The MAC status can be observed at any time in the **Status Register**.

Various events (e.g. various indications, error events, etc.) are entered in the **Interrupt Controller**. These events can be individually enabled via a mask register. Acknowledgement takes place by means of the acknowledge register. The VPC3+S has a common interrupt output.

The integrated **Watchdog Timer** is operated in three different states: BAUD_SEARCH, BAUD_CONTROL and DP_CONTROL.

The **Micro Sequencer** (MS) controls the entire process. It contains the DP-Slave state machine (DP_SM).

The integrated **4K Byte RAM** that operates as a Dual-Port-RAM contains procedure-specific parameters (buffer pointer, buffer lengths, Station_Address, etc.) and the data buffers.

In the **UART**, the parallel data flow is converted into the serial data flow and vice-versa. The VPC3+S is capable of automatically identifying the baud rates (9.6 Kbit/s - 12 Mbit/s).

The **Idle Timer** directly controls the bus times on the serial bus line.

The **IsoM-PLL** provides high-precision synchronization mechanisms as defined in the PROFIBUS DPV2 protocol extension.

3.2.5 SPI Mode

The VPC3+S can be interfaced like an SPI compatible memory device. Depending on the setting of CPOL and CPHA four different SPI modes can be selected. All unused inputs (including DB[7:0]) must be connected to GND.

Ball BGA	Pin QFP	Signal Name	In/Out	Description	Connect to
E3	9	SERMODE	I	'1': Serial Interface	**VCC**
E4	28	MOT/XINT	I	'0': not used in this mode	**GND**
D4	33	MODE	I	'0': SPI Mode	**GND**
C2	2	SPI_CPOL	I	Clock Polarity	VCC or GND
B3	44	SPI_CPHA	I	Clock Phase	VCC or GND
C1	3	SPI_XSS	I	Slave-Select Signal (active low)	CPU Slave-Select
A1	48	SPI_SCK	I(S)	Serial Clock	CPU SCK
B1	1	SPI_MOSI	I	Master-Out-Slave-In (Serial Data Input)	CPU MOSI
A2	47	SPI_MISO	O	Master-In-Slave-Out (Serial Data Output)	CPU MISO

Figure 3-8: Interface Configuration: SPI Mode

3.2.6 I2C Mode

The VPC3+S can be interfaced like an I2C compatible memory device. The VPC3+S is always in slave mode, master mode is not supported. The slave address can be configured by using the AB[6:0] inputs. All unused inputs (including DB[7:0]) must be connected to GND.

Ball BGA	Pin QFP	Signal Name	In/Out	Description	Connect to
E3	9	SERMODE	I	'1': Serial Interface	**VCC**
E4	28	MOT/XINT	I	'0': not used in this mode	**GND**
D4	33	MODE	I	'1': I2C Mode	**VCC**
C3	45	I2C_SA6	I	I2C Slave Address	VCC or GND
B2	46	I2C_SA5	I		VCC or GND
B4	41	I2C_SA4	I		VCC or GND
A5	38	I2C_SA3	I		VCC or GND
A6	37	I2C_SA2	I		VCC or GND
B5	39	I2C_SA1	I		VCC or GND
B6	36	I2C_SA0	I		VCC or GND
A1	48	I2C_SCK	I(S)	Serial Clock	CPU SCK
A2	47	I2C_SDA	I(S) / O	Serial Data Line	CPU SDA

Figure 3-9: Interface Configuration: I2C Mode

Memory Organization 4

4.1 Overview

The internal Control Parameters are located in the first 21 addresses. The latches/registers either come from the internal controller or influence the controller. Certain cells are read- or write-only. The internal working cells, which are not accessible by the user, are located in RAM at the same address locations.

The Organizational Parameters are located in RAM beginning with address 16H. The entire buffer structure (for the DP-SAPs) is based on these parameters. In addition, general parameter data (Station_Address, Ident_Number, etc.) and status information (Global_Control command, etc.) are also stored in these cells.
Corresponding to the parameter setting of the Organizational Parameters, the user-generated buffers are located beginning with address 40H. All buffers or lists must begin at segment addresses (8 bytes segmentation for 2K Byte mode, 16 bytes segmentation for 4K Byte mode).

<table>
<tr><th>Address</th><th colspan="3">Function</th></tr>
<tr><td>000H
:
015H</td><td colspan="2">Control Parameters
(latches/registers) (21 bytes)</td><td>Internal working cells</td></tr>
<tr><td>016H
:
03FH</td><td colspan="3">Organizational Parameters (42 bytes)</td></tr>
<tr><td>040H

:
:

7FFH (FFFH)</td><td colspan="3">DP-buffers: Data in (3)*
Data out (3)**
Diagnosis data(2)
Parameter data (1)
Configuration data (2)
Auxiliary buffers (2)
SSA-buffer (1)
DP-V1-buffer: SAP-List (1)
Indication / Response buffers ***
DP-V2-buffer: DXB out (3)****
DXB-buffers (2)
CS-buffer (1)
PLL-buffer (1)</td></tr>
</table>

Figure 4-1: Memory Table

* Data in means input data from DP-Slave to DP-Master
** Data out means output data from DP-Master to DP-Slave
*** Number of buffers depends on the entries in the SAP-List
**** DXB out means input data from another DP-Slave (slave-to-slave communication)

4.2 Control Parameters (Latches/Registers)

These cells can be either read-only or write-only. In the Motorola Mode the VPC3+S carries out ‘address swapping’ for an access to the address locations 00H - 07H (word registers). That is, the VPC3+S internally generates an even address from an odd address and vice-versa.

Address Intel	Address Mot.	Name	Bit No.	Significance (Read Access!)
00H	01H	Int-Req-Reg	7..0	Interrupt Controller Register
01H	00H	Int-Req-Reg	15..8	
02H	03H	Int–Reg	7..0	
03H	02H	Int–Reg	15..8	
04H	05H	Status-Reg	7..0	Status Register
05H	04H	Status-Reg	15..8	
06H	07H	Mode-Reg 0	7..0	Mode Register 0
07H	06H	Mode-Reg 0	15..8	
08H		Din_Buffer_SM	7..0	Buffer assignment of the DP_Din_Buffer_State_Machine
09H		New_Din_Buffer_Cmd	1..0	The user makes a new DP Din_Buf available in the N state.
0AH		Dout_Buffer_SM	7..0	Buffer assignment of the DP_Dout_Buffer_State_Machine
0BH		Next_Dout_Buffer_Cmd	3..0	The user fetches the last DP Dout_Buf from the N state
0CH		Diag_Buffer_SM	3..0	Buffer assignment for the DP_Diag_Buffer_State_Machine
0DH		New_Diag_Buffer_Cmd	1..0	The user makes a new DP Diag_Buf available to the VPC3+S.
0EH		User_Prm_Data_Okay	1..0	The user positively acknowledges the user parameter setting data of a Set_(Ext_)Prm telegram.
0FH		User_Prm_Data_Not_Okay	1..0	The user negatively acknowledges the user parameter setting data of a Set_(Ext_)Prm telegram.
10H		User_Cfg_Data_Okay	1..0	The user positively acknowledges the configuration data of a Chk_Cfg telegram.
11H		User_Cfg_Data_Not_Okay	1..0	The user negatively acknowledges the configuration data of a Chk_Cfg telegram.
12H		DXBout_Buffer_SM	7..0	Buffer assignment of the DXBout_Buffer_State_Machine
13H		Next_DXBout_Buffer_Cmd	2..0	The user fetches the last DXBout_Buf from the N state
14H		SSA_Buffer_Free_Cmd		The user has fetched the data from the SSA_Buf and enables the buffer again.
15H		Mode-Reg 1	7..0	

Figure 4-2: Assignment of the Internal Parameter-Latches for READ

Address Intel	Address Mot.	Name	Bit No.	Significance (Write Access!)
00H	01H	Int-Req-Reg	7..0	Interrupt-Controller-Register
01H	00H	Int-Req_Reg	15..8	
02H	03H	Int-Ack-Reg	7..0	
03H	02H	Int-Ack-Reg	15..8	
04H	05H	Int–Mask-Reg	7..0	
05H	04H	Int–Mask-Reg	15..8	
06H	07H	Mode-Reg0	7..0	Setting parameters for individual bits
07H	06H	Mode-Reg0	15..8	
08H		Mode-Reg1-S	7..0	
09H		Mode-Reg1-R	7..0	
0AH		WD_BAUD_CONTROL_Val	7..0	Square-root value for baud rate monitoring
0BH		$minT_{SDR}$_Val	7..0	$minT_{SDR}$ time
0CH		Mode-Reg2	7..0	Mode Register 2
0DH		Sync_PW_Reg	7..0	Sync Pulse Width Register
0EH		Control_Command_Reg	7..0	Control_Command value for comparison with SYNCH telegram
0FH		Group_Select_Reg	7..0	Group_Select value for comparison with SYNCH telegram
10H		Reserved		
11H				
12H		Mode-Reg3	7..0	Mode Register 3
13H		Reserved		
14H				
15H				

Figure 4-3: Assignment of the Internal Parameter-Latches for WRITE

4.3 Organizational Parameters (RAM)

The user stores the organizational parameters in the RAM under the specified addresses. These parameters can be written and read.

Address Intel	Address Mot.	Name	Bit No.	Significance
16H		R_TS_Adr		Setup Station_Address of the VPC3+S
17H		SAP_List_Ptr		Pointer to a RAM address which is preset with FFh or to SAP-List
18H	19H	R_User_WD_Value	7..0	In DP_Mode an internal 16-bit watchdog timer monitors the user.
19H	18H	R_User_WD_Value	15..8	
1AH		R_Len_Dout_Buf		Length of the 3 Dout_Buf
1BH		R_Dout_Buf_Ptr1		Segment base address of Dout_Buf 1
1CH		R_Dout_Buf_Ptr2		Segment base address of Dout_Buf 2
1DH		R_Dout_Buf_Ptr3		Segment base address of Dout_Buf 3
1EH		R_Len_Din_Buf		Length of the 3 Din_Buf
1FH		R_Din_Buf_Ptr1		Segment base address of Din_Buf 1
20H		R_Din_Buf_Ptr2		Segment base address of Din_Buf 2
21H		R_Din_Buf_Ptr3		Segment base address of Din_Buf 3
22H		R_Len_DXBout_Buf		Length of the 3 DXBout_Buf
23H		R_DXBout_Buf_Ptr1		Segment base address of DXBout_Buf 1
24H		R_Len Diag_Buf1		Length of Diag_Buf 1
25H		R_Len Diag_Buf2		Length of Diag_Buf 2
26H		R_Diag_Buf_Ptr1		Segment base address of Diag_Buf 1
27H		R_Diag_Buf_Ptr2		Segment base address of Diag_Buf 2
28H		R_Len_Cntrl_Buf1		Length of Aux_Buf 1 and the corresponding control buffer, for example SSA_Buf, Prm_Buf, Cfg_Buf, Read_Cfg_Buf
29H		R_Len_Cntrl_Buf2		Length of Aux_Buf 2 and the corresponding control buffer, for example SSA_Buf, Prm_Buf, Cfg_Buf, Read_Cfg_Buf
2AH		R_Aux_Buf_Sel		Bit array; defines the assignment of the Aux_Buf 1 and 2 to the control buffers SSA_Buf, Prm_Buf, Cfg_Buf
2BH		R_Aux_Buf_Ptr1		Segment base address of Aux_Buf 1
2CH		R_Aux_Buf_Ptr2		Segment base address of Aux_Buf 2
2DH		R_Len_SSA_Data		Length of the input data in the Set_Slave_Address_Buf
2EH		R_SSA_Buf_Ptr		Segment base address of the Set_Slave_Address_Buf
2FH		R_Len_Prm_Data		Length of the input data in the Prm_Buf

Address Intel	Address Mot.	Name	Bit No.	Significance
30H		R_Prm_Buf_Ptr		Segment base address of the Prm_Buf
31H		R_Len_Cfg_Data		Length of the input data in the Cfg_Buf
32H		R_Cfg_Buf_Ptr		Segment base address of the Cfg_Buf
33H		R_Len_Read_Cfg_Data		Length of the input data in the Read_Cfg_Buf
34H		R_Read_Cfg_Buf_Ptr		Segment base address of the Read_Cfg_Buf
35H		R_Len_DXB_Link_Buf		Length of the DXB_Linktable
36H		R_DXB_Link_Buf_Ptr		Segment base address of the DXB_Link_Buf
37H		R_Len_DXB_Status_Buf		Length of the DXB_Status
38H		R_DXB_Status_Buf_Ptr		Segment base address of the DXB_Status_Buf
39H		R_Real_No_Add_Change		This parameter specifies whether the Station_Address may be changed again later.
3AH		R_Ident_Low		The user sets the parameters for the Ident_Number_Low value.
3BH		R_Ident_High		The user sets the parameters for the Ident_Number_High value.
3CH		R_GC_Command		The Control_Command of Global_Control last received
3DH		R_Len_Spec_Prm_Buf		If parameters are set for the Spec_Prm_Buffer_Mode (see Mode Register 0), this cell defines the length of the Prm_Buf.
3EH		R_DXBout_Buf_Ptr2		Segment base address of DXBout_Buf 2
3FH		R_DXBout_Buf_Ptr3		Segment base address of DXBout_Buf 3

Figure 4-4: Assignment of the Organizational Parameters

5.1 Mode Registers

In the VPC3+S parameter bits that access the controller directly or which the controller directly sets are combined in three Mode Registers (0, 1, 2 and 3).

5.1.1 Mode Register 0

Setting parameters for Mode Register 0 may take place in the Offline state only (for example, after power-on). The VPC3+S may not exit the Offline state until Mode Register 0, all Control and Organizational Parameters are loaded (START_VPC3 = 1 in Mode Register 1).

Address	Bit Position								Designation
	7	6	5	4	3	2	1	0	
06H (Intel)	Freeze_ Supported	Sync_ Supported	Early_Rdy	Int_Pol	CS_ Supported	WD_Base	Dis_Stop_ Control	Dis_Start_ Control	Mode Reg 0 7 .. 0 See below for coding

Address	Bit Position								Designation
	15	14	13	12	11	10	9	8	
07H (Intel)	Reserved	PrmCmd_ Supported	Spec_Clear_ Mode *)	Spec_Prm_ Buf_Mode **)	Set_Ext_Prm _Supported	User_Time_ Base	EOI_Time_ Base	DP_Mode	Mode Reg 0 15 .. 8 See below for coding

*) If Spec_Clear_Mode = 1 (Fail Safe Mode) the VPC3+S will accept Data_Exchange telegrams without any output data (data unit length = 0) in the state DATA-EXCH. The reaction to the outputs can be parameterized in the parameterization telegram.

**) When a large number of parameters have to be transmitted from the DP-Master to the DP-Slave, the Aux-Buffer 1/2 must have the same length as the Parameter-Buffer. Sometimes this could reach the limit of the available memory in the VPC3+S. When Spec_Prm_Buf_Mode = 1 the parameterization data are processed directly in this special buffer and the Aux-Buffers can be held compact.

	Mode Register 0, Low-Byte, Address 06H (Intel):
bit 7 rw-0	**Freeze_Supported**: Freeze_Mode support 0 = Freeze_Mode is not supported. 1 = Freeze_Mode is supported
bit 6 rw-0	**Sync_Supported:** Sync_Mode support 0 = Sync_Mode is not supported. 1 = Sync_Mode is supported.
bit 5 rw-0	**Early_Rdy**: Early Ready 0 = Normal Ready: Ready is generated when data is valid (write) or when data has been accepted (read). 1 = Ready is generated one clock pulse earlier
bit 4 rw-0	**INT_Pol:** Interrupt Polarity 0 = The interrupt output is low-active. 1 = The interrupt output is high-active.
bit 3 rw-0	**CS_Supported:** Enable Clock Synchronization 0 = Clock Synchronization is disabled (default) 1 = Clock Synchronization is enabled
bit 2 rw-0	**WD_Base:** Watchdog Time Base 0 = Watchdog time base is 10 ms (default state) 1 = Watchdog time base is 1 ms
bit 1 rw-0	**Dis_Stop_Control:** Disable Stopbit Control 0 = Stop bit monitoring is enabled. 1 = Stop bit monitoring is switched off Set_Prm telegram overwrites this memory cell in the DP_Mode. (Refer to the user specific data.)
bit 0 rw-0	**Dis_Start_Control:** Disable Startbit Control 0 = Monitoring the following start bit is enabled. 1 = Monitoring the following start bit is switched off Set_Prm telegram overwrites this memory cell in the DP_Mode. (Refer to the user specific data.)

Figure 5-1: Coding of Mode Register 0, Low-Byte

	Mode Register 0, High-Byte, Address 07H (Intel):
bit 15 rw-0	**Reserved**
bit 14 rw-0	**PrmCmd_Supported**: PrmCmd support for redundancy 0 = PrmCmd is not supported. 1 = PrmCmd is supported
bit 13 rw-0	**Spec_Clear_Mode:** Special Clear Mode (Fail Safe Mode) 0 = No special clear mode. 1 = Special clear mode. VPC3+S will accept data telegrams with data unit = 0
bit 12 rw-0	**Spec_Prm_Buf_Mode:** Special-Parameter-Buffer Mode 0 = No Special-Parameter-Buffer. 1 = Special-Parameter-Buffer mode. Parameterization data will be stored directly in the Special-Parameter-Buffer.
bit 11 rw-0	**Set_Ext_Prm_Supported:** Set_Ext_Prm telegram support 0 = SAP 53 is deactivated 1 = SAP 53 is activated
bit 10 rw-0	**User_Time_Base:** Timebase of the cyclical User_Time_Clock-Interrupt 0 = The User_Time_Clock-Interrupt occurs every 1 ms. 1 = The User_Time_Clock-Interrupt occurs every 10 ms.
bit 9 rw-0	**EOI_Time_Base:** End-of-Interrupt Timebase 0 = The interrupt inactive time is at least 1 µs long. 1 = The interrupt inactive time is at least 1 ms long
bit 8 rw-0	**DP_Mode:** DP_Mode enable 0 = DP_Mode is disabled. 1 = DP_Mode is enabled. VPC3+S sets up all DP_SAPs (default configuration!)

Figure 5-2: Coding of Mode Register 0, High-Byte

5.1.2 Mode Register 1

Some control bits must be changed during operation. These control bits are combined in Mode Register 1 and can be set independently of each other (Mode-Reg_1_S) or can be reset independently of each other (Mode-Reg_1_R). Separate addresses are used for setting and resetting. A logical ‘1’ must be written to the bit position to be set or reset.

For example, to set START_VPC3 write a '1' to address 08H, in order to reset this bit, write a '1' to address 09H.

Address	Bit Position								Designation
	7	6	5	4	3	2	1	0	
08H	Reserved	Reserved	Res_ User_WD	En_Change_ Cfg_Buffer	User_LEAVE-MASTER	Go_Offline	EOI	START_ VPC3	Mode-Reg_1_S 7..0
09H	Reserved	Reserved	Res_ User_WD	En_Change_ Cfg_Buffer	User_LEAVE-MASTER	Go_Offline	EOI	START_ VPC3	Mode-Reg_1_R 7..0 See below for coding

	Mode Register 1, Set, Address 08H:
bit 7 rw-0	**Reserved**
bit 6 rw-0	**Reserved**
bit 5 rw-0	**Res_User_WD:** Resetting the User_WD_Timer 1 = VPC3+S sets the User_WD_Timer to the parameterized value User_WD_Value. After this action, VPC3+S sets Res_User_WD to ’0'.
bit 4 rw-0	**En_Change_Cfg_Buffer:** Enabling buffer exchange (Config-Buffer for Read_Config-Buffer) 0 = With User_Cfg_Data_Okay_Cmd, the Config-Buffer may not be exchanged for the Read_Config-Buffer. 1 = With User_Cfg_Data_Okay_Cmd, the Config-Buffer must be exchanged for the Read_Config-Buffer.
bit 3 rw-0	**User_LEAVE-MASTER.** Request to the DP_SM to go to WAIT-PRM. 1 = The user causes the DP_SM to go to WAIT-PRM. After this action, VPC3+ sets User_LEAVE-MASTER to ’0’ again.
bit 2 rw-0	**Go_Offline:** Going into the Offline state 1 = After the current request ends, VPC3+S goes to the Offline state and sets Go_Offline to ’0’ again.
bit 1 rw-0	**EOI:** End-of-Interrupt 1 = VPC3+S disables the interrupt output and sets EOI to ’0‘ again.
bit 0 rw-0	**Start_VPC3:** Exiting the Offline state 1 = VPC3+S exits offline and goes to Passive_Idle In addition the Idle Timer and Watchdog Timer are started and ‘Go_Offline = 0’ is set

Figure 5-3: Coding of Mode Register 1

5.1.3 Mode Register 2

Setting parameters for Mode Register 2 may take place in the Offline State only (like Mode Register 0).

Address	Bit Position								Designation
	7	6	5	4	3	2	1	0	
	0	0	0	0	0	0	0	1	Reset Value
0CH	4kB_Mode	No_Check_ Prm_Reserved	SYNC_Pol	SYNC_Ena	DX_Int_Port	DX_Int_Mode	No_Check_ GC_Reserved	GC_Int_Mode	Mode Reg 2 7 .. 0

	Mode Register 2, Address 0CH:
bit 7 w-0	**4KB_Mode:** size of internal RAM 0 = 2K Byte RAM (default). 1 = 4K Byte RAM
bit 6 w-0	**No_Check_Prm_Reserved:** disables checking of the reserved bits in DPV1_Status_2/3 of Set_Prm telegram 0 = reserved bits of a Set_Prm telegram are checked (default). 1 = reserved bits of a Set_Prm telegram are not checked.
bit 5 w-0	**SYNC_Pol:** polarity of SYNC pulse (for Isochronous Mode only) 0 = negative polarity of SYNC pulse (default) 1 = positive polarity of SYNC pulse
bit 4 w-0	**SYNC_Ena:** enables generation of SYNC pulse (for Isochronous Mode only) 0 = SYNC pulse generation is disabled (default) 1 = SYNC pulse generation is enabled
bit 3 w-0	**DX_Int_Port:** Port mode for DX_Out interrupt (ignored if SYNC_Ena set) 0 = DX_Out interrupt is not assigned to port DATAEXCH (default). 1 = DX_Out Interrupt (synchronized to SYNCH telegram) is assigned to port DATAEXCH.
bit 2 w-0	**DX_Int_Mode:** Mode of DX_out interrupt 0 = DX_Out interrupt is only generated, if Len_Dout_Buf is unequal 0 (default). 1 = DX_Out interrupt is generated after every Data_Exchange telegram
bit 1 w-0	**No_Check_GC_Reserved:** Disables checking of the reserved bits in Global_Control telegram 0 = reserved bits of a Global_Control telegram are checked (default). 1 = reserved bits of a Global_Control telegram are not checked.
bit 0 w-1	**GC_Int_Mode:** Controls generation of New_GC_Command interrupt 0 = New_GC_Command interrupt is only generated, if a changed Global_Control telegram is received 1 = New_GC_Command interrupt is generated after every Global_Control telegram (default)

Figure 5-4: Coding of Mode Register 2

5.1.4 Mode Register 3

Setting parameters for Mode Register 3 may take place in the Offline State only (like Mode Register 0).

Address	Bit Position								Designation
	7	6	5	4	3	2	1	0	
12H	Reserved				PLL_ Supported	En_Chk_SSAP	DX_Int_Mode _2	GC_Int_Mode _Ext	Mode Reg 3 7 .. 0

	Mode Register 3, Address 12H:
bit 7 w-0	**Reserved**
bit 6 w-0	**Reserved**
bit 5 w-0	**Reserved**
bit 4 w-0	**Reserved**
bit 3 w-0	**PLL_Supported:** Enables IsoM-PLL 0 = PLL is disabled 1 = PLL is enabled; For use of PLL, SYNC_Ena must be set.
bit 2 w-0	**En_Chk_SSAP:** Evaluation of Source Address Extension 0 = VPC3+ accept any value of S_SAP 1 = VPC3+ only process the received telegram if the S_SAP match to the default values presented by the IEC 61158
bit 1 w-0	**DX_Int_Mode_2:** Mode of DX_out interrupt 0 = DX_Out interrupt is generated after each Data_Exch telegram 1 = DX_Out interrupt is only generated, if received data is not equal to current data in DX_Out buffer of user
bit 0 w-0	**GC_Int_Mode_Ext:** extend GC_Int_Mode, works only if GC_Int_Mode=0 0 = GC Interrupt is only generated, if changed GC telegram is received 1 = GC Interrupt is only generated, if GC telegram with changed Control_Command is received

Figure 5-5: Coding of Mode Register 3

5.2 Status Register

The Status Register shows the current VPC3+S status and can be read only.

Address	Bit Position								Designation
	7	6	5	4	3	2	1	0	
04H (Intel)	WD_State		DP_State		Reserved	Diag_Flag	Reserved	Offline/ Passive_Idle	Status-Reg 7..0
	1	0	1	0					See below for coding

Address	Bit Position								Designation
	15	14	13	12	11	10	9	8	
05H (Intel)	VPC3+ Release				Baud Rate				Status-Reg 15..8
	3	2	1	0	3	2	1	0	See below for coding

	Status Register,Low-Byte, Address 04H (Intel):
bit 7,6 r-00	**WD_State 1..0:** State of the Watchdog State Machine 00 = BAUD_SEARCH state 01 = BAUD_CONTROL state 10 = DP_CONTROL state 11 = Not possible
bit 5,4 r-00	**DP_State 1..0:** State of the DP State Machine 00 = WAIT-PRM state 01 = WAIT-CFG state 10 = DATA-EXCH state 11 = Not possible
bit 3 r-0	**Reserved**
bit 2 r-0	**Diag_Flag:** Status of the Diagnosis-Buffer 0 = The Diagnosis-Buffer had been fetched by the DP-Master. 1 = The Diagnosis-Buffer had not been fetched by the DP-Master yet.
bit 1 r-0	**Reserved**
bit 0 r-0	**Offline/Passive-Idle:** Offline-/Passive_Idle state 0 = VPC3+S is in Offline. 1 = VPC3+S is in Passive_Idle.

Figure 5-6: Status Register, Low-Byte

	Status Register, High-Byte, Address 05H (Intel):
bit 15-12 r-1110	**VPC3+-Release 3..0 :** Release number for VPC3+ 1110
bit 11-8 r-1111	**Baud Rate 3..0 :** The baud rate found by VPC3+S 0000 = 12,00 Mbit/s 0001 = 6,00 Mbit/s 0010 = 3,00 Mbit/s 0011 = 1,50 Mbit/s 0100 = 500,00 Kbit/s 0101 = 187,50 Kbit/s 0110 = 93,75 Kbit/s 0111 = 45,45 Kbit/s 1000 = 19,20 Kbit/s 1001 = 9,60 Kbit/s 1111 = after reset and during baud rate search Rest = not possible

Figure 5-7: Status Register, High-Byte

5.4.1 Automatic Baud Rate Identification

The VPC3+S starts searching for the transmission rate using the highest baud rate. If no SD1 telegram, SD2 telegram, or SD3 telegram was received completely and without errors during the monitoring time, the search continues using the next lower baud rate.

After identifying the correct baud rate, the VPC3+S switches to the BAUD_CONTROL state and observes the baud rate. The monitoring time can be parameterized (WD_BAUD_CONTROL_Val). The watchdog uses a clock of 100 Hz (10 ms). Each telegram to its own Station_Address received with no errors resets the Watchdog. If the timer expires, the VPC3+S switches to the BAUD_SEARCH state again.

5.4.2 Baud Rate Monitoring

The detected baud rate is permanently monitored in BAUD_CONTROL. The Watchdog is triggered by each error-free telegram to its own Station_Address. The monitoring time results from multiplying twice WD_BAUD_CONTROL_Val (user sets this parameter) by the time base (10 ms). If the timer expires, WD_SM again goes to BAUD_SEARCH. If the user uses the DP protocol (DP_Mode = 1, see Mode Register 0), the watchdog is used for the DP_CONTROL state, after a Set_Prm telegram was received with an enabled response time monitoring (WD_On = 1). The watchdog timer remains in the baud rate monitoring state when the master monitoring is disabled (WD_On = 0). The DP_SM is not reset when the timer expires in the state BAUD_CONTROL. That is, the DP-Slave remains in the DATA-EXCH state, for example.

5.4.3 Response Time Monitoring

The DP_CONTROL state serves as the response time monitoring of the DP-Master (Diag_Master_Add). The used monitoring time results from multiplying both watchdog factors and then multiplying this result with the time base (1 ms or 10 ms):

T_{WD} = WD_Base * WD_Fact_1 * WD_Fact_2
(See byte 7 of the Set_Prm telegram.)

The user can load the two watchdog factors (WD_Fact_1 and WD_Fact_2) and the time base that represents a measurement for the monitoring time via the Set_Prm telegram with any value between 1 and 255.

EXCEPTION:
The WD_Fact_1 = WD_Fact_2 = 1 setting is not allowed. The circuit does not check this setting.

A monitoring time between 2 ms and 650 s - independent of the baud rate - can be implemented with the allowed watchdog factors.

If the monitoring time expires, the VPC3+S goes to BAUD_CONTROL state again and generates the WD_DP_CONTROL_Timeout interrupt. In addition, the DP State Machine is reset, that is, it generates the reset states of the buffer management. This operation mode is recommended for the most applications.
If another DP-Master takes over the VPC3+S, the Watchdog State Machine either branches to BAUD_CONTROL (WD_On = 0) or to DP_CONTROL (WD_On = 1).

6.2 Description of the DP Services

6.2.1 Set_Slave_Add (SAP 55)

Sequence for the Set_Slave_Add service

The user can disable this service by setting 'R_SSA_Puf_Ptr = 00H'. The Station_Address must then be determined, for example, by reading a DIP-switch or an EEPROM and writing the address in the RAM cell R_TS_Adr.

There must be a non-volatile memory available (for example an external EEPROM) to support this service. It must be possible to store the Station_Address and the Real_No_Add_Change ('True' = FFH) parameter in this EEPROM. After each restart caused by a power failure, the user must read these values from the EEPROM again and write them to the R_TS_Adr und R_Real_No_Add_Change RAM registers.

If SAP55 is enabled and the Set_Slave_Add telegram is received correctly, the VPC3+S enters the pure data in the Aux-Buffer 1/2, exchanges the Aux-Buffer 1/2 for the Set_Slave_Add-Buffer, stores the entered data length in R_Len_SSA_Data, generates the New_SSA_Data interrupt and internally stores the New_Slave_Add as Station_Address and the No_Add_Chg as Real_No_Add_Chg. The user does not need to transfer this changed parameter to the VPC3+S again. After reading the buffer, the user generates the SSA_Buffer_Free_Cmd (read operation on address 14H). This makes the VPC3+S ready again to receive another Set_Slave_Add telegram (for example, from a different DP-Master).

The VPC3+S reacts automatically to errors.

Address	Bit Position								Designation
	7	6	5	4	3	2	1	0	
14H	0	0	0	0	0	0	0	0	SSA_Buf_ Free_Cmd

	SSA_Buf_Free_Cmd, Address 14H:
bit 7-0	**Don't care:** Read as '0'

Figure 6-3: Coding of SSA_Buffer_Free_Command

Structure of the Set_Slave_Add Telegram

The net data are stored as follows in the SSA buffer:

Byte	Bit Position								Designation
	7	6	5	4	3	2	1	0	
0									New_Slave_Address
1									Ident_Number_High
2									Ident_Number_Low
3									No_Add_Chg
4 : 243									Rem_Slave_Data additional application specific data

Figure 6-4: Structure of the Set_Slave_Add Telegram

6.2.2 Set _Prm (SAP 61)

Parameter Data Structure

The VPC3+S evaluates the first seven data bytes (without User_Prm_Data), or the first eight data bytes (with User_Prm_Data). The first seven bytes are specified according to the standard. The eighth byte is used for VPC3+S specific characteristics. The additional bytes are available to the application.

If a PROFIBUS DP extension shall be used, the bytes 7-9 are called DPV1_Status and must be coded as described in section 7, “PROFIBUS DP Extensions”. Generally it is recommended to start the User_Prm_Data first with byte 10.

Byte	Bit Position								Designation
	7	6	5	4	3	2	1	0	
0	Lock_ Req	Unlock_ Req	Sync_ Req	Freeze_ Req	WD_On	Reserved	Reserved	Reserved	Station Status
1									WD_Fact_1
2									WD_Fact_2
3									$minT_{SDR}$
4									Ident_Number_High
5									Ident_Number_Low
6									Group_Ident
7	DPV1_ Enable	Fail_Safe	Publisher_ Enable	0	0	WD_Base	Dis_Stop_ Control	Dis_Start_ Control	Spec_User_Prm_Byte /DPV1_Status_1
8									DPV1_Status_2
9									DPV1_Status_3
10 : 243									User_Prm_Data

Figure 6-5: Format of the Set_Prm Telegram

	Spec_User_Prm_Byte / DPV1_Status_1:
bit 7	**DPV1_Enable:** 0 = DP-V1 extensions disabled (default) 1 = DP-V1 extensions enabled
bit 6	**Fail_Safe:** 0 = Fail Safe mode disabled (default) 1 = Fail Safe mode enabled
bit 5	**Publisher_Enable:** 0 = Publisher function disabled (default) 1 = Publisher function enabled
bit 4-3	**Reserved:** To be parameterized with '0'
bit 2	**WD_Base:** Watchdog Time Base 0 = Watchdog time base is 10 ms (default) 1 = Watchdog time base is 1 ms
bit 1	**Dis_Stop_Control:** Disable Stop bit Control 0 = Stop bit monitoring in the receiver is enabled (default) 1 = Stop bit monitoring in the receiver is disabled
bit 0	**Dis_Start_Control:** Disable Start bit Control 0 = Start bit monitoring in the receiver is enabled (default) 1 = Start bit monitoring in the receiver is disabled

Figure 6-6: Spec_User_Prm_Byte / DPV1_Status_1

It is recommended <u>not</u> to use the DPV1_Status bytes (bytes 7-9) for user parameter data.

Parameter Data Processing Sequence

In the case of a positive validation of more than seven data bytes, the VPC3+S carries out the following reaction:

The VPC3+S exchanges Aux-Buffer 1/2 (all data bytes are entered here) for the Parameter-Buffer, stores the input data length in R_Len_Prm_Data and triggers the New_Prm_Data interrupt. The user must then check the User_Prm_Data and either reply with User_Prm_Data_Okay_Cmd or with User_Prm_Data_Not_Okay_Cmd. The entire telegram is entered in this buffer. The user parameter data are stored beginning with data byte 8, or with byte 10 if DPV1_Status bytes used.

The user response (User_Prm_Data_Okay_Cmd or User_Prm_Data_Not_Okay_Cmd) clears the New_Prm_Data interrupt. The user cannot acknowledge the New_Prm_Data interrupt in the IAR register.

With the User_Prm_Data_Not_Okay_Cmd message, relevant diagnosis bits are set and the DP_SM branches to WAIT-PRM.

The User_Prm_Data_Okay and User_Prm_Data_Not_Okay acknowledgements are read accesses to defined registers with the relevant signals:

- User_Prm_Finished: No additional parameter telegram is present.
- Prm_Conflict: An additional parameter telegram is present, processing again
- Not_Allowed: Access not permitted in the current bus state

Address	Bit Position								Designation
	7	6	5	4	3	2	1	0	
0EH	0	0	0	0	0	0	⇓	⇓	User_Prm_ Data_Okay
							0	0	User_Prm_Finished
							0	1	Prm_Conflict
							1	1	Not_Allowed

Address	Bit Position								Designation
	7	6	5	4	3	2	1	0	
0FH	0	0	0	0	0	0	⇓	⇓	User_Prm_ Data_Not_Okay
							0	0	User_Prm_Finished
							0	1	Prm_Conflict
							1	1	Not_Allowed

Figure 6-7: Coding of User_Prm_(Not)_Okay_Cmd

If another Set_Prm telegram is supposed to be received in the meantime, the signal Prm_Conflict is returned for the positive or negative acknowledgement of the first Set_Prm telegram. Then the user must repeat the validation because the VPC3+S has made a new Parameter-Buffer available.

6.2.3 Chk_Cfg (SAP 62)

The user checks the correctness of the configuration data. After receiving an error-free Chk_Cfg telegram, the VPC3+S exchanges the Aux-Buffer 1/2 (all data bytes are entered here) for the Config-Buffer, stores the input data length in R_Len_Cfg_Data and generates the New_Cfg_Data interrupt.

Then the user has to check the User_Config_Data and either respond with User_Cfg_Data_Okay_Cmd or with User_Cfg_Data_Not_Okay_Cmd. The pure data is entered in the buffer in the format of the standard.

The user response (User_Cfg_Data_Okay_Cmd or the User_Cfg_Data_Not_Okay_Cmd response) clears the New_Cfg_Data interrupt. The user cannot acknowledge the New_Cfg_Data in the IAR register.

If an incorrect configuration is reported, several diagnosis bits are changed and the VPC3+S branches to state WAIT-PRM.

For a correct configuration, the transition to DATA-EXCH takes place immediately, if trigger counters for the parameter telegrams and configuration telegrams are at 0. When entering into DATA-EXCH, the VPC3+S also generates the Go/Leave_DATA-EXCH Interrupt.

If the received configuration data from the Config-Buffer is supposed to result in a change to the Read_Config-Buffer (contains the data for the Get_Cfg telegram), the user have to make the new Read_Config data available in the Read_Config-Buffer before the User_Cfg_Data_Okay_Cmd acknowledgement, that is the user has to copy the new configuration data into the Read_Config-Buffer.

During acknowledgement, the user receives information about whether there is a conflict or not. If another Chk_Cfg telegram was supposed to be

received in the meantime, the user receives the Cfg_Conflict signal during the positive or negative acknowledgement of the first Chk_Cfg telegram. Then the user must repeat the validation, because the VPC3+S have made a new Config-Buffer available.

The User_Cfg_Data_Okay_Cmd and User_Cfg_Data_Not_Okay_Cmd acknowledgements are read accesses to defined memory cells with the relevant Not_Allowed, User_Cfg_Finished, or Cfg_Conflict signals.

If the New_Prm_Data and New_Cfg_Data are supposed to be present simultaneously during start-up, the user must maintain the Set_Prm and then the Chk_Cfg acknowledgement sequence.

Address	Bit Position								Designation
	7	6	5	4	3	2	1	0	
10H	0	0	0	0	0	0	⇓	⇓	User_Cfg_ Data_Okay
							0	0	User_Cfg_Finished
							0	1	Cfg_Conflict
							1	1	Not_Allowed

Address	Bit Position								Designation
	7	6	5	4	3	2	1	0	
11H	0	0	0	0	0	0	⇓	⇓	User_Cfg_ Data_Not_Okay
							0	0	User_Cfg_Finished
							0	1	Cfg_Conflict
							1	1	Not_Allowed

Figure 6-8: Coding of User_Cfg_(Not)_Okay_Cmd

6.2.4 Slave_Diag (SAP 60)

Diagnosis Processing Sequence

Two buffers are available for diagnosis. These two buffers can have different lengths. One Diagnosis-Buffer, which is sent on a diagnosis request, is always assigned to the VPC3+S. The user can pre-process new diagnosis data in the other buffer parallel. If the new diagnosis data are to be sent, the user issues the New_Diag_Cmd to make the request to exchange the Diagnosis-Buffers. The user receives confirmation of the buffer exchange with the Diag_Buffer_Changed interrupt.

When the buffers are exchanged, the internal Diag_Flag is also set. For an activated Diag_Flag, the VPC3+S responds during the next

Data_Exchange with high-priority response data. That signals the DP-Master that new diagnosis data are present at the DP-Slave. The DP-Master then fetches the new diagnosis data with a Slave_Diag telegram. Then the Diag_Flag is cleared again. However, if the user signals 'Diag.Stat_Diag = 1' (that is static diagnosis, see the structure of the Diagnosis-Buffer), the Diag_Flag still remains activated after the relevant DP-Master has fetched the diagnosis. The user can poll the Diag_Flag in the Status Register to find out whether the DP-Master has already fetched the diagnosis data before the old data is exchanged for the new data.

According to IEC 61158, Static Diagnosis should only be used during start-up.

Status coding for the diagnosis buffers is stored in the Diag_Buffer_SM control parameter. The user can read this cell with the possible codings for both buffers: User, VPC3+, or VPC3+_Send_Mode.

Address	Bit Position								Designation
	7	6	5	4	3	2	1	0	
0CH	0	0	0	0	Diag_Buf2		Diag_Buf1		Diag_Buffer_SM

	Diag_Buffer_SM, Address 0CH:
bit 7-4	**Don't care:** Read as '0'
bit 3-2	**Diag_Buf2:** Assignment of Diagnosis Buffer 2 00 = Nil 01 = User 10 = VPC3+ 11 = VPC3_Send_Mode
bit 1-0	**Diag_Buf1:** Assignment of Diagnosis Buffer 1 00 = Nil 01 = User 10 = VPC3+ 11 = VPC3_Send_Mode

Figure 6-9: Diagnosis Buffer Assignment

The New_Diag_Cmd is also a read access to a defined control parameter indicating which Diagnosis-Buffer belongs to the user after the exchange or whether both buffers are currently assigned to the VPC3+S (No_Buffer, Diag_Buf1, Diag_Buf2).

Address	Bit Position								Designation
	7	6	5	4	3	2	1	0	
0DH	0	0	0	0	0	0	⇓	⇓	New_Diag_ Buffer_Cmd
							0	0	No_Buffer
							0	1	Diag_Buf1
							1	0	Diag_Buf2

Figure 6-10: Coding of New_Diag_Cmd

Byte	Bit Position								Designation
	7	6	5	4	3	2	1	0	
0						Ext_Diag_ Overflow	Stat_Diag	Ext_Diag	
1									
2									
3									
4									
5									
6 : n	user input								Ext_Diag_Data (n = max. 243)

Figure 6-11: Format of the Diagnosis-Buffer

The Ext_Diag_Data must be entered into the buffers after the VPC3+S internal diagnosis data. Three different formats are possible here: device-related, ID-related and port-related. If PROFIBUS DP extensions shall be used, the device-related diagnosis is substituted by alarm and status messages. In addition to the Ext_Diag_Data, the buffer length also includes the VPC3+S diagnosis bytes (R_Len_Diag_Buf 1, R_Len_Diag_Buf 2).

6.2.5 Write_Read_Data / Data_Exchange (Default_SAP)

Writing Outputs

The VPC3+S writes the received output data in the 'D' buffer. After an error-free receipt, the VPC3+S shifts the newly filled buffer from ‘D’ to ‘N'. In addition, the DX_Out interrupt is generated. The user now fetches the current output data from ‘N’. The buffer changes from ‘N’ to ‘U’ with the Next_Dout_Buffer_Cmd, so that the current data can be transmitted to the application by a RD_Output request from a DP-Master.

If the user's evaluation cycle time is shorter than the bus cycle time, the user does not find any new buffers with the next Next_Dout_Buffer_Cmd in 'N'. Therefore, the buffer exchange is omitted. At a 12 Mbit/s baud rate, it is more likely, however, that the user's evaluation cycle time is larger than the bus cycle time. This makes new output data available in 'N' several times before the user fetches the next buffer. It is guaranteed, however, that the user receives the data last received.

For power-on, LEAVE-MASTER and the Global_Control telegram with 'Clear_Data = 1', the VPC3+S deletes the 'D' buffer and then shifts it to 'N'. This also takes place during power-up (entering the WAIT-PRM state). If the user fetches this buffer, he receives U_Buffer_Cleared during the Next_Dout_Buffer_Cmd. If the user is supposed to enlarge the output data buffer after the Chk_Cfg telegram, the user must delete this deviation in the 'N' buffer himself (possible only during the start-up phase in the WAIT-CFG state).

If 'Diag.Sync_Mode = 1', the 'D' buffer is filled but not exchanged with the Data_Exchange telegram. It is exchanged at the next Sync or Unsync command sent by Global_Control telegram.

Address	Bit Position								Designation
	7	6	5	4	3	2	1	0	
0AH	F		U		N		D		Dout_Buffer_SM

	Dout_Buffer_SM, Address 0AH:
bit 7-6	**F:** Assignment of the F-Buffer 00 = Nil 01 = Dout_Buf_Ptr1 10 = Dout_Buf_Ptr2 11 = Dout_Buf_Ptr3
bit 5-4	**U:** Assignment of the U-Buffer 00 = Nil 01 = Dout_Buf_Ptr1 10 = Dout_Buf_Ptr2 11 = Dout_Buf_Ptr3
bit 3-2	**N:** Assignment of the N-Buffer 00 = Nil 01 = Dout_Buf_Ptr1 10 = Dout_Buf_Ptr2 11 = Dout_Buf_Ptr3
bit 1-0	**D:** Assignment of the D-Buffer 00 = Nil 01 = Dout_Buf_Ptr1 10 = Dout_Buf_Ptr2 11 = Dout_Buf_Ptr3

Figure 6-12: Dout-Buffer Management

When reading the Next_Dout_Buffer_Cmd the user gets the information which buffer (‘U’ buffer) belongs to the user after the change, or whether a change has taken place at all.

Address	Bit Position								Designation
	7	6	5	4	3	2	1	0	
0BH	0	0	0	0	U_Buffer_ Cleared	State_U_ Buffer	Ind_U_ Buffer		Next_Dout_ Buf_Cmd See coding below

	Next_Dout_Buf_Cmd, Address 0BH:
bit 7-4	**Don’t care:** Read as ‘0’
bit 3	**U_Buffer_Cleared:** User-Buffer-Cleared Flag 0 = U buffer contains data 1 = U buffer is cleared
bit 2	**State_U_Buffer:** State of the User-Buffer 0 = no new U buffer 1 = new U buffer
bit 1-0	**Ind_U_Buffer:** Indicated User-Buffer 01 = Dout_Buf_Ptr1 10 = Dout_Buf_Ptr2 11 = Dout_Buf_Ptr3

Figure 6-13: Coding of Next_Dout_Buf_Cmd

The user must clear the ‘U’ buffer during initialization so that defined (cleared) data can be sent for a RD_Output telegram before the first data cycle.

Reading Inputs

The VPC3+S sends the input data from the ‘D’ buffer. Prior to sending, the VPC3+S fetches the Din-Buffer from ‘N’ to ‘D'. If no new buffer is present in ‘N', there is no change.

The user makes the new data available in ‘U’. With the New_Din_Buffer_Cmd, the buffer changes from ‘U’ to ‘N’. If the user’s preparation cycle time is shorter than the bus cycle time, not all new input data are sent, but just the most current. At a 12 Mbit/s baud rate, it is more likely, however, that the user’s preparation cycle time is larger than the bus cycle time. Then the VPC3+S sends the same data several times in succession.

During start-up, the VPC3+S does not go to DATA-EXCH before all parameter telegrams and configuration telegrams have been acknowledged.

If ‘Diag.Freeze_Mode = 1’, there is no buffer change prior to sending.

The user can read the status of the state machine cell with the following codings for the four states: Nil, Dout_Buf_Ptr1, Dout_Buf_Ptr2 and Dout_Buf_Ptr3. The pointer for the current data is in the 'N' state.

Address	Bit Position								Designation
	7	6	5	4	3	2	1	0	
08H	F		U		N		D		Din_Buffer_SM

	Din_Buffer_SM, Address 08H:
bit 7-6	**F:** Assignment of the F-Buffer 00 = Nil 01 = Din_Buf_Ptr1 10 = Din_Buf_Ptr2 11 = Din_Buf_Ptr3
bit 5-4	**U:** Assignment of the U-Buffer 00 = Nil 01 = Din_Buf_Ptr1 10 = Din_Buf_Ptr2 11 = Din_Buf_Ptr3
bit 3-2	**N:** Assignment of the N-Buffer 00 = Nil 01 = Din_Buf_Ptr1 10 = Din_Buf_Ptr2 11 = Din_Buf_Ptr3
bit 1-0	**D:** Assignment of the D-Buffer 00 = Nil 01 = Din_Buf_Ptr1 10 = Din_Buf_Ptr2 11 = Din_Buf_Ptr3

Figure 6-14: Din-Buffer Management

Address	Bit Position								Designation
	7	6	5	4	3	2	1	0	
09H	0	0	0	0	0	0	⇓	⇓	New_Din_Buf_Cmd
							0	1	Din_Buf_Ptr1
							1	0	Din_Buf_Ptr2
							1	1	Din_Buf_Ptr3

Figure 6-15: Coding of New_Din_Buf_Cmd

User_Watchdog_Timer

After start-up (DATA-EXCH state), it is possible that the VPC3+S continually answers Data_Exchange telegrams without the user fetching the received Dout-Buffers or making new Din-Buffers available. If the user processor ‘hangs up' the DP-Master would not receive this information. Therefore, a User_Watchdog_Timer is implemented in the VPC3+S.

This User_WD_Timer is an internal 16-bit RAM cell that is started from a user parameterized value R_User_WD_Value and is decremented by the VPC3+S with each received Data_Exchange telegram. If the timer reaches the value 0000H, the VPC3+S goes to the WAIT-PRM state and the DP_SM carries out a LEAVE-MASTER. The user must cyclically set this timer to its start value. Therefore, ‘Res_User_WD = 1’ must be set in Mode Register 1. Upon receipt of the next Data_Exchange telegram, the VPC3+S again loads the User_WD_Timer to the parameterized value R_User_WD_Value and sets ‘Res_User_WD = 0’ (Mode Register 1). During power-up, the user must also set ‘Res_User_WD = 1’, so that the User_WD_Timer is set to its parameterized value.

6.2.6 Global_Control (SAP 58)

The VPC3+S processes the Global_Control telegrams like already described.

The first byte of a valid Global_Control is stored in the R_GC_Command RAM cell. The second telegram byte (Group_Select) is processed internally.

The interrupt behavior regarding to the reception of a Global_Control telegram can be configured via bit 8 of Mode Register 2. The VPC3+S either generates the New_GC_Control interrupt after each receipt of a Global_Control telegram (default) or just in case if the Global_Control differs from the previous one.

The R_GC_Command RAM cell is not initialized by the VPC3+S. Therefore the cell has to be preset with 00H during power-up. The user can read and evaluate this cell.

In order to use Sync and Freeze, these functions must be enabled in the Mode Register 0.

Address	Bit Position								Designation
	7	6	5	4	3	2	1	0	
3CH	Reserved	Reserved	Sync	Unsync	Freeze	Unfreeze	Clear_Data	Reserved	R_GC_ Command See below for coding

	R_GC_Command, Address 3CH:
bit 7-6	**Reserved**
bit 5	**Sync:** The output data transferred with a Data_Exchange telegram is changed from ‘D’ to ‘N’. The following transferred output data is kept in ‘D’ until the next Sync command is given.
bit 4	**Unsync:** The Unsync command cancels the Sync command.
bit 3	**Freeze:** The input data is fetched from ‘N’ to ‘D’ and „frozen“. New input data is not fetched again until the DP-Master sends the next Freeze command.
bit 2	**Unfreeze:** The Unfreeze command cancels the Freeze command.
bit 1	**Clear_Data:** With this command, the output data is deleted in ‘D’ and is changed to ‘N’.
bit 0	**Reserved**

Figure 6-16: Format of the Global_Control Telegram

6.2.7 RD_Input (SAP 56)

The VPC3+S fetches the input data like it does for the Data_Exchange telegram. Prior to sending, ‘N’ is shifted to ‘D', if new input data are available in ‘N'. For ‘Diag.Freeze_Mode = 1', there is no buffer change.

6.2.8 RD_Output (SAP 57)

The VPC3+S fetches the output data from the Dout_Buffer in ‘U’. The user must preset the output data with ‘0’ during start-up so that no invalid data can be sent here. If there is a buffer change from ‘N’ to ‘U’ (through the

Next_Dout_Buffer_Cmd) between the first call-up and the repetition, the new output data is sent during the repetition.

6.2.9 Get_Cfg (SAP 59)

The user makes the configuration data available in the Read_Config-Buffer. For a change in the configuration after the Chk_Cfg telegram, the user writes the changed data in the Config-Buffer, sets 'En_Change_Cfg_buffer = 1' (see Mode Register 1) and the VPC3+S then exchanges the Config-Buffer for the Read_Config-Buffer. If there is a change in the configuration data during operation (for example, for a modular DP systems), the user must return with Go_Offline command (see Mode Register 1) to WAIT-PRM.

7.1 Set_(Ext_)Prm (SAP 53 / SAP 61)

The PROFIBUS DP extensions require three bytes to implement the new parameterization function. The bits of the Spec_User_Prm_Byte are included.

Byte	Bit Position								Designation
	7	6	5	4	3	2	1	0	
0 : 6									
7	DPV1_ Enable	Fail_Safe	Publisher_ Enable	Reserved	Reserved	WD_Base	Dis_Stop_ Control	Dis_Start_ Control	DPV1_Status_1
8	Enable_ Pull_Plug_Alarm	Enable_ Process_Alarm	Enable_ Diagnostic_Alarm	Enable_ Manufacturer_ Specific_Alarm	Enable_ Status_Alarm	Enable_ Update_Alarm	0	Chk_Cfg_Mode	DPV1_Status_2
9	PrmCmd	0	0	IsoM_Req	Prm_ Structure	Alarm_Mode			DPV1_Status_3
10 : 243									User_Prm_Data

Figure 7-1: Set_Prm with DPV1_Status bytes

If the extensions are used, the bit Spec_Clear_Mode in Mode Register 0 serves as Fail_Safe_required. Therefore it is used for a comparison with the bit Fail_Safe in parameter telegram. Whether the DP-Master supports the Fail_Safe mode or not is indicated by the telegram bit. If the DP-Slave requires Fail_Safe but the DP-Master doesn't the Prm_Fault bit is set.

If the VPC3+S should be used for DXB, IsoM or redundancy mode, the parameterization data must be packed in a Structured_Prm_Data block to distinguish between the User_Prm_Data. The bit Prm_Structure indicates this.

If redundancy should be supported, the PrmCmd_Supported bit in Mode Register 0 must be set.

Byte	Bit Position								Designation
	7	6	5	4	3	2	1	0	
0									Structured_Length
1									Structure_Typ
2									Slot_Number
3									Reserved
4 : 243									User_Prm_Data

Figure 7-2 : Format of the Structured_Prm_Data block

Additional to the Set_Prm telegram (SAP 61) a Set_Ext_Prm (SAP 53) telegram can be used for parameterization. This service is only available in state WAIT-CFG after the reception of a Set_Prm telegram and before the reception of a Chk_Cfg telegram. The new Set_Ext_Prm telegram simply consists of Structured_Prm_Data blocks.

The new service uses the same buffer handling as described by Set_Prm. By means of the New_Ext_Prm_Data interrupt the user can recognize which kind of telegram is entered in the Parameter-Buffer. Additional the SAP 53 must be activated by Set_Ext_Prm_Supported bit in Mode Register 0.

The Aux-Buffer for the Set_Ext_Prm is the same as the one for Set_Prm and has to be different from the Chk_Cfg Aux-Buffer. Furthermore the Spec_Prm_Buf_Mode in Mode Register 0 must not be used together with SAP 53.

7.2 PROFIBUS DP-V1

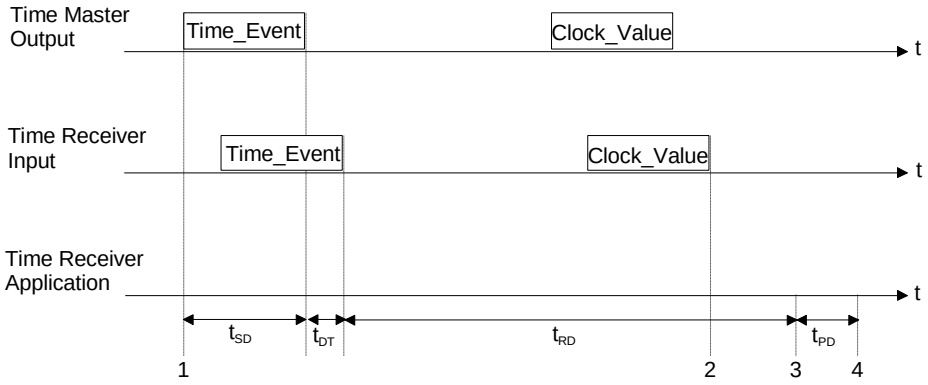
7.2.1 Acyclic Communication Relationships

The VPC3+S supports acyclic communication as described in the DP-V1 specification. Therefore a memory area is required which contains all SAPs needed for the communication. The user must do the initialization of this area (SAP-List) in Offline state. Each entry in the SAP-List consists of 7 bytes. The pointer at address 17H contains the segment base address of the first element of the SAP-List. The last element in the list is always indicated with FFH. If the SAP-List shall not be used, the first entry must be FFH, so the pointer at address 17H must point to a segment base address location that contains FFH.

The new communication features are enabled with DPV1_Enable in the Set_Prm telegram.

7.3.3 CS (Clock Synchronization)

The Clock Synchronization mechanism synchronizes the time between devices on a PROFIBUS segment. A time master is a DP-Master. The scheme used is a “backwards time based correction”. The knowledge of when a special timer event message was broadcasted is subsequently used to calculate appropriate clock adjustments.
The synchronized time can be used for time stamp mechanism.



1: Time Event
2: Clock_Sync Interrupt
3: read access Receive_Delay_Time
4: update system timer

Figure 7-26: clock synchronization mechanism

The clock synchronization sequence consists of two messages broadcasted by the time master. When the first message, called Time_Event, is received the VPC3+S starts the receive delay timer (t_{RD}). The time master then sends a second message, called Clock_Value, which contains the actual time when the Time_Event was sent plus the send delay time (t_{SD}). By receiption of the second message the Clock_Sync interrupt will be generated. To achieve the most accuracy the receive delay timer is running until the user reads the Clock_Sync-Buffer.
The VPC3+S only synchronizes the received telegrams, the system time management is done by the user. The user has also to account for the time after the receive delay timer has been read till the update of the system time (t_{PD}: process delay time).

The time for transmission delay (t_{DT}: CS_Delay_Time) and the Clock_Sync_Interval are communicated to the VPC3+S by a Structured_Prm_Data block. The CS_Delay_Time is used by the user to calculate the system time: t_S = Clock_Value_Time_Event + t_{DT} + t_{RD} + t_{PD}

Byte	Bit Position								Designation
	7	6	5	4	3	2	1	0	
0									Structured_Length
1	0	0	0	0	1	0	0	0	Structure_Type
2	0	0	0	0	0	0	0	0	Slot_Number
3	0	0	0	0	0	0	0	0	Reserved
4 : 5									Clock_Sync_Interval Time Base 10 ms
6 : 13	Seconds (2^{31}..0) Fraction Part of Seconds (2^{31}..0) Base is 1/(2^{32}) Seconds								CS Delay Time can be omitted

Figure 7-27: Format of Structured_Prm_Data with Time AR

Byte	Bit Position								Designation
	7	6	5	4	3	2	1	0	
0 : 7	Seconds (2^{31}..0) since 1.1.1900 0:00,00 or since 7.2.2036 6:28:16 if value < 0x9dff4400 Fraction Part of Seconds (2^{31}..0) Base is 1/(2^{32}) Seconds								Clock_Value_ Time_Event
8 : 15	Seconds (2^{31}..0) since 1.1.1900 0:00,00 or since 7.2.2036 6:28:16 if value < 0x9dff4400 Fraction Part of Seconds (2^{31}..0) Base is 1/(2^{32}) Seconds								Clock_Value_ previous_TE
16	C	CV					reserviert		Clock_Value_Status1
17	ANH	SWT	reserviert	CR		reserviert		SYF	Clock_Value_Status2

Figure 7-28: Format of Clock_Value

Processing Sequence

The Clock_Sync_Interval is a time for monitoring and has to be written into the Clock_Sync-Buffer by the user. The Time Receiver state machine in the VPC3+S is started after this write access. The value for Clock_Sync_Interval is locked until the next LEAVE-MASTER or a new parameterization occurs. In addition it can be unlocked if the user set the Stop_Clock_Sync in Command byte.

Following to a clock synchronization sequence the Clock_Sync interrupt will be asserted. Further information is contained in the Status byte. If an overflow of the Receive_Delay_Timer occurs the Status byte will be cleared. The VPC3+S cannot write new data to the Clock_Sync-Buffer until the user has acknowledged the Clock_Sync interrupt. Hence to ensure no new data overwrites the buffer, the user should read out the buffer before acknowledging the interrupt.

The base address of the Clock_Sync-Buffer depends on the memory mode:
2K Byte mode: 7E0H
4K Byte mode: FE0H

<table>
<tr><th rowspan="2">Byte</th><th colspan="8">Bit Position</th><th rowspan="2">Designation</th></tr>
<tr><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td>0</td><td colspan="6">reserved</td><td>Clock_Sync_ Violation</td><td>Set_Time</td><td>Status</td></tr>
<tr><td>1</td><td colspan="5">reserved</td><td>Clock_Value_ Check_Ena</td><td>Ignore_Cyclic_ State_Machine</td><td>Stop_ Clock_Sync</td><td>Command</td></tr>
<tr><td>2</td><td>C</td><td colspan="4">CV</td><td colspan="2">reserved</td><td>Clock_Value_Status1</td></tr>
<tr><td>3</td><td>ANH</td><td>SWT</td><td>reserved</td><td colspan="2">CR</td><td colspan="2">reserved</td><td>SYF</td><td>Clock_Value_Status2</td></tr>
<tr><td rowspan="2">4
:
11</td><td colspan="8">Seconds (2^{32}-1 .. 0) since 1.1.1900 0:00,00
or since 7.2.2036 6:28:16 if value < 9DFF4400H</td><td rowspan="2">Clock_Value_ Time_Event</td></tr>
<tr><td colspan="8">Fraction Part of Seconds (2^{32}-1 .. 0)
Base is 1/(2^{32}) Seconds</td></tr>
<tr><td>12
:
15</td><td colspan="8">(2^{32}-1 .. 0)
Time Base 1 µs</td><td>Receive_Delay_Time</td></tr>
<tr><td rowspan="2">16
:
23</td><td colspan="8">Seconds (2^{32}-1 .. 0) since 1.1.1900 0:00,00
or since 7.2.2036 6:28:16 if value < 9DFF4400H</td><td rowspan="2">Clock_Value_ previous_TE</td></tr>
<tr><td colspan="8">Fraction Part of Seconds (2^{32}-1 .. 0)
Base is 1/(2^{32}) Seconds</td></tr>
<tr><td>24
:
25</td><td colspan="8">(2^{16}-1 .. 0)
Time Base 10 ms</td><td>Clock_Sync_Interval</td></tr>
</table>

	Clock_Sync-Buffer
Status bit 7-2 r-000000	**Reserved**
Status bit 1 r-0	**Clock_Sync_Violation**: Wrong telegram or Time period of 2*T_{CSI} expired after reception of Time_Event.
Status bit 0 r-0	**Set_Time**: The VPC3+D has received a valid 'Clock_Value telegram' and made the data available in the Clock_Sync-Buffer.
Command bit 7-3 r-00000	**Reserved**
Command bit 2 rw-0	**Clock_Value_Check_Ena**: 0 = don't evaluate Clock_Value_previous_TE 1 = check Clock_Value_previous_TE with local variable Time_Last_Rcvd
Command bit 1 rw-0	**Ignore_Cyclic_State_Machine**: 0 = Clock Synchronization stops after the receiption of a new Set_Prm or a LEAVE-MASTER 1 = Clock Synchronization continues until the user set Stop_Clock_Sync
Command bit 0 w-0	**Stop_Clock_Sync**: Stop the Clock Synchronization, in order to write a new T_{CSI} without a previous Set_Prm or LEAVE-MASTER. The Bit is cleared by the Time_Receiver State Machine.
Clock_Value_ Status1 bit 7 r-0	**C: Sign of CV** 0 = add correction value to Time 1 = substract correction value to Time
Clock_Value_ Status1 bit 6-2 r-00000	**CV: Correction Value** 0 = 0 min 1..31 = 30..930 min
Clock_Value_ Status1 bit 1-0 r-00	**Reserved**

	Clock_Sync-Buffer
Clock_Value_ Status2 bit 7 r-0	**ANH: Announcment Hour** 0 = no change planned within the next hour 1 = a change of SWT will occur within the next hour
Clock_Value_ Status2 bit 6 r-0	**SWT: Summertime** 0 = Winter Time 1 = Summer Time
Clock_Value_ Status2 bit 5 r-0	**Reserved**
Clock_Value_ Status2 bit 4-3 r-00	**CR: Accuracy** 0 = 1 ms 1 = 10 ms 2 = 100 ms 3 = 1 s
Clock_Value_ Status2 bit 2-1 r-00	**Reserved**
Clock_Value_ Status2 bit 0 r-0	**SYF: Synchronisation Active**: 0 = Clock_Value_Time_Event is synchronized 1 = Clock_Value_Time_Event is not synchronized
r-0	**Clock_Value_Time_Event**: Same format as defined in IEC 61158-6 is used. Value is stored with the most significant byte at the lowest address. No address swapping is done for Intel format.
r-0	**Receive_Delay_Time**: Value is stored with the most significant byte in address 12. No address swapping is done for Intel format.
r-0	**Clock_Value_previous_TE**: Same format as defined in IEC 61158-6 is used. Value is stored with the most significant byte at the lowest address. No address swapping is done for Intel format.
rw-0	**Clock_Sync_Interval**: Value is stored with the most significant byte in address 24. No address swapping is done for Intel format.

Figure 7-29: Format of the Clock_Sync-Buffer

VPC3+S	Firmware
	set CS_Supported
	⋮
reception of Set_(Ext_)Prm	
set New_(Ext_)Prm_Data interrupt	
	acknowledge interrupt
	write Clock_Sync_Interval to CS-Buffer
	⋮
reception of Time_Event	
start Receive_Delay_Timer	
reception of Clock_Value	
set Clock_Sync interrupt	
	read CS_Status
	IF (Set_Time='1') THEN
stop Receive_Delay Timer	read CS_Buffer
	update system time
	END IF
	acknowledge interrupt

Figure 7-30: communication scheme

8.1 Universal Processor Bus Interface

8.1.1 Overview

The VPC3+S can be interfaced by using either a parallel 8-bit data interface or an SPI or I2C interface.

In parallel mode the VPC3+S provides an 8-bit data interface with an 11-bit address bus. The VPC3+S supports all 8-bit processors and micro-controllers based on the 80C51/52 (80C32) from Intel, the Motorola HC11 family, as well as 8- /16-bit processors or microcontrollers from the Siemens 80C166 family, X86 from Intel and the HC16 and HC916 family from Motorola. Because the data formats from Intel and Motorola are different, VPC3+S automatically carries out ‘byte swapping’ for accesses to the following 16-bit registers (Interrupt Register, Status Register and Mode Register 0) and the 16-bit RAM cell (R_User_WD_Value). This makes it possible for a Motorola processor to read the 16-bit value correctly. Reading or writing takes place, as usual, through two accesses (8-bit data bus).

Four SPI modes are supported which differ in clock polarity and clock phase. In these interface modes the VPC3+S acts like a memory device with serial (SPI) interface connected to the CPU. The chip needs to be selected by pulling the Slave-Select pin (SPI_XSS) low before receiving clock pulses via SPI_SCK pin from the CPU. Depending on the OP-code received the VPC3+S carries out a read or write operation starting at the specified address inside the internal memory. Serial data is shifted in via SPI_MOSI pin and shifted out via SPI_MISO pin.

In I2C mode the VPC3+S can be connected to an I2C network by using the pins I2C_SCK and I2C_SDA. In this mode the VPC3+S acts like a memory device with serial (I2C) interface connected to the CPU. The chip supports slave mode only and the desired slave address can be selected by using the pins I2C_A[6:0]. Upon reception of the correct slave address and depending on the status of the R/W bit the VPC3+S carries out a read or write operation starting at the specified address inside the internal memory.

The Bus Interface Unit (BIU) and the Dual Port RAM Controller (DPC) that controls accesses to the internal RAM belong to the processor interface of the VPC3+S.

The VPC3+S is supplied with a clock pulse rate of 48MHz. In addition, a clock divider is integrated. The clock pulse is divided by 2 (Pin: DIVIDER = '1') or 4 (Pin: DIVIDER = '0') and applied to the pin CLKOUT. This allows the connection of a slower controller without additional expenditures in a low-cost application.

8.1.3 SPI Interface Mode

The VPC3+S is designed to interface directly with the Serial Peripheral Interface (SPI) port of many of today's popular microcontroller families. It may also interface with microcontrollers that do not have a built-in SPI port by using discrete I/O lines programmed to match the SPI protocol.

The SPI mode allows a duplex, synchronous, serial communication between the CPU and peripheral devices. The CPU is always master while the VPC3+S is always slave in this configuration.

Four associated SPI port pins are dedicated to the SPI function as:

- Slave-Select (SPI_XSS)
- Serial Clock (SPI_SCK)
- Master-Out-Slave-In (SPI_MOSI)
- Master-In-Slave-Out (SPI_MISO)

The clock phase control bit (SPI_CPHA) and the clock polarity control bit (SPI_CPOL) select one of four possible clock formats to be used by the SPI system. The CPOL bit simply selects a non-inverted or inverted clock. The CPHA bit is used to accommodate two fundamentally different protocols by sampling data on odd numbered SCK edges (SPI_CPHA='0') or on even numbered SCK edges (SPI_CPHA='1').

The main element of the SPI system is the SPI Data Register. The 8-bit data register in the master and the 8-bit data register in the slave are linked by the MOSI and MISO pins to form a distributed 16-bit register. When a data transfer operation is performed, this 16-bit register is serially shifted eight bit positions by the SCK clock from the master, so data is exchanged between the master and the slave.

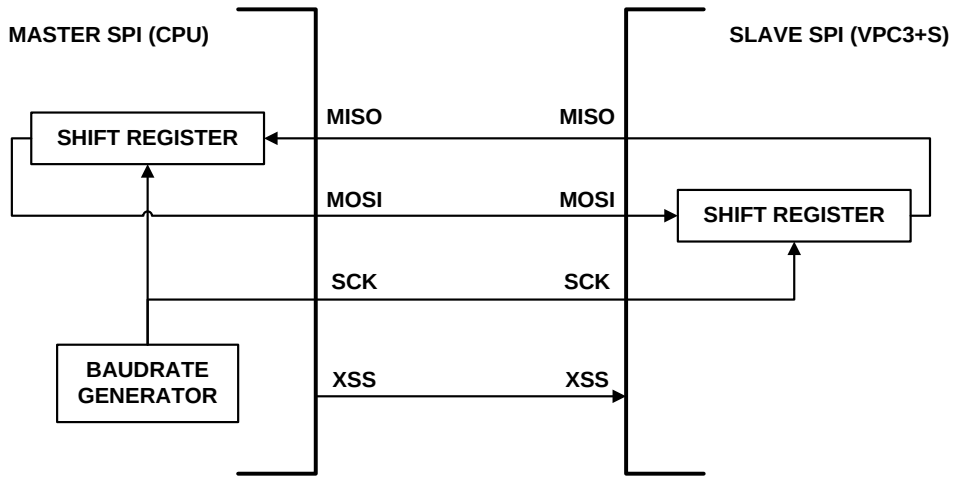


Figure 8-3: SPI Master-Slave-Transfer (Block Diagram)

Data written to the master SPI Data Register becomes the output data for the slave, and data read from the master SPI Data Register after a transfer operation is the input data from the slave.

Transmission Formats

During an SPI transmission, data is transmitted (shifted out serially) and received (shifted in serially) simultaneously. The serial clock (SCK) synchronizes shifting and sampling of the information on the two serial data lines. The slave select line allows selection of an individual slave SPI device, slave devices that are not selected do not interfere with SPI bus activities.

The **CPOL** clock polarity control bit specifies an active high or low clock and has no significant effect on the transmission format. The **CPHA** clock phase control bit selects one of two fundamentally different transmission formats. Clock phase and polarity should be identical for the master SPI device and the communicating slave device.

CPHA = 0 Transfer Format

The first edge on the SCK line is used to clock the first data bit of the slave into the master and the first data bit of the master into the slave. In some peripherals, the first bit of the slave's data is available at the slave's data out pin as soon as the slave is selected. In this format, the first SCK edge is issued a half cycle after SS has become low.

A half SCK cycle later, the second edge appears on the SCK line. When this second edge occurs, the value previously latched from the serial data input pin is shifted into the shift register.

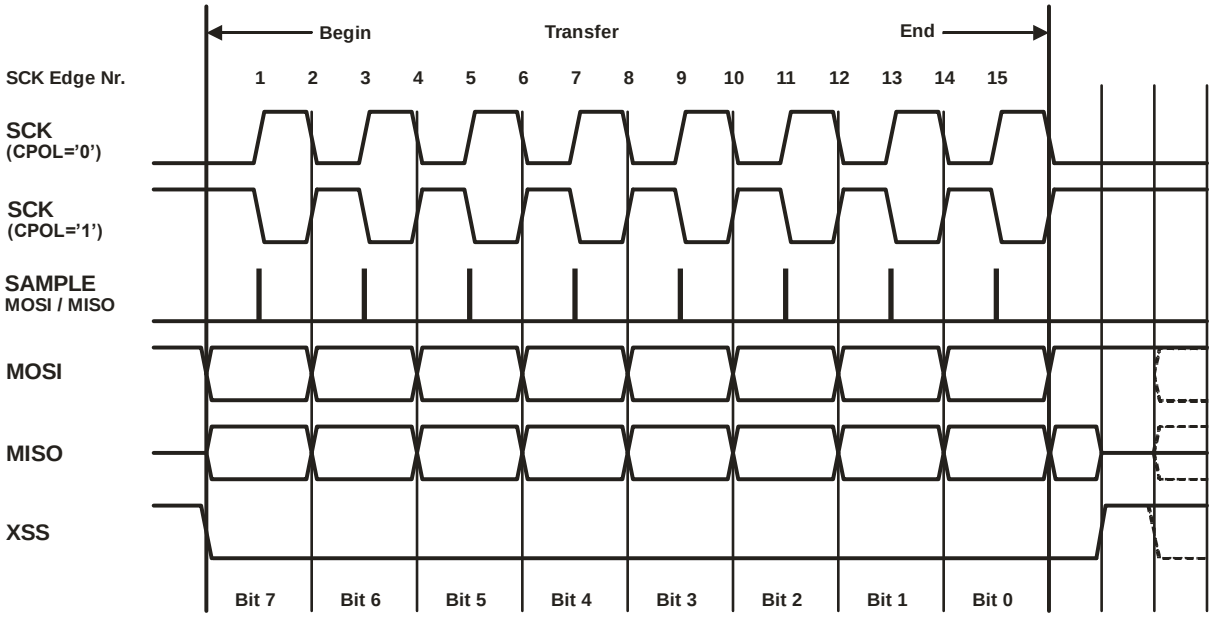


Figure 8-4: SPI Transfer Format (CPHA='0')

After this second edge, the next bit of the SPI master data is transmitted out of the serial data output pin of the master to the serial input pin on the slave. This process continues for a total of 16 edges on the SCK line, with data being latched on odd numbered edges and shifted on even numbered edges.

Data reception is double buffered. Data is shifted serially into the SPI shift register during the transfer and is transferred to the parallel SPI Data Register after the last bit is shifted in.

CPHA = 1 Transfer Format

Some peripherals require the first SCK edge before the first data bit becomes available at the data out pin, the second edge clocks data into the system. In this format, the first SCK edge is issued by setting the CPHA bit at the beginning of the 8-cycle transfer operation.

The first edge of SCK occurs immediately after the half SCK clock cycle synchronization delay. This first edge commands the slave to transfer its first data bit to the serial data input pin of the master.

A half SCK cycle later, the second edge appears on the SCK pin. This is the latching edge for both the master and slave.

When the third edge occurs, the value previously latched from the serial data input pin is shifted into the SPI shift register. After this edge, the next bit of the master data is coupled out of the serial data output pin of the master to the serial input pin on the slave.

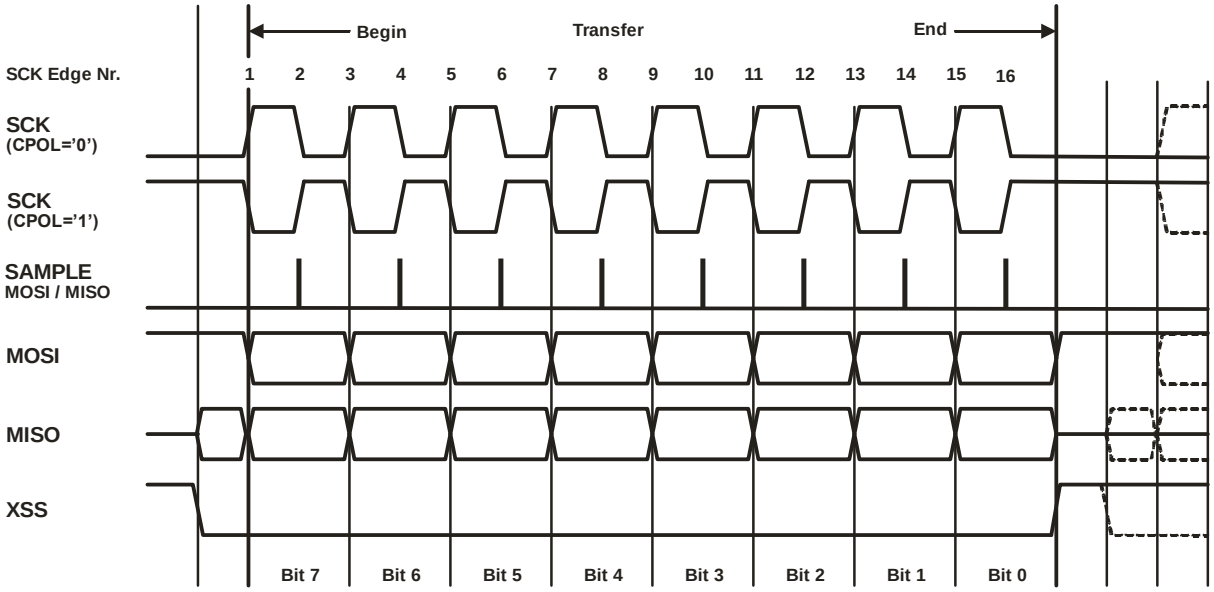


Figure 8-5: SPI Transfer Format (CPHA='1')