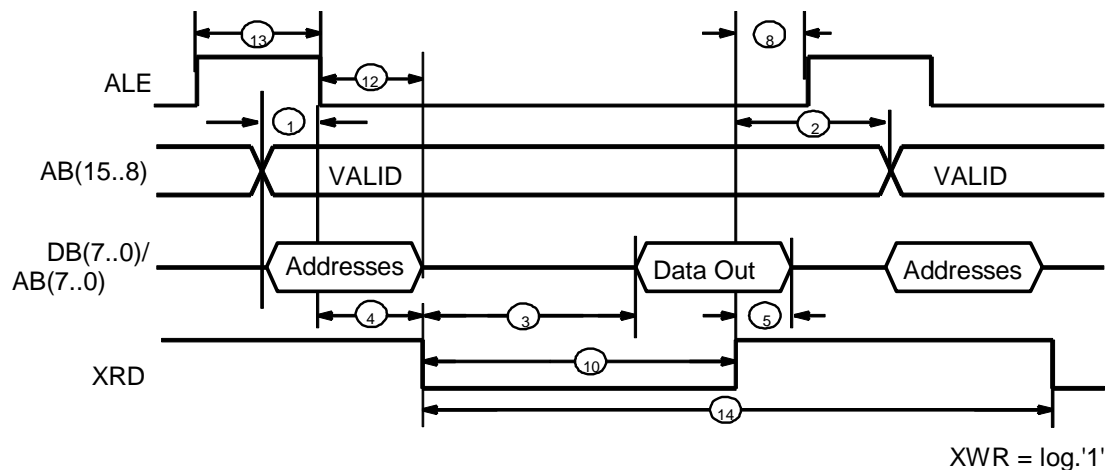
9.5.2.5.1 Synchronous Intel Mode (80C32)

No.	Parameters	Min	Max	Unit
1	Address to ALE ↓ Setup time	10		ns
2	Address (AB <sub>8..15</sub> ) hold time after XRD ↑ or XWR ↑	5		ns
3	XRD ↓ to Data Out (access to RAM)		4T+14	ns
	XRD ↓ to Data Out (access to the registers)		4T+14	ns
4	ALE ↓ to XRD ↓	20		ns
5	Data hold time after XRD ↑	1.5	5	ns
6	Data hold time after XWR ↑	10		ns
7	Data setup time to XWR ↑	10		ns
8	XRD ↑ to ALE ↑	10		ns
10	XRD Pulse Width	6T - 10		ns
11	XWR Pulse Width	4T		ns
12	Address hold time after ALE ↓	10		ns
13	ALE Pulse Width	10		ns
14	XRD, XWR cycle time	6T + 30		ns
15	ALE ↓ to XWR ↓	20		ns
16	XWR ↑ to ALE ↑	10		ns

**Table 9.5-7: Timing Values in the Synchronous Intel Mode**

In the synchronous Intel mode, the DPC31 stores the least significant address bits with the falling edge of ALE. At the same time, it expects the most significant address bits at the address bus; from them, it generates itself a chip select signal.

The request for an access to the DPC31 is generated from the falling edge of the read signal or the rising edge of the write signal.



**Figure 9.5-4: Synchronous Intel Mode, Processor Read Timing**

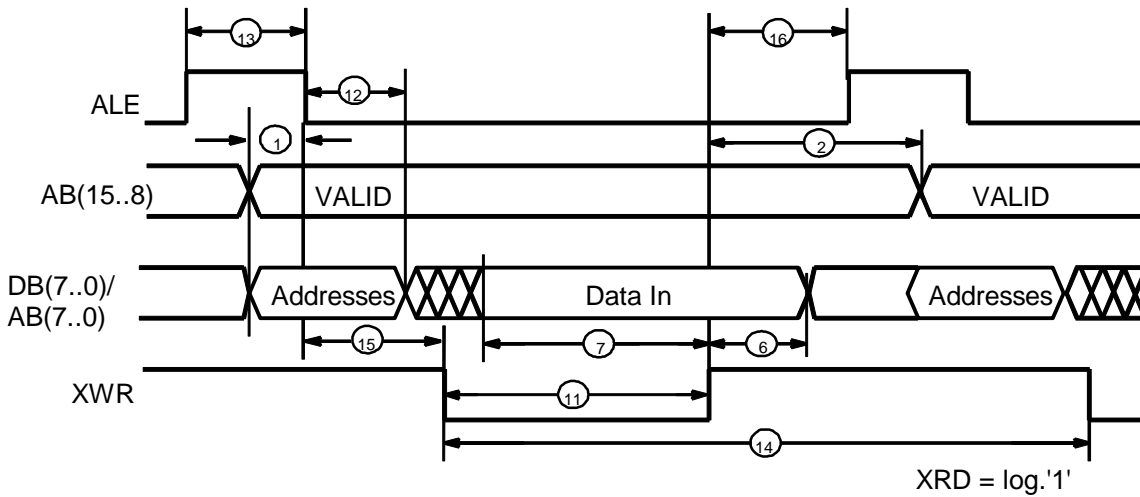


Figure 9.5-5: Synchronous Intel Mode, Processor Write Timing

9.5.2.5.2 Asynchronous Intel Mode (X86 Mode)

In 80X86 operation, the DPC31 in principle behaves like a memory with Ready logic; the access timing depends on the type of access. The request for an access to the DPC31 is generated from the falling edge of the Read signal or the rising edge of the Write signal.

No.	Parameters	Min	Max	Unit
20	Address setup time to XRD ↓ or XWR ↓	0		ns
21	XRD ↓ to Data valid (access to RAM)		4T+14	ns
	XRD ↓ to Data valid (access to the registers)		4T+14	ns
22	Address (AB <sub>12..0</sub> ) hold time after XRD or XWR ↑	0		ns
23	XCS ↓ Setup time to XRD ↓ or XWR ↓	0		ns
24	XRD Pulse Width	6T – 10		ns
25	Data hold time after XRD ↑	1.5	5	ns
26	Read/Write inactive Time	10		ns
27	XCS hold time after XRD ↑ or XWR ↑	0		ns
28	XRD/XWR ↓ to XRDY ↓ (normal Ready)	4T	5T + 14	ns
29	XRD/XWR ↓ to XRDY ↓ (early Ready)	3T	4T + 14	ns
30	XREADY hold time after XRD or XWR	7	19	ns
31	Data setup time to XWR ↑	10		ns
32	Data hold time after XWR ↑	10		ns
33	XWR Pulse Width	4T		ns
34	XRD, XWR cycle time	6T		ns
35	last XRD ↓ to XCS ↓	4T+10		ns
36	XCS ↑ to next XWR ↑	4T		ns
37	XWR ↑ to next XWR ↑ (XCS don't care)	6T		ns

Table 9.5-8: Timing Values in the Asynchronous Intel Mode

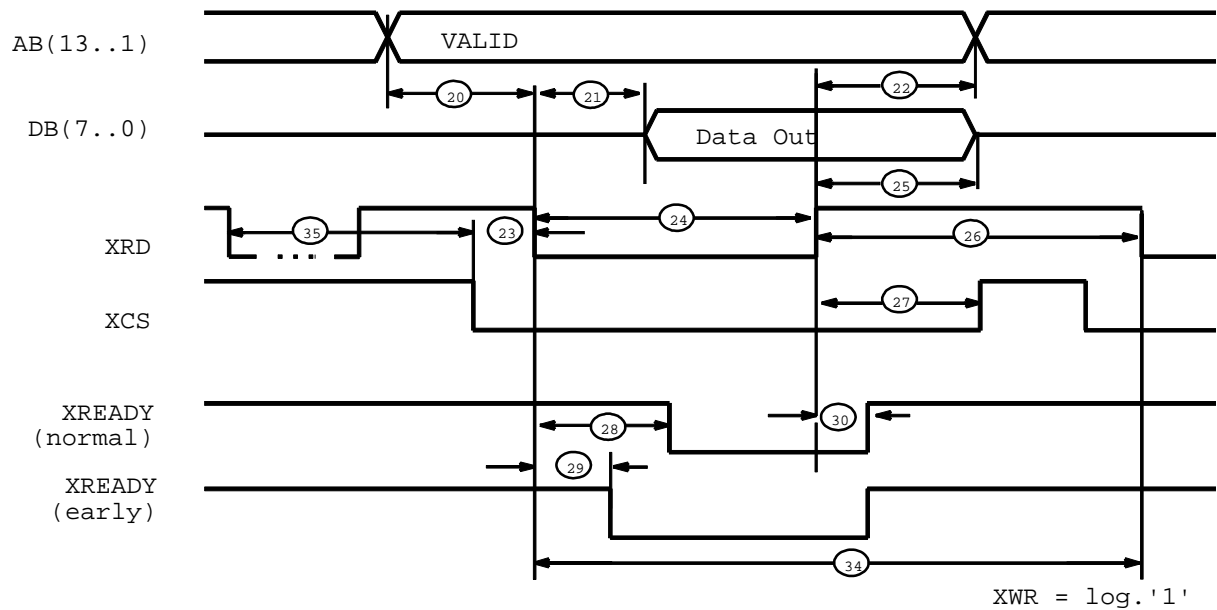


Figure 9.5-6: Asynchronous Intel Mode, Processor Read Timing

The Ready signal is generated by the DPC31 synchronously to the clock supplied and reset by the deactivation the Read or Write signal. With XRD = 1, the data bus is switched to Tristate.

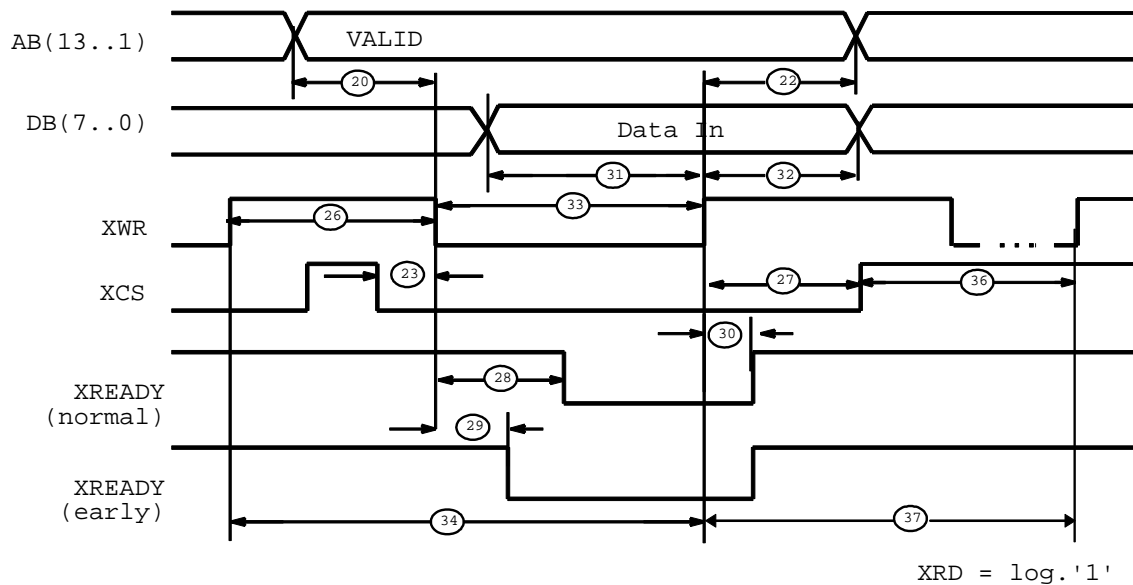


Figure 9.5-7: Asynchronous Intel Mode, Processor Write Timing

9.5.2.5.3 Synchronous Motorola Mode (E\_Clock mode; for example, 68HC11)

If the DPC31 supplies the CPU with the clock, the output clock has to be 4 times larger than the E\_CLOCK. The DPC31 input clock (CLK) has to be **at least 10 times** larger than the desired system clock (E\_Clock). Therefore, the clock output CLKOUT1x4 that specifies the E\_Clock of 3 MHz is to be used (asyn. physics).

The request for a read access to the DPC31 is generated from the rising edge of the E\_Clock (in addition: XCS = '0', R\_W = '1') and for a write access from the falling edge of the E\_Clock (in addition: XCS = '0', R\_W = '0').

No.	Parameters	Min	Max	Unit
40	E_Clock Pulse Width	$4T + 67$		ns
41	Address (AB <sub>12..0</sub> ) setup time to E_Clock ↑	10		ns
42	Address (AB <sub>12..0</sub> ) hold time after E_Clock ↓	5		ns
43	E_Clock ↑ to Data Active Delay	3.5	10	ns
44	E_Clock ↑ to Data valid (access to RAM)	3T	$4T + 14$	ns
	E_Clock ↑ to Data valid (access to the registers)	3T	$4T + 14$	ns
45	Data hold time after E_Clock ↓	2	5.5	ns
46	R_W setup time to E_Clock ↑	10		ns
47	R_W hold time after E_Clock ↓	5		ns
48	XCS setup time to E_Clock ↑	0		ns
49	XCS hold time after E_Clock ↓	0		ns
50	Data setup time to E_Clock ↓	10		ns
51	Data hold time after E_Clock ↓	10		ns

Table 9.5-9: Timing Values for the Synchronous Motorola Mode

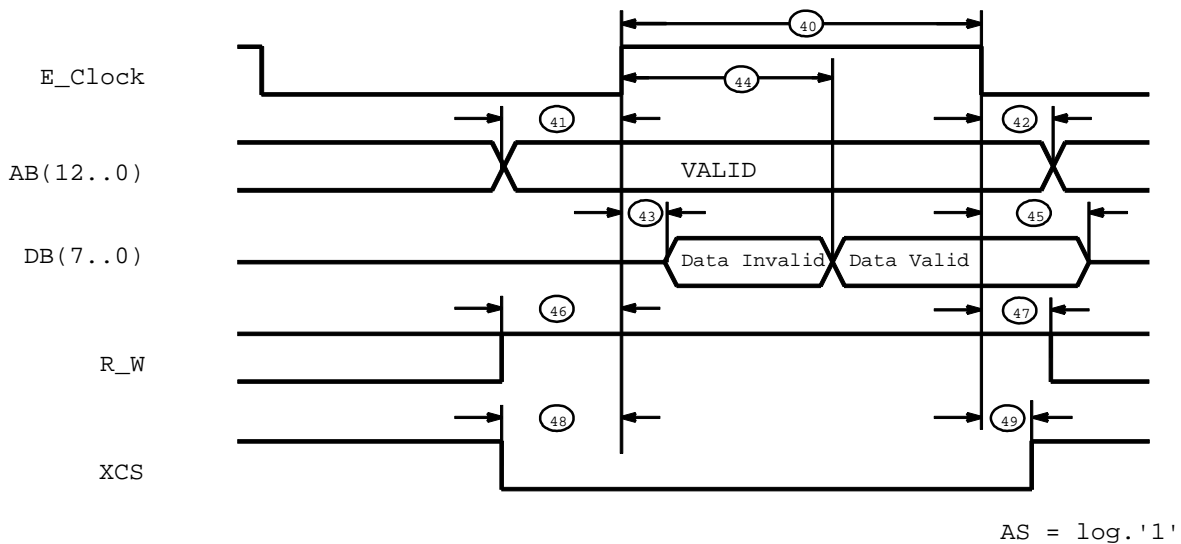
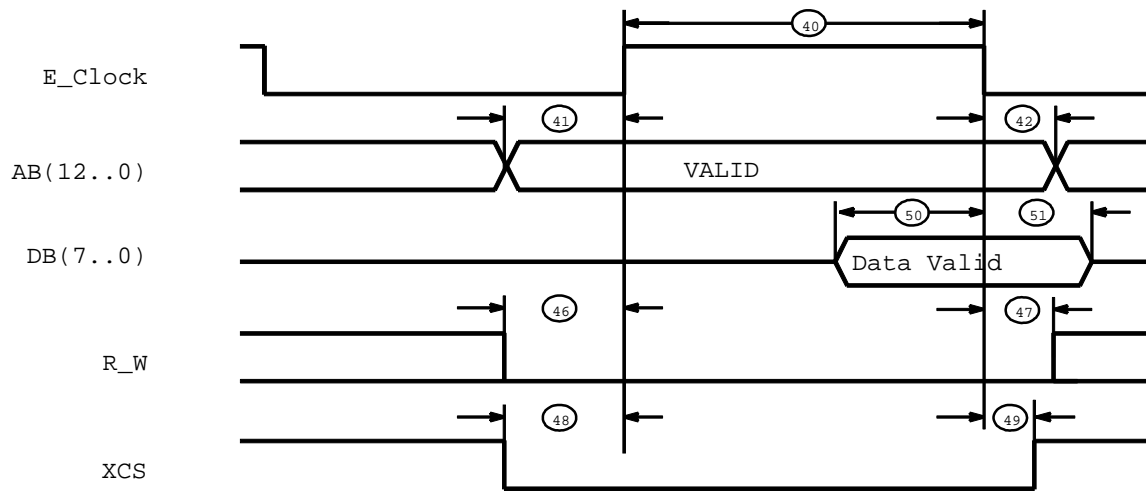


Figure 9.5-8: Synchronous Motorola Mode, Processor Read Timing



AS = log.'1'

**Figure 9.5-9: Synchronous Motorola Mode, Processor Write Timing**

### 9.5.2.5.4 Asynchronous Motorola Mode (for example, 68HC16)

In the asynchronous Motorola mode, the DPC31 behaves in principle like a memory with Ready logic and the access timing depends on the type of accesses.

The request for a Read access to the DPC31 is generated from the rising edge of the AS signal (in addition: XCS = '0', R\_W = '1') and for a write access from the rising edge of the AS signal (in addition: XCS = '0', R\_W = '0').

No.	Parameters	Min	Max	Unit
60	Address setup time to AS ↓	0		ns
61	AS ↓ to Data valid (access to RAM)		4T + 14	ns
	AS ↓ to Data valid (access to the registers)		4T + 14	ns
62	Address (AB <sub>12..0</sub> ) hold time after AS ↑	10		ns
63	R_W ↓ setup time to AS ↓	10		ns
64	AS Pulse Width	6T - 10		ns
65	Data hold time after AS ↑	2	5	ns
66	AS inactive time	10		ns
67	R_W hold time after AS ↑	10		ns
68	XCS ↓ setup time to AS ↓	-5		ns
69	XCS hold time after AS ↑	0		ns
70	AS ↓ to XDSACK ↓ (standard Ready)	4T	5T + 14	ns
71	AS ↓ to XDSACK ↓ (early Ready)	3T	4T + 14	ns
72	XDSACK hold time after AS ↑	7.5	19	ns
73	AS cycle time	6T		ns
74	Data setup time to AS ↑	10		ns
75	Data hold time after AS ↑	10		ns
76	AS Pulse Width	4T		ns
77	last AS ↓ (Read) to XCS ↓	4T+10		ns
78	XCS ↑ to next AS ↑ (Write)	4T		ns
79	AS ↑ to next AS ↑ (Write, XCS don't care)	6T		ns

**Table 9.5-10: Timing Values for the Asynchronous Motorola Mode**

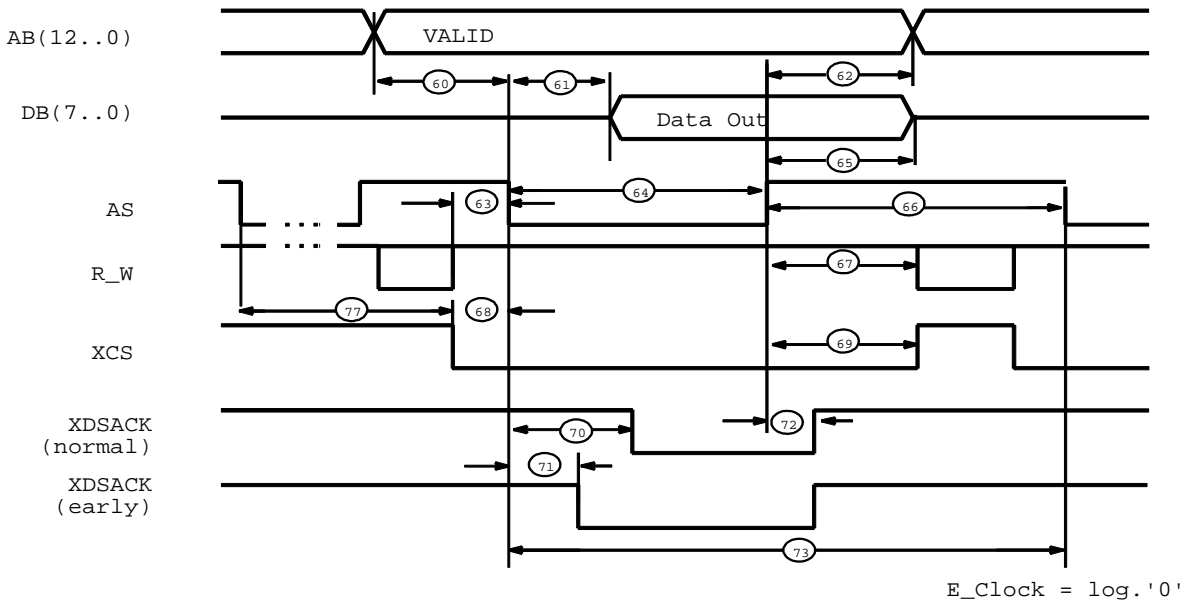


Figure 9.5-10: Asynchronous Motorola Mode, Processor Read Timing

The Ready signal XDSACK is generated by the DPC31 synchronously to the supplied clock pulse and it is reset with the deactivation of the AS signal. AS = 1 switches the data bus to Tristate.

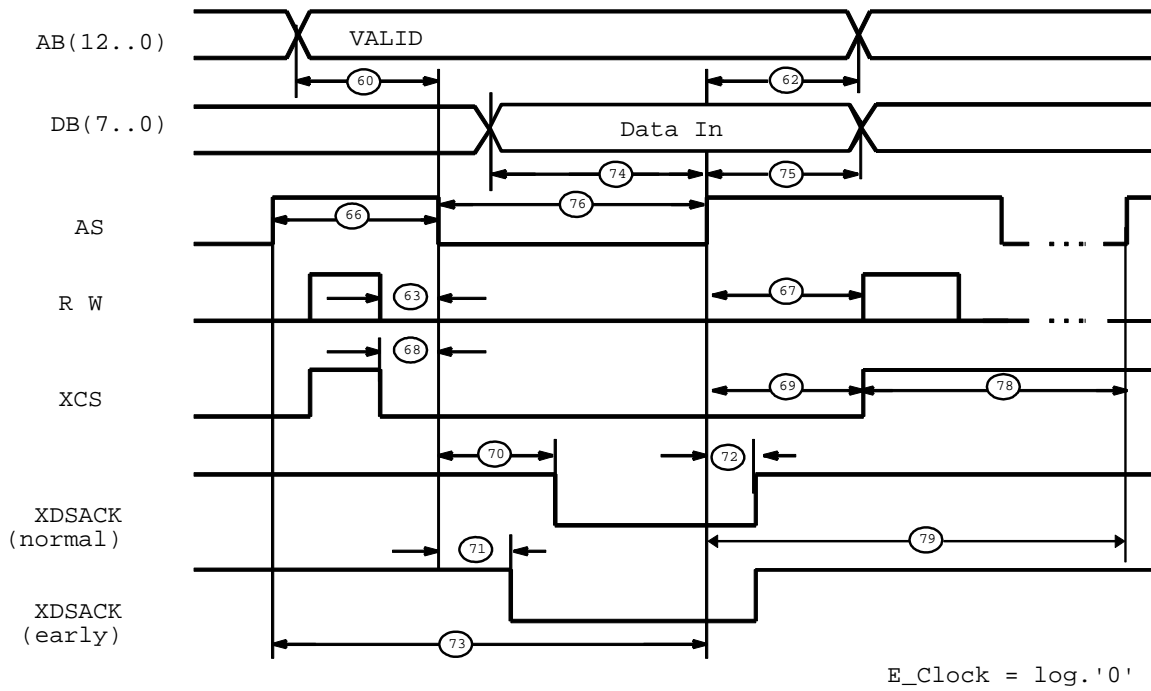


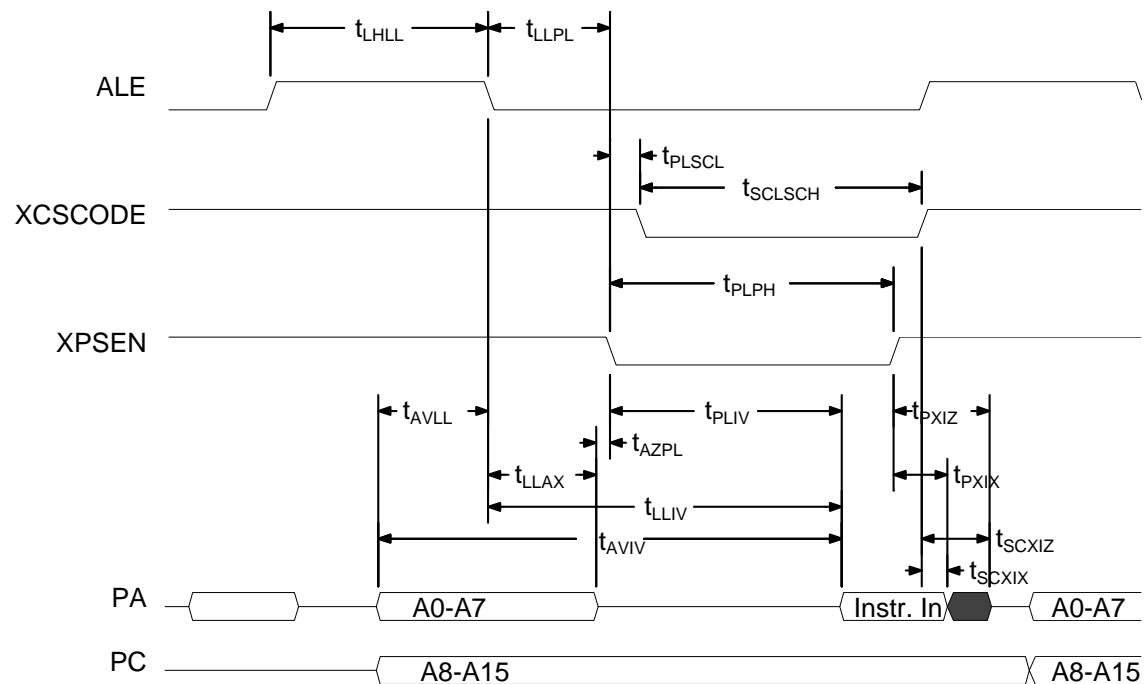
Figure 9.5-11: Asynchronous Motorola Mode, Processor Write Timing

## 9.5.2.6 C31 Memory Interface (internal C31 on external memory)

Symbol	Parameters	Min	Max	Unit
$t_{LHLL}$	ALE pulse width	4T - 4.5		ns
$t_{AVLL}$	Address setup to ALE	2T - 9.5		ns
$t_{LLAX}$	Address hold after ALE	2T - 4.3		ns
$t_{LLIV}$	ALE low to valid instr in		8T - 31.6	ns
$t_{LLPL}$	ALE to XPSEN	2T - 3.5		ns
$t_{PLPH}$	XPSEN pulse width	6T - 0.8		ns
$t_{PLIV}$	XPSEN to valid instr in		6T - 27.0	ns
$t_{PXIX}$	Input instruction hold after XPSEN	0		ns
$t_{PXIZ}$	Input instruction float after XPSEN		2T + 4.0	ns
$t_{AVIV}$	Address to valid instr in		10T - 45.6	ns
$t_{AZPL}$	Address float to XPSEN	0.3		ns
$t_{PLSCL}$	XPSEN to XCSCODE		5.0	ns
$t_{SCLSCH}$	XCSCODE pulse width	6T + 1.6		ns
$t_{SCXIX}$	Input instruction hold after XCSCODE	0		ns
$t_{SCXIZ}$	Input instruction float after XCSCODE		2T + 1.9	ns

( $C_L$  for Port A = 120pF;  $C_L$  for XPSEN = 10pF;  $C_L$  for all others = 80pF)

**Table 9.5-11: Timing Values for Accesses to Code Memory**



**Figure 9.5-12: Code Read Cycle**

Symbol	Parameters	Min	Max	Unit
$t_{LHLL}$	ALE pulse width	$4T - 4.5$		ns
$t_{RLRH}$	XRD pulse width	$12T + 1.8$		ns
$t_{WLWH}$	XWR pulse width	$12T + 1.9$		ns
$t_{LLAX2}$	Address hold after ALE	$4T + 4.3$		ns
$t_{RLDV}$	XRD to valid data in		$10T - 33.9$	ns
$t_{RHDX}$	Data hold after XRD	0		ns
$t_{RHDZ}$	Data float after XRD		$4T - 2.3$	ns
$t_{LLDV}$	ALE low to valid data in		$16T - 31.7$	ns
$t_{AVDV}$	Address to valid data in		$18T - 41.7$	ns
$t_{LLWL}$	ALE to XWR or XRD	$6T + 0.3$	$6T + 1.5$	ns
$t_{AVWL}$	Address valid to XWR or XRD	$8T - 9.6$		ns
$t_{WHLH}$	XWR or XRD high to ALE high	$2T - 1.8$	$2T - 0.3$	ns
$t_{QVWX}$	Data valid to XWR↓	$2T - 8.0$		ns
$t_{QVWH}$	Data setup to XWR	$14T - 3.3$		ns
$t_{WHQX}$	Data hold after XWR	$2T - 3.1$		ns
$t_{RLAZ}$	Address float after XRD		0	ns
$t_{AVSDL}$	Address valid to XCSDATA		14.0	ns
$t_{SDLSDH}$	XCSDATA pulse width	$24T - 0.4$		ns

( $C_L$  for Port A = 120pF;  $C_L$  for XPSEN = 10pF;  $C_L$  for all others = 80pF)

Table 9.5-12: Timing Values for Accesses to the Data Memory

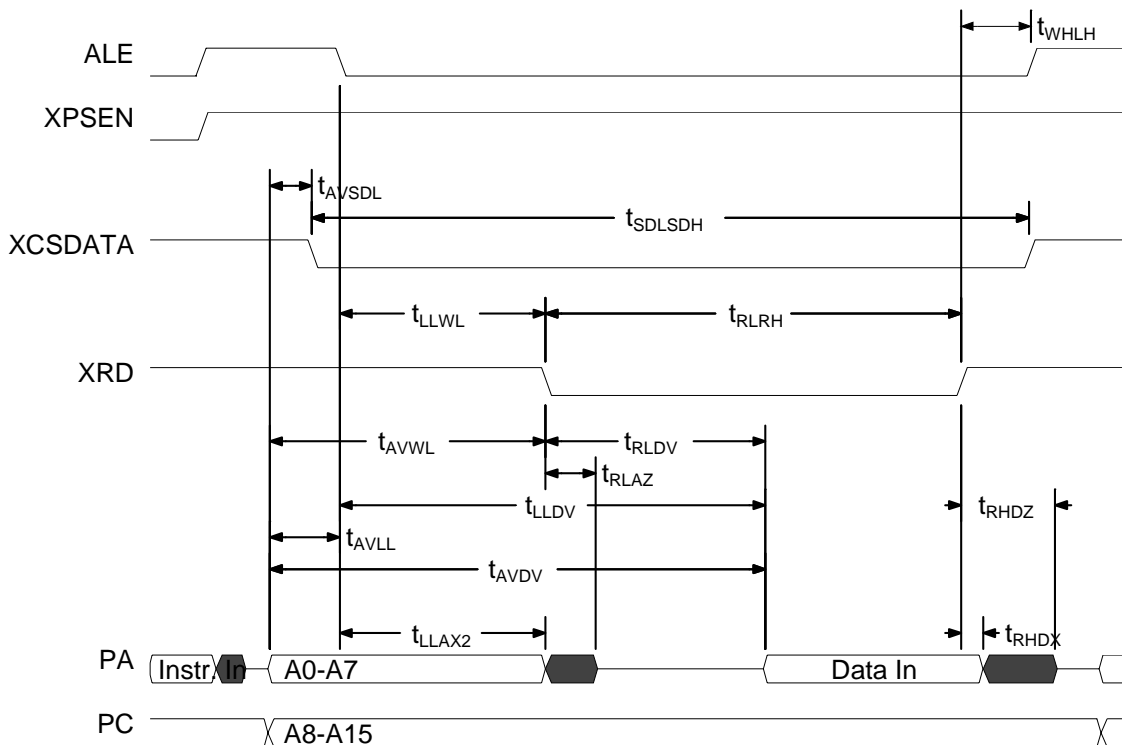


Figure 9.5-13: Data Read Cycle

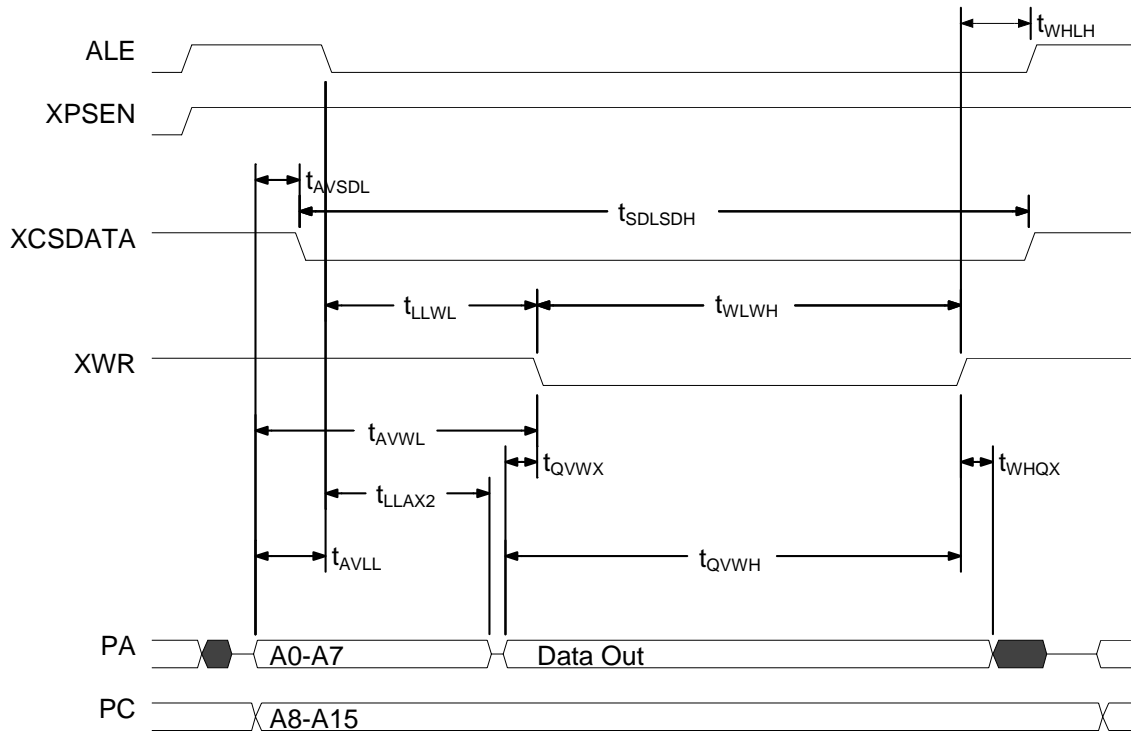


Figure 9.5-14: Data Write Cycle

### 9.5.2.7 Emulator-Interface

Symbol	Parameter	Min	Max	Unit
$t_{LHLL}$	ALE pulse width	3T		ns
$t_{AVLL}$	Address setup to ALE	1T		ns
$t_{LLAX}$	Address hold after ALE	1T		ns
$t_{LLPL}$	ALE to XPSEN	1T		ns
$t_{PLPH}$	XPSEN pulse width	5T		ns
$t_{LLIV}$	ALE↓ to valid instr out		4T + 24	ns
$t_{PXIX}$	Output instruction hold after XPSEN	2		ns
$t_{PXIZ}$	Output instruction float after XPSEN <sup>19</sup>		10	ns
$t_{AZPL}$	Address float to XPSEN <sup>19</sup>	0		ns
$t_{PLSCL}$	XPSEN↓ to XCSCODE↓	3	10	ns
$t_{PHSCH}$	XPSEN↑ to XCSCODE↑	5	15	ns

( $C_L$  for Port A = 120pF;  $C_L$  for XPSEN = 10pF;  $C_L$  for all others = 80pF)

Table 9.5-13: Timing Values for Emulator Accesses to Code Memory

<sup>19</sup> After this time, the output driver of the DPC31 is switched off. The signal level at the bust then depends on the bus load the size of the pull-up resistors.

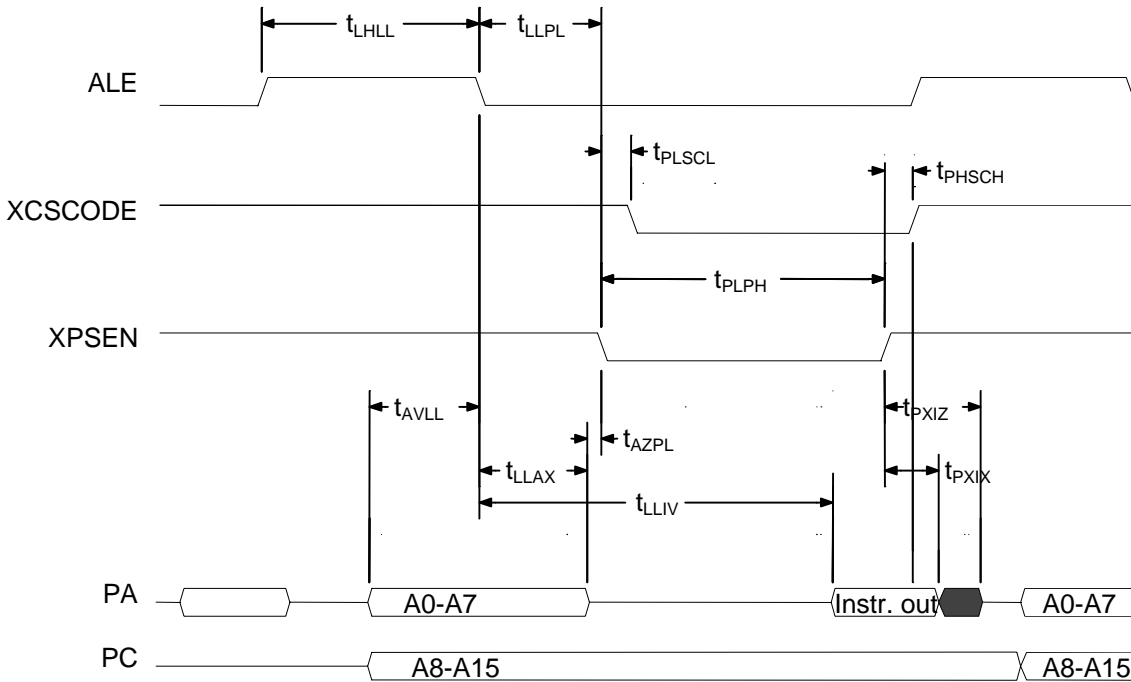


Figure 9.5-15: Emulator Code Read-Cycle

PA(=data bus) of DPC31 is only active if

- Internal code RAM is programmed
- Address created by emulator is < 1000H
- In this case, XCSCODE = konst. = 1

Symbol	Parameter	Min	Max	Unit
$t_{LHLL}$	ALE pulse width	3T		ns
$t_{AVLL}$	Address setup to ALE↓	1T		ns
$t_{RLRH}$	XRD pulse width	6T		ns
$t_{WLWH}$	XWR pulse width	6T		ns
$t_{LLAX2}$	Address hold after ALE↓	1T		ns
$t_{RLDV}$	XRD↓ to valid data out		4T + 24	ns
$t_{RHDX}$	Data hold after XRD↑	2		ns
$t_{RHDZ}$	Data float after XRD↑ <sup>20</sup>		10	ns
$t_{LLWL}$	ALE to XWR↓ or XRD↓	1T		ns
$t_{AVWL}$	Address valid to XWR↓ or XRD↓	2T		ns
$t_{WHLH}$	XWR↑ or XRD↑ high to ALE high	1T		ns
$t_{QVWH}$	Data setup to XWR↑	2T		ns
$t_{WHQX}$	Data hold after XWR↑	1T		ns
$t_{RLAZ}$	Address float after XRD↓ <sup>20</sup>		0	ns
$t_{AVSDL}$	Address valid to XCSDATA↓	3	15	ns

( $C_L$  for Port A = 120pF;  $C_L$  for XPSEN = 10pF;  $C_L$  for all others = 80pF)

Table 9.5-14: Timing Values for Emulator Accesses to Data Memory

<sup>20</sup> After this time, the output driver of the DPC31 is switched off. The signal level on the bus depends on the bus load and the size of the pull-up resistors.

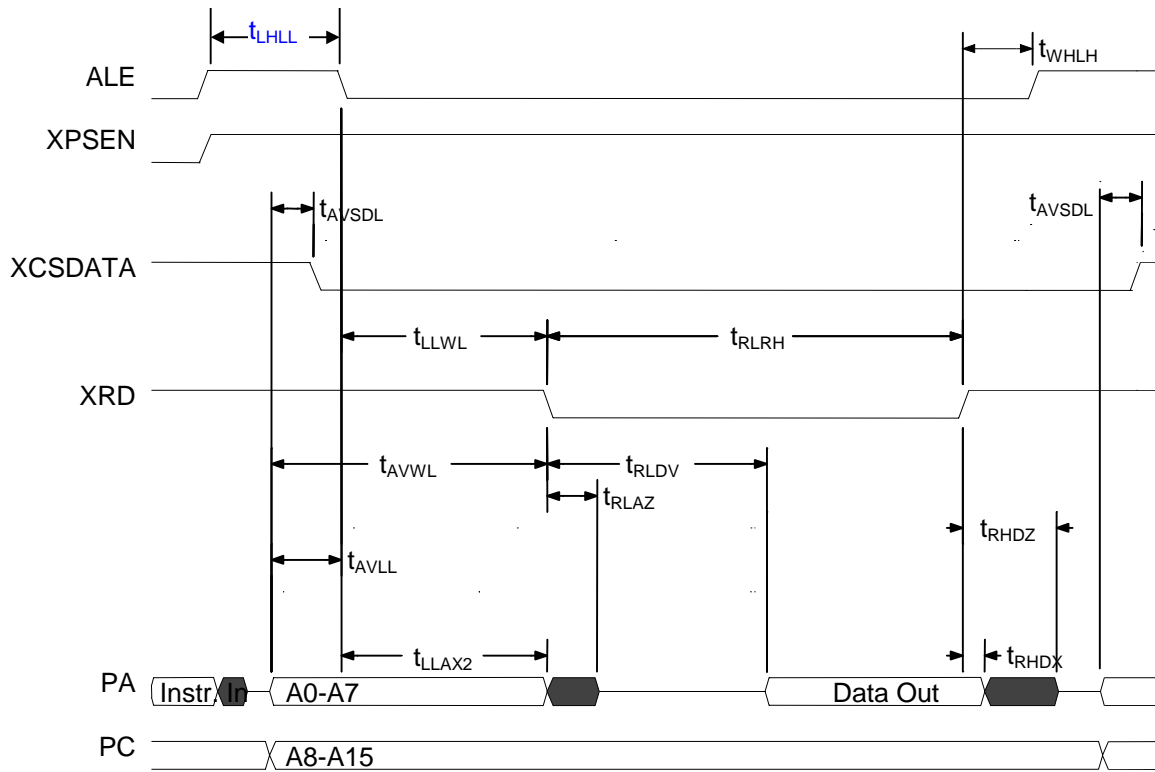


Figure 9.5-16: Emulator Data Read-Cycle

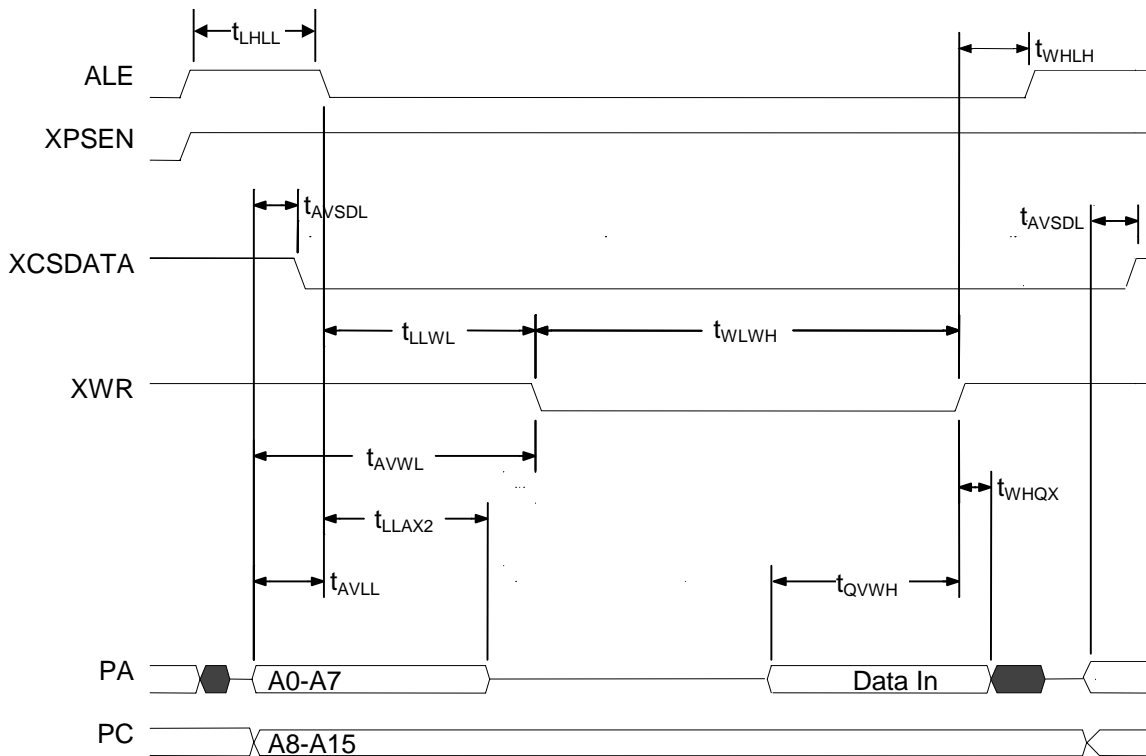


Figure 9.5-17: Emulator Data Write Cycle

PA (=data bus) of DPC31 is only active if

- Access is to the internal register or the area of the multiport RAM,
- In this case, XCSDATA = konst. = 1

### 9.5.2.8 SSC Interface (SPI)

Symbol	Parameters	Min	Max	Unit
$f_{SSCLK}$	Operating Frequency		12	MHz
$t_{CYC}$	Cycle Time	83,3		ns
$t_{WH}$	Clock High Time	40		ns
$t_{WL}$	Clock Low Time	40		ns
$t_{SU}$	Data Setup Time (Inputs)		28	ns
$t_H$	Data Hold Time (Inputs)	0		ns
$t_V$	Data Valid Time after Enable Edge		1,0	ns
$t_{HO}$	Data Hold Time (Outputs, after Enable Edge)	-1,0		ns

Table 9.5-15: Timing Values of the SSC Interface

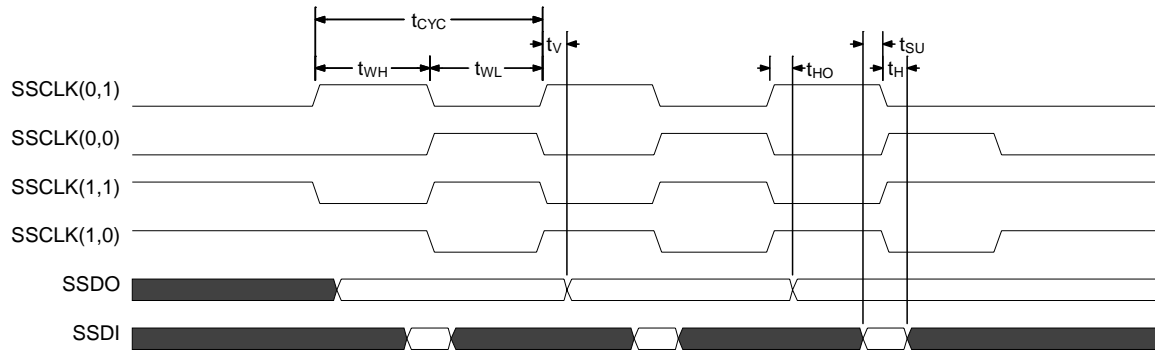
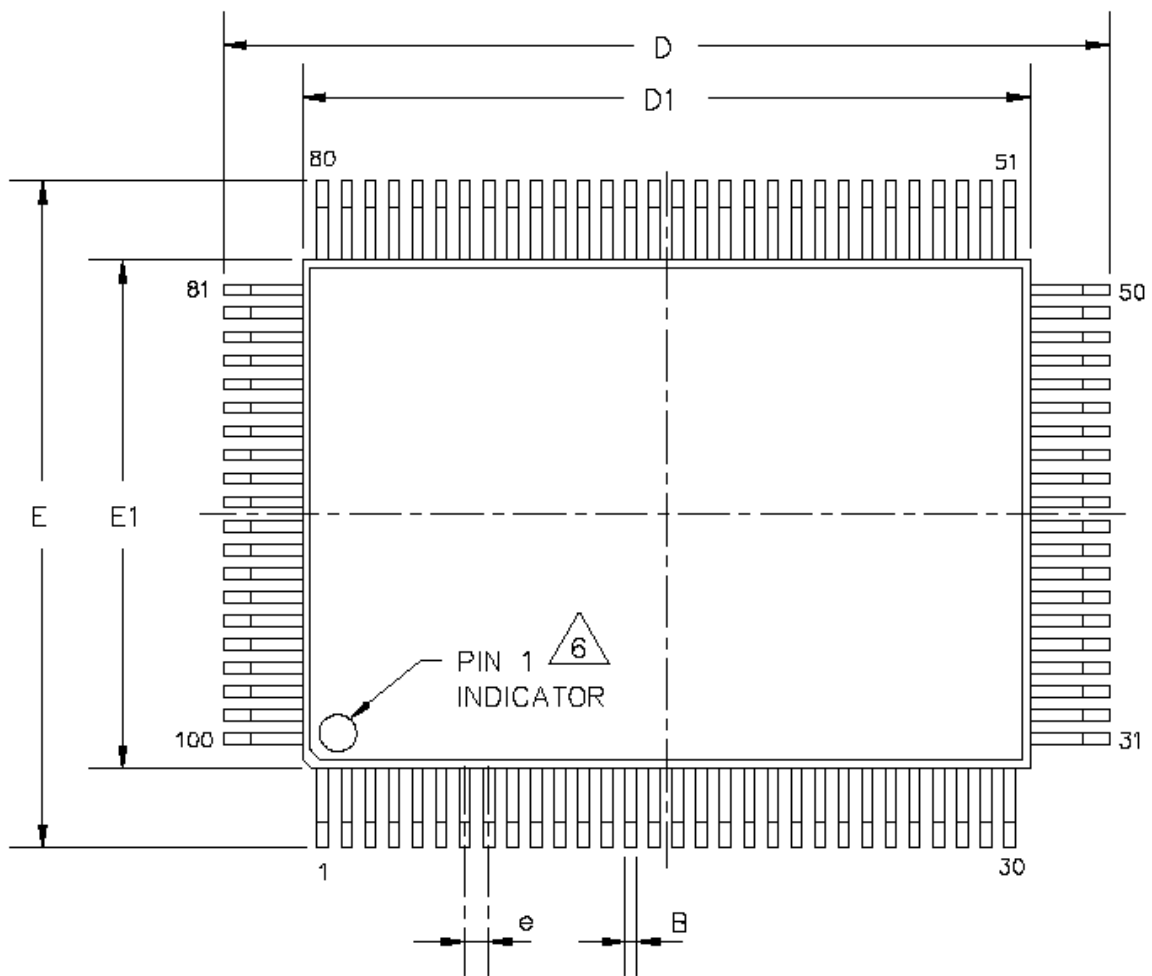


Figure 9.5-18: SSC Interface Timing Diagram

## 10 Mechanical Specification

### 10.1 PQFP 100 Casing

100 pin plastic QFP (14 x 20) pin pitch = 0.65mm (NEC CODE: S100GF-65-JBT)



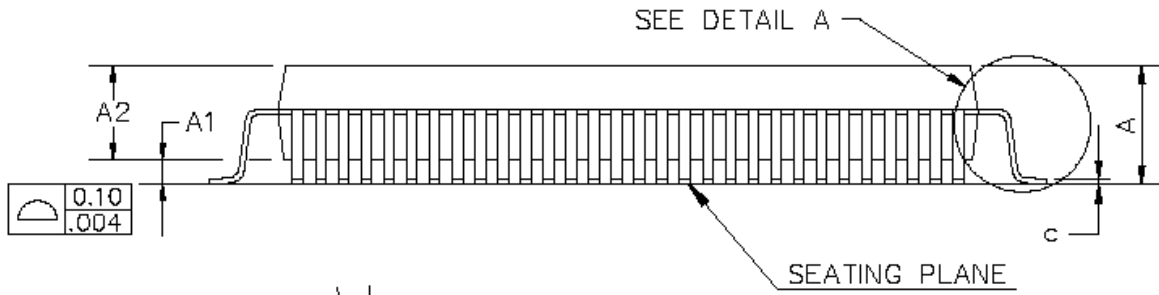


Figure 10.1-1: QFP-100 Housing

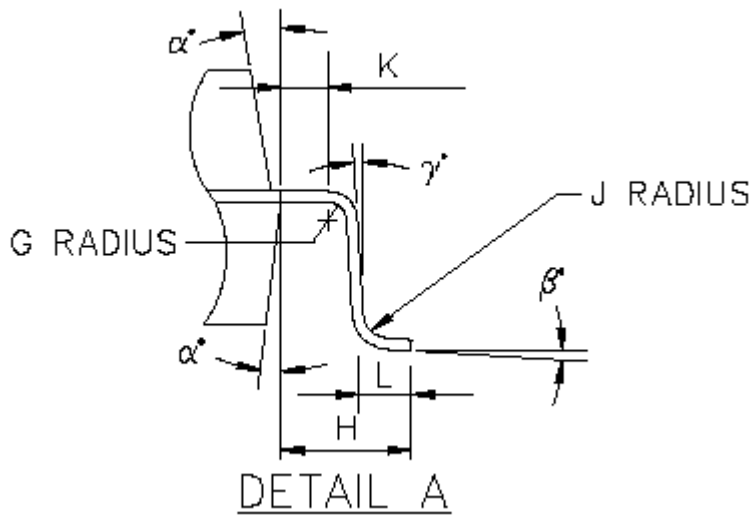


Figure 10.1-2: Detail A on Housing

ANAM 1.6 P/N 31767			
SYMBOL	MIN	NOM	MAX
A	—	3.04	3.40
A1	0.25	0.33	—
A2	2.57	2.71	2.87
D	22.95	23.20	23.45
D1	19.90	20.00	20.10
E	16.95	17.20	17.45
E1	13.90	14.00	14.10
L	0.65	0.70	0.95
e	0.65 BSC		
B	0.22	—	0.38
c	0.13	—	0.23
$\alpha^\circ$	12	—	16
$\beta^\circ$	0	—	7
$\gamma^\circ$	0	—	—
G	0.13	—	—
H	1.60 REF.		
J	—	0.30	—
K	0.40	—	—
2H	3.2		

**Table 10.1-1: Dimensions of PQFP Housing of AMI (in mm)**

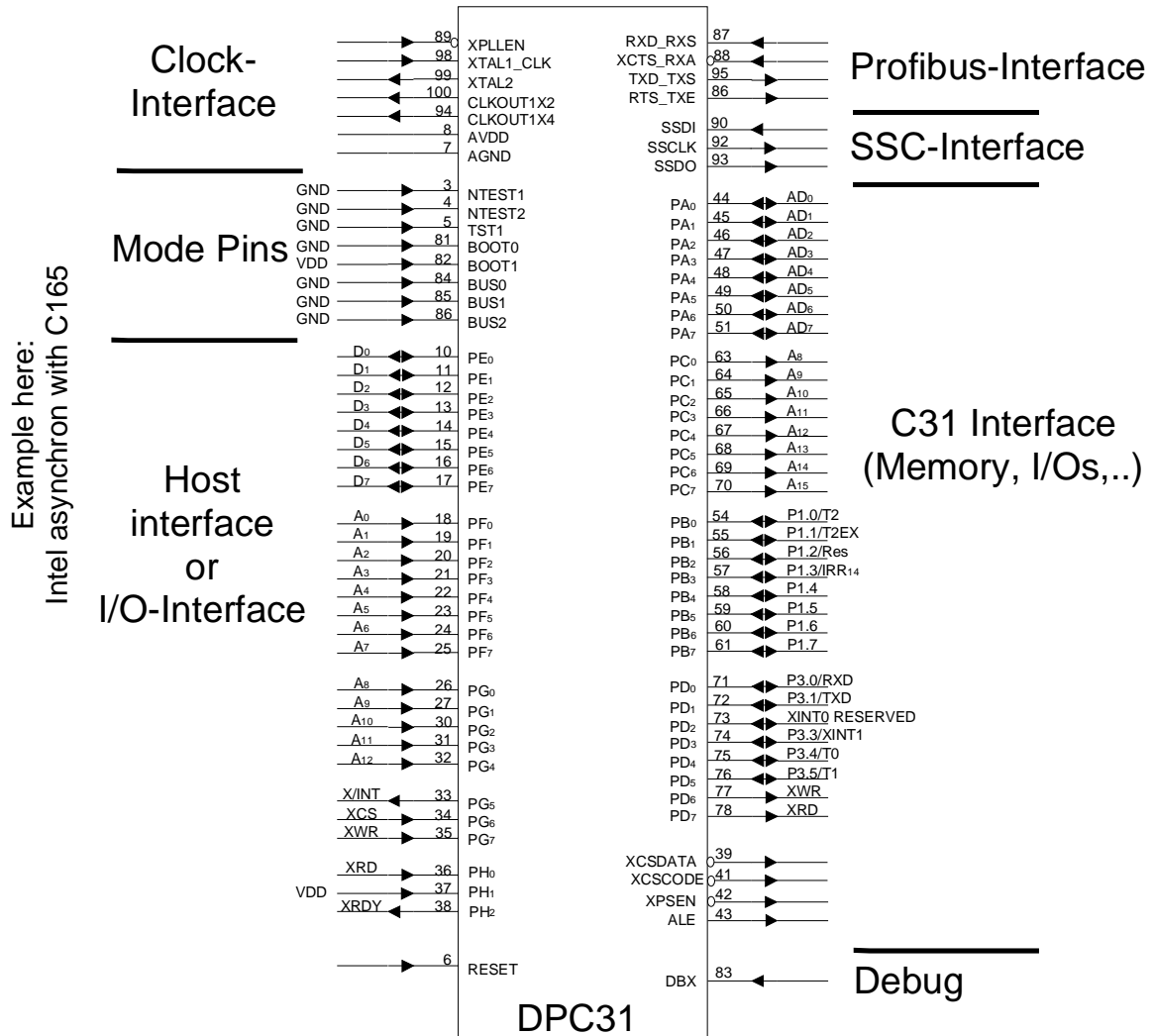
## 11 DPC31 Pinout

Pin	Name	Typ	Remarks	Pin	Name	Typ	Remarks
1	GND	Supply		51	PA <sub>7</sub>	In/Out	4mA
2	VDD	Supply		52	VDD	Supply	
3	NTEST1	In		53	GND	Supply	
4	NTEST2	In		54	PB <sub>0</sub>	In/Out	4mA
5	TST1	In		55	PB <sub>1</sub>	In/Out	4mA
6	RESET	In	Schmitt-Trig.	56	PB <sub>2</sub>	In/Out	4mA
7	AGND	Supply		57	PB <sub>3</sub>	In/Out	4mA
8	AVDD	Supply		58	PB <sub>4</sub>	In/Out	4mA
9	GND	Supply		59	PB <sub>5</sub>	In/Out	4mA
10	PE <sub>0</sub>	In/Out	9mA	60	PB <sub>6</sub>	In/Out	4mA
11	PE <sub>1</sub>	In/Out	9mA	61	PB <sub>7</sub>	In/Out	4mA
12	PE <sub>2</sub>	In/Out	9mA	62	GND	Supply	
13	PE <sub>3</sub>	In/Out	9mA	63	PC <sub>0</sub>	In/Out	4mA
14	PE <sub>4</sub>	In/Out	9mA	64	PC <sub>1</sub>	In/Out	4mA
15	PE <sub>5</sub>	In/Out	9mA	65	PC <sub>2</sub>	In/Out	4mA
16	PE <sub>6</sub>	In/Out	9mA	66	PC <sub>3</sub>	In/Out	4mA
17	PE <sub>7</sub>	In/Out	9mA	67	PC <sub>4</sub>	In/Out	4mA
18	PF <sub>0</sub>	In/Out	3mA	68	PC <sub>5</sub>	In/Out	4mA
19	PF <sub>1</sub>	In/Out	3mA	69	PC <sub>6</sub>	In/Out	4mA
20	PF <sub>2</sub>	In/Out	3mA	70	PC <sub>7</sub>	In/Out	4mA
21	PF <sub>3</sub>	In/Out	3mA	71	PD <sub>0</sub>	In/Out	4mA
22	PF <sub>4</sub>	In/Out	3mA	72	PD <sub>1</sub>	In/Out	4mA
23	PF <sub>5</sub>	In/Out	3mA	73	PD <sub>2</sub>	In/Out	4mA
24	PF <sub>6</sub>	In/Out	3mA	74	PD <sub>3</sub>	In/Out	4mA
25	PF <sub>7</sub>	In/Out	3mA	75	PD <sub>4</sub>	In/Out	4mA
26	PG <sub>0</sub>	In/Out	3mA	76	PD <sub>5</sub>	In/Out	4mA
27	PG <sub>1</sub>	In/Out	3mA	77	PD <sub>6</sub>	In/Out	4mA
28	GND	Supply		78	PD <sub>7</sub>	In/Out	4mA
29	VDD	Supply		79	VDD	Supply	
30	PG <sub>2</sub>	In/Out	3mA	80	GND	Supply	
31	PG <sub>3</sub>	In/Out	3mA	81	BOOTYP <sub>0</sub>	In	
32	PG <sub>4</sub>	In/Out	3mA	82	BOOTYP <sub>1</sub>	In	
33	PG <sub>5</sub>	In/Out	3mA	83	DBX	In	
34	PG <sub>6</sub>	In/Out	3mA	84	BUSTYP <sub>0</sub>	In	
35	PG <sub>7</sub>	In/Out	3mA	85	BUSTYP <sub>1</sub>	In	
36	PH <sub>0</sub>	In/Out	3mA	86	BUSTYP <sub>2</sub>	In	
37	PH <sub>1</sub>	In/Out	3mA	87	RXD_RXS	In	Schmitt-Trig.
38	PH <sub>2</sub>	In/Out	3mA	88	XCTS_RXA	In	Schmitt-Trig.
39	XCSDATA	Out	3mA	89	XPLEN	In	
40	GND	Supply		90	SSDI	In	
41	XCSCODE	Out	3mA	91	GND	Supply	
42	XPSEN	In/Out	3mA	92	SSCLK	Out	8mA
43	ALE	In/Out	3mA	93	SSDO	Out	8mA
44	PA <sub>0</sub>	In/Out	3mA	94	CLKOUT1X4	Out	8mA
45	PA <sub>1</sub>	In/Out	3mA	95	TXD_TXS	Out	8mA, 3.3V
46	PA <sub>2</sub>	In/Out	3mA	96	RTS_TXE	Out	8mA, 3.3V
47	PA <sub>3</sub>	In/Out	3mA	97	GND	Supply	
48	PA <sub>4</sub>	In/Out	3mA	98	XTAL1_CLK	In	
49	PA <sub>5</sub>	In/Out	3mA	99	XTAL2	Out	
50	PA <sub>6</sub>	In/Out	3mA	100	CLKOUT1X2	Out	8mA

Table 11-1: Pin Assignment of the QFP-100 Casing (signals starting with 'X' are low active)

## 12 Application Notes

### 12.1 DPC31 Wiring



### 12.2 PROFIBUS Interface

#### 12.2.1 Pin Assignment

Data is transmitted in the operating mode RS485 (RS485 physics).

The DPC31 is connected to the galvanically isolated interface driver via the following signals:

Signal Name	Input/Output	Function
RTS	Output	Request to Send
TXD	Output	Send Data
RXD	Input	Receive Data

The PROFIBUS interface is implemented as 9-pole SUB D connector with the following pin assignment:

- Pin 1 - free
- Pin 2 - free

Pin 3 - B line  
 Pin 4 - Request to Send (RTS)  
 Pin 5 - Ground 5V (**M5**)  
 Pin 6 - Potential 5V (**potential free P5**)  
 Pin 7 - free  
 Pin 8 - A line  
 Pin 9 - free

The line shield is to be connected to the connector housing.

The free pins are used optionally in the EN 50170 Vol.2 and should correspond to this description if the user uses them.

**Attention:**

The designations **A** and **B** for the lines at the connector correspond to the names in the RS485 standard and not to the pin name of driver ICs.

The line length from the driver to the connector is to be kept as short as possible.

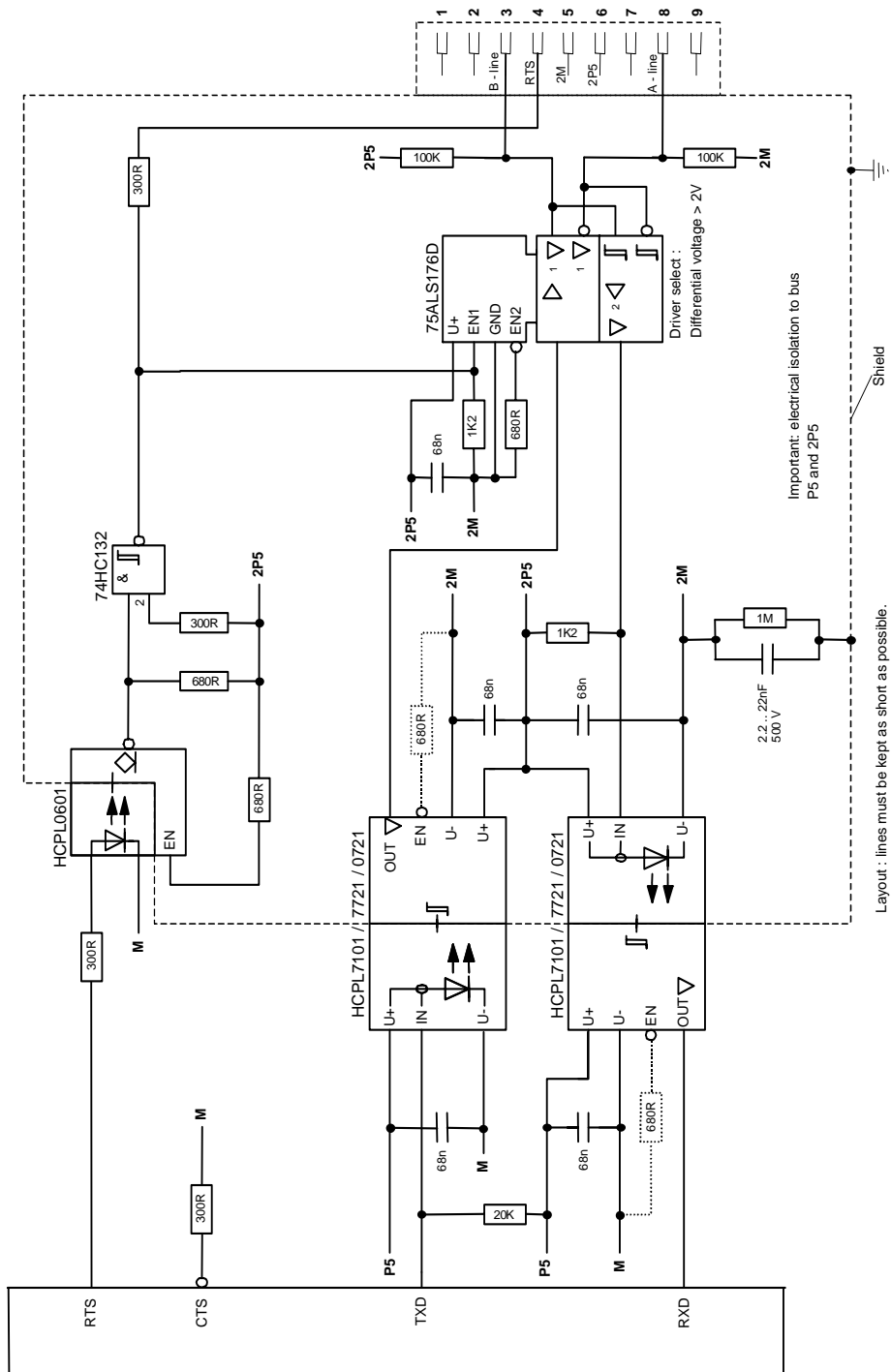
If the higher baudrates of 3 to 12 MBaud are used, suitable connectors are to be used. These connectors compensate for line influences regarding all possible line combinations.

**12.2.2 Optimizations of the bus cycle**

For optimizations of the bus cycle the following adjustments of the max TSDR timings can be done in the GSD file.

Transmission rate (kbit/s)	187.5	500	1500	3000	6000	12000
Optimized Max TSDR	15	15	25	50	100	200

## 12.2.2 Wiring Example RS485 Interface



**Explanation of the Circuit:**

At the bus driver 75ALS176D, the EN2 input is to be connected to ground so that the DPC31 can listen in during transmission.

No additional filters are to be installed in the send and receive line in order to keep the capacity of the lines as low as possible (15 .. 25 pF).

## 13 Appendix

### 13.1 Addresses

**PROFIBUS Trade Organization**

PNO Office  
Haid-und-Neu-Strasse 7  
76131 Karlsruhe/Germany  
Phone: (0721) 9658-590

**Technical Contact Persons at the Interface Center in Germany**

Siemens AG  
A&D SE RD73  
Mr. Putschky

Mailing Address:  
Postfach 2355  
90713 Fuerth/Germany

Street Address:  
Wuerzburger Strasse 121  
90766 Fuerth/Germany

Phone: (0911) 750 2078  
Fax: (0911) 750 2100

E-Mail:  
Gerd.Putschky@siemens.com

**Technical Contact Persons at the Interface Center in the USA**

PROFIBUS Interface Center  
One Internet Plaza  
Johnson City, TN 37602-4991

Fax: (423) 461-2103

Your Partner:  
Phone: (423) 461-2576

E-Mail:  
[profibus.center@sea.siemens.com](mailto:profibus.center@sea.siemens.com)

### 13.2 General Definitions of Terms

ASPC2	Advanced Siemens PROFIBUS Controller, 2 <sup>nd</sup> Generation
DPS	DP Slave
Din	Input Data

Dout	Output Data
MAC	Medium Access Control
MSAC1	Master Slave Acyclic Communication Class1 Master
SPC2	Siemens PROFIBUS Controller, 2 <sup>nd</sup> Generation
SPC3	Siemens PROFIBUS Controller, 3 <sup>rd</sup> Generation
SPM2	Siemens PROFIBUS Multiplexer, 2 <sup>nd</sup> Generation
LSPM2	Lean Siemens PROFIBUS Multiplexer, 2 <sup>nd</sup> Generation
DP	Distributed IO
FMS	Fieldbus Message Specification
MS	Micro-Sequencer
PLL	Phase Lock Loop
SM	State Machine

### 13.3 Order Numbers

The DPC31 Step C can be ordered via your Siemens contact person on location. Please use the order numbers with the number of units reference provided below:

Product	Order Number	Delivery Units	No. of Units
ASIC DPC 31	6ES7 195-0BF00-0XA0	Mini Package.	6
	6ES7 195-0BF10-0XA0	Single Tray	66
	6ES7 195-0BF20-0XA0	Tray Box	660
	6ES7 195-0BF301-0XA0	7-Tray Box	4620
FW DPV1 DPC 31	6ES7 195-2BB00-0XA0	Diskette	



Siemens AG  
Division Automation Engineering  
Combination Engineering  
PO Box 23 55, D-90713 Fuerth/Germany  
**SIEMENS Aktiengesellschaft**

© Siemens AG  
Subject to change without prior notice

---

Printed in the Fed. Rep. of Germany  
Order No. J31070-E2257-R300-A1-0009