Cologne Chip

HFC - S2M
ISDN PRI

ISDN HDLC FIFO controller
with
Primary Rate Interface

preliminary edition: March 2001

Revision History

Date	Remarks
Mar. 2001	Information added to section: Register description, register bit description.
Oct. 2000	Information added to section: Features, block diagram, register bit description.
Feb. 2000	Sections added: Register array description, Correspondency between FIFOs, CHANNELs and SLOTs, Subchannel processing, PCM interface function, Resets, EEPROM, RAM accesses. Information added to section: Register description, register bit description.
Nov. 1999	Register descriptions added for: CONF_EN, F0_CNT_L, F0_CNT_H register.
Sep. 1999	Section added: Register list. Information added to section: Features, PCI buffer signaling and power supply environment, PCI access description, register description, FIFO channel operation: receive channels, FIFO reset, FIFO size setup tables, register bit description.
May 1999	Added: Multiparty audio conferences description, SLOT_DATA / CONFER register bit description. Changes made on: HFC-S2M block diagram, figure of CON_HDLC register bits 7..5 function, SLOT_SEL register bit description.
Apr. 1999	preliminary edition

Cologne Chip AG
Eintrachtstrasse 113
D-50668 Köln
Germany

Tel.: +49 (0) 221 / 91 24-0
Fax: +49 (0) 221 / 91 24-100
http://www.CologneChip.com
http://www.CologneChip.de
info@CologneChip.com

Contents

Figures

Tables

Features

- single chip ISDN-S2M-controller with B- and D-channel HDLC support
- integrated S2M interface
- full I.431 ITU E1/T1 ISDN support in TE and NT mode for 3V - 5V power supply
- 32 independent read and write HDLC-channels
  for e.g. 30 ISDN B-channels, 1 ISDN D-channel and 1 timeslot on the PCM interface
- B-channel transparent mode independently selectable
- up to 64 FIFOs; FIFO-sizes configurable
- each FIFO can be assigned to an arbitrary channel of the S2M or PCM interface
- max. 31 HDLC frames (with 128k or 512k RAM) and 15 HDLC frames (with 32k RAM) per FIFO
- 1 - 8 bit processing for subchannels selectable
- B-channels for higher data rate combinable
- PCM128 / PCM64 / PCM30 interface configurable to interface MITEL ST™ bus (MVIP™) or Siemens IOM2™ and Motorola GCI™ (bearer slots supported only) for inter chip connection (e.g. external CODECs)
- H.100 data rate supported
- integrated ISA Plug and Play interface with buffers for ISA-databus
- integrated PCMCIA interface
- integrated PCI bus interface (Spec. 2.2) for 3.3V and 5V bus signals (PCI power management supported)
- microprocessor interface compatible to Motorala bus and Siemens/Intel bus
- multiparty audio conferences switchable
- DTMF detection on all B-channels
- Timer and watchdog with interrupt capability
- CMOS technology 3V-5V
- PQFP 144 case

1 General description

The HFC-S2M is an ISDN S2M HDLC primary rate controller for all kind of PRI equipment, such like high performance ISDN PC cards, ISDN PABX, ISDN LAN routers, ISDN least cost routers as well as ISDN test equipment. It has an integrated universal bus interface which can be configured into PCI bus, ISA Plug and Play, PCMCIA or microprocessor interface. A PCM128 / PCM64 / PCM30 interface for CODEC or inter chip connection is also integrated.
The very deep FIFOs of the HFC-S2M are realized with an external SRAM.

1.1 Applications

- ISDN PRI terminal adapters
- ISDN PABX for PRI
- ISDN LAN routers for PRI
- ISDN least cost routers for PRI
- ISDN test equipment for PRI

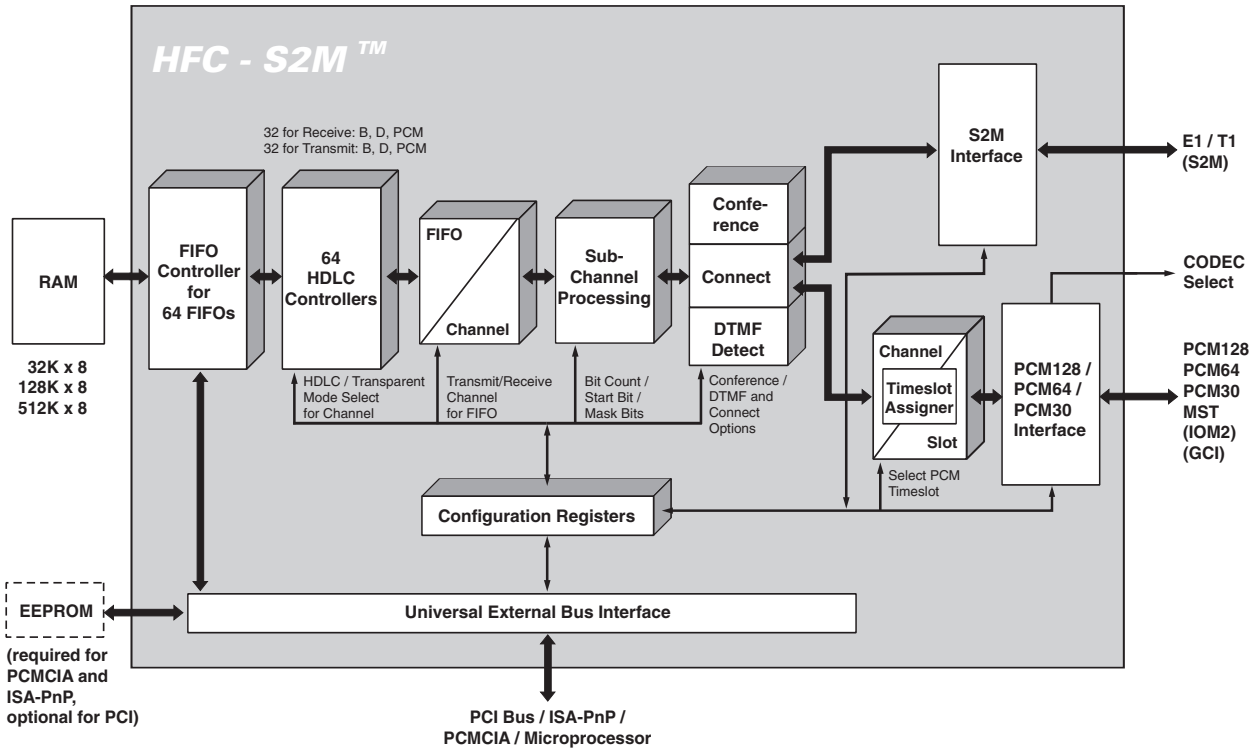

Figure 1: HFC-S2M block diagram

1.2 Bus mode description

The HFC-S2M has an integrated universal external bus interface which can be configured into ISA-PnP, PCI bus, PCMCIA and microprocessor interface.

BMODE1	BMODE0	Bus type
0	0	PCI bus
0	1	Processor mode
1	0	ISA Plug and Play
1	1	PCMCIA

Table 1: Bus mode selection

1.2.1 ISA Plug and Play mode

In ISA PnP mode the HFC-S2M is addressed by eight successive port addresses on the ISA-PC bus. The port address is selected by the lines SA0 - SA15.
The address with SA2='1' is for register selection and the address with SA2='0' is used for data read/write (see also: 2.1).

1.2.2 Processor interface modes

The HFC-S2M has 3 different microprocessor modes:

CPU-Mode 1: Motorola bus with control signals /CS, R/W, /DS

CPU-Mode 2: Siemens/Intel bus with seperated address bus and databus and control signals /CS, /WR, /RD

CPU-Mode 3: Intel bus with multiplexed address and databus with control signals /CS, /WR, /RD, ALE. ALE latches the address. The address lines SA0-SA7 must be connected to the data lines BD0-BD7.

The lines SA0-SA7 are used for direct addressing the internal registers of the HFC-S2M (see also 2.2).

1.2.3 PCI mode

In PCI mode the HFC-S2M is accessed by PCI bus target mode accesses (see also: 2.3).

1.2.4 PCMCIA mode

In PCMCIA mode the HFC-S2M is addressed by eight successive port addresses. The port address is selected by the lines SA0 - SA11.
The address with SA2='1' is for register selection and the address with SA2='0' is used for data read/write (see also: 2.5).

2 Functional description

The HFC-S2M has an integrated universal external bus interface which can be configured into ISA PnP, PCI bus, PCMCIA and microprocessor interface. The following sections show how to use the HFC-S2M in the different interface modes.

2.1 ISA Plug and Play mode

ISA Plug and Play mode is selected by BMODE1=1 and BMODE0=0. The HFC-S2M needs eight consecutive addresses in the I/O map of a PC for operation. Usually also one IRQ line is used. The following section describes how to configure the interrupts of the HFC-S2M.

2.1.1 ISA-PC bus interface

In ISA Plug and Play mode 16 bits SA0 - SA15 are decoded by the address decoder.

SA2	/IOR	/IOW	/AEN	Operation
X	X	X	1	no access
X	1	1	X	no access
0	0	1	0	read data
0	1	0	0	write data
1	0	1	0	read status
1	1	0	0	write control

X = don't care

The HFC-S2M has no memory or DMA access to any component on the ISA-PC bus.

Because of its power drive characteristic it needs no external driver for the ISA-PC bus data lines.

If necessary an external bus driver can be added. In this case the output BUSDIR determines the driver direction.

BUSDIR = 1 means that data is driven into the HFC-S2M;
BUSDIR = 0 means that the HFC-S2M is read and data is driven to the external bus.

2.1.2 IRQ assignment

The IRQ lines are disabled after a hardware reset.
The IRQ assigned by the PnP BIOS can be read from register CHIP_ID (16h), bits [3:0]. Bits [2:0] of the CIRM register have to be set according to the hardware wiring on the PCB and the IRQ number assigned by the PnP BIOS.

2.1.3 ISA Plug and Play control registers

2.1.3.1 Card level control regsisters

Plug and Play control register address	Read Write	Accessable in state	Description
00h	w	Isolation state Config state *)	**Set read data port address register**. Bits[7:0] become bits[9:2] of the port's I/O address. Bits[11:10] are hardwired to 00b and bits[1:0] are hardwired to 11b.
01h	r	Isolation state	**Serial isolation register**. Used to read the serial identifier during the card isolation process.
02h	w	Sleep state, Isolation state, Config state	**Configuration control register**. Bits[7:3] are reserved and must be zero. The defined bits are: 0 **Reset Bit**. When set to one, resets all of the card's configuration registers to their default state. The CSN is not affected. 1 **Return to wait for key state**. When set to one, all cards return to wait for key state. Their CSNs and configuration registers are not affected. This command is issued after all cards have been configured and activated. 2 **Reset CSN to zero**. When set to one, all cards reset their CSN to zero. All bits are automatically cleared by the hardware.

Plug and Play control register address	Read Write	Accessable in state	Description
03h	w	Sleep state, Isolation state, Config state	**Wake command register.** Writing a CSN to this register has the following effects: - If the value written is 00h, all cards in the sleep state with a CSN=00h go to the isolation state. Any card in configure state (CSN not 00h) goes to the sleep state. - If the value written is not 00h, any card in the sleep state with a matching CSN goes to configure state. Any card in the isolation state goes to sleep state. Any write to a card's wake command register with a match on its CSN causes the pointer to the serial identifier / resource data to be reset to the first byte of the serial identifier.

*) This is an extension to the Plug and Play Specification

Plug and Play control register address	Read Write	Accessable in state	Description
04h	r	Config state	**Resource data register.** This register is used to read the device's recource data. Each time that a read is performed from this register a byte of the resource data is returned and the resource data pointer is incremented. Prior to reading each byte, the programmer must read from the status register to determine if the next byte is available for reading from the resource data register. The card's serial identifier and checksum must be read prior to accessing the resource requirement list via this register.
05h	r	Config state	**Status register.** Prior to reading the next byte of the device's resource data, the programmer must read from this register and check bit 0 for a one. This is the resource data byte available bit. Bits[7:1] are reserved.
06h	r/w	Isolation state*), Config state	**Card select number (CSN) register.** The configuration software uses the CSN register to assign a unique ID to the card. The CSN is then used to wake up the card's configuration logic whenever the configuration program must access its configuration registers.

Plug and Play control register address	Read Write	Accessable in state	Description
07h	r	Config state	**Logical device number register.** The number in this register points to the logical device the next commands will operate on. The HFC-S2M only supports one logical device. This register is hardwired to all zeros.

*) only if the isolation process is finished; the last card remains in isolation state until a CSN is assigned.

2.1.3.2 Logical device control registers

Plug and Play control register address	Read Write	Accessable in state	Description
30h	r/w	Config state	**Activate register.** Setting bit 0 to a one activates the card on the ISA bus. When cleared, the card cannot respond to any ISA bus transactions (other than accesses to its Plug and Play configuration ports). Reset clears bit 0. Bits[7:1] are reserved and return zeros when read. The HFC-S2M only supports one logical device, so it is not necessary to write the logical device number into the card's logical device number register prior to writing to this register.
31h	r/w	Config state	**I/O range check register.** Bit(s) Description 7:2 Reserved, return zero when read 1 When set to one, enables I/O range checking and disables it when cleared to zero. When enabled, bit 0 is used to select a pattern for the logical device to return. **This bit is only valid if the logical device is deactivated (see *Activate register*).** 0 When set, the logical device returns 55h in response to any read from the logical device's assigned I/O space. When cleared, AAh is returned.

2.1.4 ISA Plug and Play configuration registers

2.1.4.1 I/O port configuration registers

Plug and Play configuration register address	Read Write	Accessable in state	Description
60h	r/w	Config state	**I/O decoder 0 base address upper byte.** I/O port base address bits[15:8].
61h	r/w	Config state	**I/O decoder 0 base address lower byte.** I/O port base address bits[7:0].

2.1.4.2 Interrupt configuration registers

Plug and Play configuration register address	Read Write	Accessable in state	Description
70h	r/w	Config state	**IRQ select configuration register 0.** Bits[3:0] specify the sleceted IRQ number. Bits[7:4] are reserved.
71h	r/w	Config state	**IRQ type configuration register 0.** Bits[1:0] are ignored. Bits[7:2] are reserved.

☝ **important!**
All registers not implemented return 00h when read except the DMA configuration registers 74h and 75h. These two registers return 04h when read. This means no DMA channel has been selected.

2.1.5 Writing the Plug and Play configuration EEPROM

The EEPROM Writing Spec. is only available on special request to avoid destruction of configuration information by not authorized programs or software viruses.

2.2 Processor mode

Processor mode is selected by BMODE1=0 and BMODE0=1.

In the microprocessor mode the HFC-S2M uses 256 I/O addresses (SA0 - SA7).

/IOR /DS	/IOW R/W	/CS	ALE	Operation	CPU-Mode
X	X	1	X	no access	all
1	1	X	X	no access	all
0	1	0	1	read data	1
0	0	0	1	write data	1
0	1	0	0	read data	2
1	0	0	0	write data	2
0	1	0	0*)	read data	3
1	0	0	0*)	write data	3

X = don't care

*) 1-pulse latches register address.

All registers are directly accessable by their I/O address (see register description).

Except in CPU-mode 3 ALE is assumed to be stable after a RESET.

The HFC-S2M has 3 different microprocessor modes:

CPU-Mode 1: Motorola bus with control signals /CS, R/W, /DS

CPU-Mode 2: Siemens/Intel bus with seperated address bus and databus and control signals /CS, /WR, /RD

CPU-Mode 3: Intel bus with multiplexed address and databus with control signals /CS, /WR, /RD, ALE. ALE latches the address. The address lines SA0-SA7 must be connected to the data lines BD0-BD7.

The lines SA0-SA7 are used for direct addressing the internal registers of the HFC-S2M (see also 2.2).

2.3 PCI interface

PCI mode is selected by BMODE1=0 and BMODE0=0.

2.3.1 PCI access types used by HFC-S2M

C/BE3#	C/BE2#	C/BE1#	C/BE0#	Command Type	HFC-S2M mode
0	0	1	0	I/O Read	target mode
0	0	1	1	I/O Write	target mode
0	1	1	0	Memory Read	target mode
1	1	0	0	Memory Read Multiple	target mode
1	1	1	0	Memory Read Line	target mode
0	1	1	1	Memory Write	target mode
1	1	1	1	Memory Write and Invalidate	target mode
1	0	1	0	Configuration Read	target mode
1	0	1	1	Configuration Write	target mode

Table 2: PCI command types

Memory Read Line and Memory Read Multiple commands are aliased to Memory Read. Memory Write and Invalidate is aliased to Memory Write.

2.3.2 PCI modes supported

The HFC-S2M only supports target mode PCI accesses.

2.3.3 PCI buffer signaling and power supply environment

The HFC-S2M supports 5V and 3.3V PCI bus environments. The environment mode is set during RESET (RST# low) by the input value of /V_SEL (see also: CHIP_ID register bit decription).

PCI bus power and signaling environment	/V_SEL during RST# low
3.3V	high *)
5V	low

*) external pull-up resistor required (10k)

2.3.4 PCI configuration registers

Byte 3	Byte 2	Byte 1	Byte 0	Hex Address
Device ID		**Vendor ID**		00h
Status Register		**Command Register**		04h
Class Code			**Revision ID**	08h
BIST	**Header Type**	**Latency Timer**	**Cache Line Size**	0Ch
I/O Base Address				10h
Memory Base Address				14h
Base Address 2				18h
Base Address 3				1Ch
Base Address 4				20h
Base Address 5				24h
CardBus CIS Pointer				28h
Subsystem ID		**Subsystem Vendor ID**		2Ch
Expansion ROM Base Address				30h
Reserved			**Cap_Ptr**	34h
Reserved				38h
Max_Lat	**Min_Gnt**	**Interrupt Pin**	**Interrupt Line**	3Ch
PMC		**Next Item Ptr**	**Cap_ID**	40h
Data	**PMCSR BSE**	**PMCSR**		44h

■ Register is implemented, value can be read from EEPROM
■ Register is implemented
□ Register is not implemented and returns all 0's when read

The external EEPROM is optional. If no EEPROM is available, EE_SCL/EN must be connected to GND. Without EEPROM the PCI configuration registers will be loaded with the default values shown in Table 3.
All registers which can be read from EEPROM can also be written by configuration write accesses. The addresses for configuration write are shown in the table below.

Register Name	Default Value	Remarks
Vendor ID	1397h	Value can be read from EEPROM. Base address for configuration write is C0h.
Device ID	30BDh	Value can be read from EEPROM. Base address for configuration write is C0h.
Command Register		Bits Function 0 Enables/disables I/O space accesses. 1 Enables/disables memory space accesses. 5..2 fixed to '0' 6 **PERR#** enable/disable 7 fixed to '0' 8 **SERR#** enable/disable 15..9 fixed to '0'
Status Register	0210h	Bits[7:0] can be read from EEPROM. Base address for configuration write is C4h. Bits Function 3..0 reserved 4 fixed to '1' 5 66MHz capable 6 User definable features supported 7 fast Back-to-Back capable 8 data parity error detected 10..9 fixed to '01': timing of DEVSEL# is medium 11 signaled target abort (fixed to '0') 13..12 fixed to '0' 14 signaled system error (Addr. parity error) 15 detected partity error
Revision ID	01h	HFC-S2M Revision 01
Class Code	02 80 00h	Value can be read from EEPROM. Base address for configuration write is C8h.
Header Type	00h	Header Type 0
BIST	00h	No build in self test supported.
I/O Base Address		Bits[31:3] are r/w by configuration accesses
Memory Base Address		Bits[31:8] are r/w by configuration accesses
Subsystem Vendor ID	1397h	Value can be read from EEPROM. Base address for configuration write is ECh.
Subsystem ID	30BDh	Value can be read from EEPROM. Base address for configuration write is ECh.
Cap_Ptr	40h	Offset to Power Management register block.
Interrupt Line	FFh	This register must be configured by configuration write.
Interrupt Pin	01h	INTA supported

Register Name	Default Value	Remarks
Cap_ID	01h	Capability ID. 01h identifies the linked list item as PCI Power Management registers.
Next Ptr	00h	There are no next items in the linked list.
PMC	7E21h	Power Management Capabilities. See also PCI Bus Power Management Interface Specification. This register's value can be read from EEPROM. Base address for configuration write is E0h. **PME#** can be asserted from D0, D1, D2 and $D3_{hot}$. Device specific initialisation is required. The HFC-S2M does not require PCI-clock to generate PME#. This function complies with the PCI Power Management Spec. Version 1.0.
PMCSR	0000h	Power Management Control/Status Bits Function 15 **PME_Status** - This bit is set when the function would normally assert the **PME**# signal independent of the state of the **PME_En** bit. Writing a '1' to this bit will clear it and cause the function to stop asserting a **PME**# (if enabled). Writing a '0' has no effect. 14..9 fixed to '0' 8 **PME_En** - A '1' enables the function to assert **PME#.** When '0', **PME**# assertion is disabled. 7..2 fixed to '0' 1..0 **PowerState** - This 2-bit field is used both to determine the current power state of a function and to set the function into a new power state. 00b - ***D0*** 01b - ***D1*** 10b - ***D2*** 11b - $D3_{hot}$

Table 3: PCI configuration registers' initial values

Unimplemented registers return all 0's when read.

2.4 PCI access description

If the HFC-S2M is used in memory mapped mode all register can directly be accessed by adding their CIP address to the configured Memory Base Address.

In I/O address mapped mode the HFC-S2M occupies 8 bytes in the I/O address space. Byte 0 is for data read/write, byte 4 for register selection. The AUX-port address is selected by byte 3, AUX-port data is read/written by byte 1.

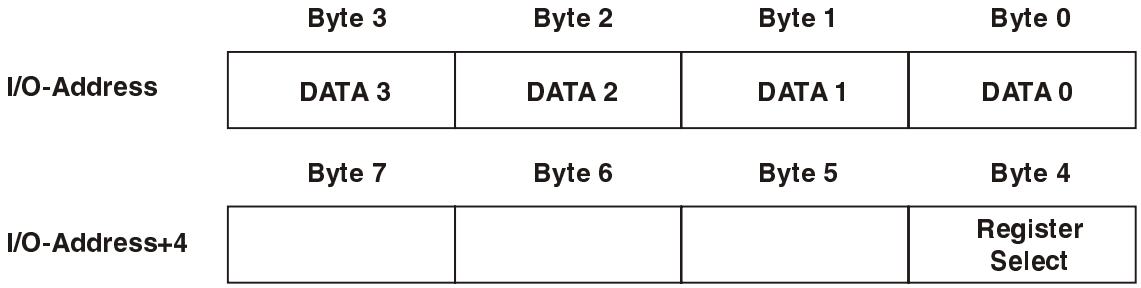

Figure 2: HFC-S2M in I/O address mapped mode

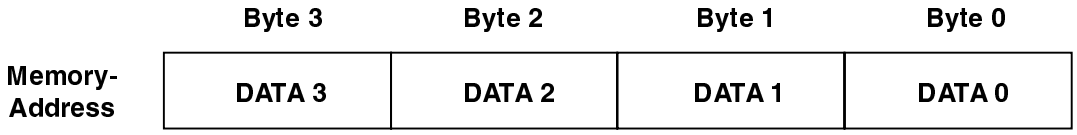

Figure 3: HFC-S2M in memory address mapped mode

2.5 PCMCIA mode

2.5.1 Internal HFC-S2M register selection

The HFC-S2M occupies eight consecutive addresses in the I/O map. The base I/O address must be 8 byte aligned so the lower of both addresses is the one with SA2 = 0 and the higher address is the one with SA2 = 1. The lines SA3 to SA15 are don't care. The registers of the HFC-S2M are selected by writing the registers' address to the higher I/O address (SA2=1). Registers are read/written by reading/writing the base I/O address (SA2=0).

2.5.2 Attribute memory

After hardware reset the card's information structure (CIS) is copied from the EEPROM to even numbered addresses of the SRAM starting with 0000h (512 byte are occupied for the CIS). To avoid accesses in this phase the /WAIT signal is active.

2.5.3 PCMCIA registers

Configuration Option Register (COR): Register address: 400h in Configuration Memory

D7	D6	D5	D4	D3	D2	D1	D0
SRESET	LevIREQ 1	Configuration Index					

The fields are as follows:

SRESET	SRESET card. Setting this bit to one places the card in the reset state. This bit must be cleared to zero by the user.
LevIREQ	This bit is not implemented and returns always 1 when read to indicate usage of level mode interrupts.
Configuration Index	Configuration Index. Bit 0 must be set to 1 to enable I/O accesses to the HFC-S2M. Bit 5 must be set to 1 to write data to the EEPROM.

Card Configuration and Status Register (CSR): Register address: 402h in Configuration Memory

D7	D6	D5	D4	D3	D2	D1	D0
Changed 0	SigChg 0	IOis8 1	Rsvd 0	Audio 0	PwrDwn 0	Intr	Rsvd 0

The fields are as follows:

Changed SigChg Rsvd Audio PwrDwn	Unimplemented and return 0 when read.
IOis8	Unimplemented and return 1 when read to indicate an 8 bit data path.
Intr	Internal state of interrupt request (IREQ).

2.5.4 CIS programming

The EEPROM Programming Spec. is only available on special request to avoid destruction of configuration information by not authorized programs or software viruses.

2.6 Internal HFC-S2M register description

In ISA Plug and Play mode, PCI mode and PCMCIA mode all registers are selected by writing the register address into the Control Internal Pointer (CIP) register. This is done by writing the HFC-S2M on the higher I/O address (SA2 = 1).
All consecutive read or write data accesses (SA2 = 0) are done with the selected register until the CIP register is changed.

In processor mode all internal registers can be directly accessed. The registers are selected by SA0 - SA7. In PCI mode SA0 and SA1 are generated from the byte enable lines.

2.6.1 Register array description

As you will see in the register list on the following pages some registers are listed like follows:

REGISTER_NAME [INDEX_REGISTER_NAME]

e.g.: FIF_DATA [FIFO#]

This means that the register with the name REGISTER_NAME is available n times (register array). For example: The HFC-S2M can handle up to 64 FIFOs so each of the FIFO control/configuration registers is available 64 times. 6 registers are required to control one FIFO. Additionally there are 6 registers for configuration. The register FIFO# indexes 1 of 64 blocks of 12 registers for the desired FIFO.

REGISTER_NAME [INDEX_REGISTER_NAME]

> REGISTER_NAME is the name of the register to access
> INDEX_REGISTER_NAME is the name of the index register which makes the corresponding register for REGISTER_NAME accessible.

Index Register Name	Depending Registers
FIFO#	FIF_DATA FIF_F1 FIF_F2 FIF_Z1H FIF_Z1L FIF_Z2H FIF_Z2L CH_MASK CON_HDLC HDLC_PAR CHANNEL# F_SEQ INT_MASK
SLOT#	CONFER SLOT_CFG

Examples:

To access register HDLC_PAR for FIFO number 12 do the following:

1) write 12 to register FIFO# (register address is 0Fh)
2) register HDLC_PAR can be read/written (register address is FBh)

To access register SLOT_CFG for PCM time slot 20 do the following:

1) write 20 to register SLOT# (register address is 10h)
2) register SLOT_CFG can be read/written (register address is D0h)

2.6.2 FIFO registers

The FIFO control registers are used to select and control the FIFOs of the HFC-S2M. The FIFO register selection is independent of the B- or D-channel FIFO number. The FIFO is selected by the FIFO select register (FIFO#).

CIP / I/O-address		Name	r/w	Function
00001111	0Fh	FIFO#	w	select FIFO#
00001011	0Bh	F_INIT	w	number of first FIFO in sequence
00001101	0Dh	F_MODE	w	FIFO mode
11111101	FDh	F_SEQ [FIFO#]	w	next FIFO in FIFO sequence
00001100	0Ch	R_SIZE	w	RAM size setup register
00001110	0Eh	INC_RES_F	w	increment F-counter/reset FIFO
100000xx	80h..83h	FIF_DATA [FIFO#]	r/w	data read/write FIFO and increment Z-counter (80h => 1 byte is accessed, 83h => 4 bytes are accessed)
100001xx	84h..87h	FIF_DATA [FIFO#]	r/w	data read/write FIFO without incrementing Z-counter (84h => 1 byte is accessed, 87h => 4 bytes are accessed)
00000100	04h	FIF_Z1L [FIFO#]	r	FIFO input counter (Z1) low byte
00000101	05h	FIF_Z1H [FIFO#]	r	FIFO input counter (Z1) high byte
00000110	06h	FIF_Z2L [FIFO#]	r	FIFO output counter (Z2) low byte
00000111	07h	FIF_Z2H [FIFO#]	r	FIFO output counter (Z2) high byte
00001100	0Ch	FIF_F1 [FIFO#]	r	FIFO input HDLC frame counter (F1)
00001101	0Dh	FIF_F2 [FIFO#]	r	FIFO output HDLC frame counter (F2)

☝ **important!**

FIFO change, FIFO reset and F1/F2 incrementation
Changing the FIFO, reseting the FIFO or incrementing the frame counters causes a short BUSY period of the HFC-S2M. This means an access to FIFO control registers is NOT allowed until BUSY status is reset (bit 0 of STATUS register). This has a maximum duration of 25 clock cycles (2µs). Status, interrupt and control registers can be read and written at any time.

2.6.3 Registers of the S2M section

CIP / I/O-address		Name	r/w	Function
00100010	22h	LOS0	w	LOS alarm
00100011	23h	LOS1	w	LOS alarm
00100100	24h	REC_STAT0 RECEIVE0	r w	synchronisation status register 0 receive register
00100101	25h	REC_STAT1 REC_FRAME	r w	synchronisation status register 1 receive frame setup register
00100110	26h	REC_STAT2	r	synchronisation status register 2
00100111	27h	REC_STAT3	r	synchronisation status register 3
00101000	28h	TRANSM0	w	transmit setup register 0
00101001	29h	TRANSM1	w	transmit setup register 1
00101100	2Ch	TRANS_FRA0	w	transmit frame setup register 0
00101101	2Dh	TRANS_FRA1	w	transmit frame setup register 1
00101110	2Eh	TRANS_FRA2	w	transmit frame setup register 2
00110000	30h	RECEIVE_OFF	w	receive offset setup register
00110100	34h	TRANS_OFF	w	transmit offset setup register
00100000	20h	STATE / FSM	r/w	S2M state register
00110000	30h	E_C_FAS_L	r	Error count FAS (low byte)
00110001	31h	E_C_FAS_H	r	Error count FAS (high byte)
00110010	32h	E_C_VIO_L	r	Error count VIO (low byte)
00110011	33h	E_C_VIO_H	r	Error count VIO (high byte)
00110100	34h	E_C_CRC_L	r	Error count CRC (low byte)
00110101	35h	E_C_CRC_H	r	Error count CRC (high byte)
00110110	36h	E_C_E12_L	r	Error count E1,2 (low byte)
00110111	37h	E_C_E12_H	r	Error count E1,2 (high byte)

CIP / I/O-address		Name	r/w	Function
00111000	38h	E_C_SA613_L	r	Error count SA6,1,3 (low byte)
00111001	39h	E_C_SA613_H	r	Error count SA6,1,3 (high byte)
00111010	3Ah	E_C_SA623_L	r	Error count SA6,2,3 (low byte)
00111011	3Bh	E_C_SA623_H	r	Error count SA6,2,3 (high byte)

2.6.4 Registers of the PCM bus section

CIP / I/O-address		Name	r/w	Function
00010100	14h	MST_MODE0	w	mode register for PCM bus and register at address 15h select
00010101	15h	MST_MODE1	w	extended mode register for PCM bus
00010101	15h	MST_MODE2	w	extended mode register for OKI™ codecs and PCM synchronization
00010101	15h	SLOT_SEL0	w	Slot number/SHAPE select for F_Q0
00010101	15h	SLOT_SEL1	w	Slot number/SHAPE select for F_Q1
00010101	15h	SLOT_SEL2	w	Slot number/SHAPE select for F_Q2
00010101	15h	SLOT_SEL3	w	Slot number/SHAPE select for F_Q3
00010101	15h	SLOT_SEL4	w	Slot number/SHAPE select for F_Q4
00010101	15h	SLOT_SEL5	w	Slot number/SHAPE select for F_Q5
00010101	15h	SLOT_SEL6	w	Slot number/SHAPE select for F_Q6
00010101	15h	SLOT_SEL7	w	Slot number/SHAPE select for F_Q7
00010101	15h	SHAPE0_L	w	CODEC enable signal SHAPE 0 (low byte)
00010101	15h	SHAPE0_H	w	CODEC enable signal SHAPE 0 (high byte)
00010101	15h	SHAPE1_L	w	CODEC enable signal SHAPE 1 (low byte)
00010101	15h	SHAPE1_H	w	CODEC enable signal SHAPE 1 (high byte)
00011000	18h	F0_CNT_L	r	F0IO pulse count (low byte)
00011001	19h	F0_CNT_H	r	F0IO pulse count (high byte)
00010000	10h	SLOT#	w	select PCM bus slot#
11010000	D0h	SLOT_CFG [SLOT#]	w	PCM bus data channel configuration register
11010001	D1h	CONFER [SLOT#]	w	multiparty audio conference setup register
00011000	18h	CONF_EN	w	Conference enable register

2.6.5 Interrupt, status and control registers

CIP / I/O address		Name	r/w	Function
00010110	16h	CHIP_ID	r	register for chip identification and ISA PnP/PCI setup
00000000	00h	CIRM	w	interrupt selection and reset register
00000001	01h	X_ACC_EN	w	external access enable register
11110100	F4h	CH_MASK [FIFO#]	w	bit value for not processed bits in channel
00010011	13h	INT_CTRL	w	FIFO interrupt control register
00010000	10h	INT_OVIEW	r	FIFO interrupt overview register
00010001	11h	INT_MISC	r	Miscellaneous interrupts register
00010010	11h	INT_MISC_M	w	Miscellaneous interrupts mask register
11001000	C8h	INT_FIF_0	r	FIFO 0..7 interrupt register
11001001	C9h	INT_FIF_1	r	FIFO 8..15 interrupt register
11001010	CAh	INT_FIF_2	r	FIFO 16..23 interrupt register
11001011	CBh	INT_FIF_3	r	FIFO 24..31 interrupt register
11001100	CCh	INT_FIF_4	r	FIFO 32..39 interrupt register
11001101	CDh	INT_FIF_5	r	FIFO 40..47 interrupt register
11001110	CEh	INT_FIF_6	r	FIFO 48..55 interrupt register
11001111	CFh	INT_FIF_7	r	FIFO 56..63 interrupt register
11111111	FFh	INT_MASK [FIFO#]	w	interrupt mask register
00011010	1Ah	TIMER_WD	w	timer/watchdog period selection register
00011011	1Bh	WD_MODE	w	watchdog mode and watchdog reset register
00011100	1Ch	STATUS	r	common status register
11111010	FAh	CON_HDLC [FIFO#]	w	connect functions for S2M, HFC, PCM and HDLC setup register
11111011	FBh	HDLC_PAR [FIFO#]	w	HDLC and transparent mode parameter register
11111100	FCh	CHANNEL# [FIFO#]	w	link FIFO to ISDN channel
00001000	08h	RAM_ADR_0	w	address bits 7..0 for RAM accesses
00001001	09h	RAM_ADR_1	w	address bits 15..8 for RAM accesses
00001010	0Ah	RAM_ADR_2	w	address bits 23..16 for RAM accesses
11000000	C0h	RAM_DATA	r	SRAM data

2.6.6 Register Reference List

Registers by Address				
Address	**Name**	**Reset**	**Reset Value**	**Page**
00h	CIRM	0	00h	48
01h	X_ACC_EN	0	00h	48
04h	FIF_Z1L [ ]	1, 4	max	53
05h	FIF_Z1H [ ]	1, 4	max	53
06h	FIF_Z2L [ ]	1, 4	max	53
07h	FIF_Z2H [ ]	1, 4	max	53
08h	RAM_ADR_0	4	00h	51
09h	RAM_ADR_1	4	00h	51
0Ah	RAM_ADR_2	4	00h	51
0Bh	F_INIT	1, 4	00h	55
0Ch	FIF_F1 [ ]	1, 4	max	52
0Ch	R_SIZE	0	00h	51
0Dh	F_MODE	0	00h	51
0Dh	FIF_F2 [ ]	1, 4	max	52
0Eh	INC_RES_F	-	-	51
0Fh	FIFO#	1, 4	00h	51
10h	SLOT#	2, 4	00h	58
10h	INT_OVIEW	1, 4	00h	49
11h	INT_MISC	1, 4	00h	49
11h	INT_MISC_M	1, 4	00h	49
13h	INT_CTRL	1, 4	00h	49
14h	MST_MODE0	2, 4	00h	59
15h	MST_MODE1	2, 4	00h	62
15h	MST_MODE2	2, 4	00h	62
15h	SHAPE0_H	2, 4	00h	63
15h	SHAPE0_L	2, 4	00h	63
15h	SHAPE1_H	2, 4	00h	63
15h	SHAPE1_L	2, 4	00h	63
15h	SLOT_SEL0	2, 4	00h	59
15h	SLOT_SEL1	2, 4	00h	60
15h	SLOT_SEL2	2, 4	00h	60
15h	SLOT_SEL3	2, 4	00h	60
15h	SLOT_SEL4	2, 4	00h	60
15h	SLOT_SEL5	2, 4	00h	60
15h	SLOT_SEL6	2, 4	00h	60
15h	SLOT_SEL7	2, 4	00h	61
16h	CHIP_ID	-	-	57
18h	F0_CNT_L	1, 4	00h	64
18h	CONF_EN	2, 4	00h	59
19h	F0_CNT_H	1, 4	00h	64
1Ah	TIMER_WD	1, 4	00h	56
1Bh	WD_MODE	1, 4	00h	56
1Ch	STATUS	-	-	57
20h	STATE / FSM	3, 4	deact.	70
22h	LOS0	3, 4	00h	65
23h	LOS1	3, 4	00h	65
24h	RECEIVE0	3, 4	00h	65
24h	REC_STAT0	3, 4	00h	65
25h	REC_FRAME	3, 4	00h	67
25h	REC_STAT1	3, 4	00h	66
26h	REC_STAT2	3, 4	00h	66

Registers by Address				
Address	**Name**	**Reset**	**Reset Value**	**Page**
27h	REC_STAT3	3, 4	00h	66
28h	TRANSM0	3, 4	00h	68
29h	TRANSM1	3, 4	00h	68
2Ch	TRANS_FRA0	3, 4	00h	69
2Dh	TRANS_FRA1	3, 4	00h	69
2Eh	TRANS_FRA2	3, 4	00h	69
30h	E_C_FAS_L	3, 4	00h	70
30h	RECEIVE_OFF	3, 4	00h	70
31h	E_C_FAS_H	3, 4	00h	70
32h	E_C_VIO_L	3, 4	00h	70
33h	E_C_VIO_H	3, 4	00h	70
34h	E_C_CRC_L	3, 4	00h	70
34h	TRANS_OFF	3, 4	00h	70
35h	E_C_CRC_H	3, 4	00h	70
36h	E_C_E12_L	3, 4	00h	70
37h	E_C_E12_H	3, 4	00h	70
38h	E_C_SA613_L	3, 4	00h	70
39h	E_C_SA613_H	3, 4	00h	70
3Ah	E_C_SA623_L	3, 4	00h	70
3Bh	E_C_SA623_H	3, 4	00h	70
80h..83h	FIF_DATA [ ]	-	-	52
C0h	RAM_DATA	-	-	51
C8h	INT_FIF_0	1, 4	00h	50
C9h	INT_FIF_1	1, 4	00h	50
CAh	INT_FIF_2	1, 4	00h	50
CBh	INT_FIF_3	1, 4	00h	50
CCh	INT_FIF_4	1, 4	00h	50
CDh	INT_FIF_5	1, 4	00h	50
CEh	INT_FIF_6	1, 4	00h	50
CFh	INT_FIF_7	1, 4	00h	50
D0h	SLOT_CFG [ ]	-	-	58
D1h	CONFER [ ]	-	-	58
F4h	CH_MASK [ ]	1, 4	FFh	55
FAh	CON_HDLC [ ]	1, 4	00h	54
FBh	HDLC_PAR [ ]	1, 4	00h	55
FCh	CHANNEL# [ ]	1, 4	Fif-No	55
FDh	F_SEQ [ ]	1, 4	Fif-No	55
FFh	INT_MASK [ ]	1, 4	00h	55

Reset 0 = hardware reset; Reset 1 = HFC reset; Reset 2 = PCM reset; Reset 3 = S2M reset; Reset 4 = soft reset (= reset 1+2+3)

Registers by Name				
Address	**Name**	**Reset**	**Reset Value**	**Page**
F4h	CH_MASK [ ]	1, 4	FFh	55
FCh	CHANNEL# [ ]	1, 4	Fif-No	55
16h	CHIP_ID	-	-	57
00h	CIRM	0	00h	48
FAh	CON_HDLC [ ]	1, 4	00h	54
18h	CONF_EN	2, 4	00h	59
D1h	CONFER [ ]	-	-	58
35h	E_C_CRC_H	3, 4	00h	70
34h	E_C_CRC_L	3, 4	00h	70
37h	E_C_E12_H	3, 4	00h	70
36h	E_C_E12_L	3, 4	00h	70
31h	E_C_FAS_H	3, 4	00h	70
30h	E_C_FAS_L	3, 4	00h	70
39h	E_C_SA613_H	3, 4	00h	70
38h	E_C_SA613_L	3, 4	00h	70
3Bh	E_C_SA623_H	3, 4	00h	70
3Ah	E_C_SA623_L	3, 4	00h	70
33h	E_C_VIO_H	3, 4	00h	70
32h	E_C_VIO_L	3, 4	00h	70
0Bh	F_INIT	1, 4	00h	55
0Dh	F_MODE	0	00h	51
FDh	F_SEQ [ ]	1, 4	Fif-No	55
19h	F0_CNT_H	1, 4	00h	64
18h	F0_CNT_L	1, 4	00h	64
80h..83h	FIF_DATA [ ]	-	-	52
0Ch	FIF_F1 [ ]	1, 4	max	52
0Dh	FIF_F2 [ ]	1, 4	max	52
05h	FIF_Z1H [ ]	1, 4	max	53
04h	FIF_Z1L [ ]	1, 4	max	53
07h	FIF_Z2H [ ]	1, 4	max	53
06h	FIF_Z2L [ ]	1, 4	max	53
0Fh	FIFO#	1, 4	00h	51
FBh	HDLC_PAR [ ]	1, 4	00h	55
0Eh	INC_RES_F	-	-	51
13h	INT_CTRL	1, 4	00h	49
C8h	INT_FIF_0	1, 4	00h	50
C9h	INT_FIF_1	1, 4	00h	50
CAh	INT_FIF_2	1, 4	00h	50
CBh	INT_FIF_3	1, 4	00h	50
CCh	INT_FIF_4	1, 4	00h	50
CDh	INT_FIF_5	1, 4	00h	50
CEh	INT_FIF_6	1, 4	00h	50
CFh	INT_FIF_7	1, 4	00h	50
FFh	INT_MASK [ ]	1, 4	00h	55
11h	INT_MISC	1, 4	00h	49
11h	INT_MISC_M	1, 4	00h	49
10h	INT_OVIEW	1, 4	00h	49
22h	LOS0	3, 4	00h	65
23h	LOS1	3, 4	00h	65
14h	MST_MODE0	2, 4	00h	59
15h	MST_MODE1	2, 4	00h	62

Registers by Name				
Address	**Name**	**Reset**	**Reset Value**	**Page**
15h	MST_MODE2	2, 4	00h	62
0Ch	R_SIZE	0	00h	51
08h	RAM_ADR_0	4	00h	51
09h	RAM_ADR_1	4	00h	51
0Ah	RAM_ADR_2	4	00h	51
C0h	RAM_DATA	-	-	51
25h	REC_FRAME	3, 4	00h	67
24h	REC_STAT0	3, 4	00h	65
25h	REC_STAT1	3, 4	00h	66
26h	REC_STAT2	3, 4	00h	66
27h	REC_STAT3	3, 4	00h	66
30h	RECEIVE_OFF	3, 4	00h	70
24h	RECEIVE0	3, 4	00h	65
15h	SHAPE0_H	2, 4	00h	63
15h	SHAPE0_L	2, 4	00h	63
15h	SHAPE1_H	2, 4	00h	63
15h	SHAPE1_L	2, 4	00h	63
10h	SLOT#	2, 4	00h	58
D0h	SLOT_CFG [ ]	-	-	58
15h	SLOT_SEL0	2, 4	00h	59
15h	SLOT_SEL1	2, 4	00h	60
15h	SLOT_SEL2	2, 4	00h	60
15h	SLOT_SEL3	2, 4	00h	60
15h	SLOT_SEL4	2, 4	00h	60
15h	SLOT_SEL5	2, 4	00h	60
15h	SLOT_SEL6	2, 4	00h	60
15h	SLOT_SEL7	2, 4	00h	61
20h	STATE / FSM	3, 4	deact.	70
1Ch	STATUS	-	-	57
1Ah	TIMER_WD	1, 4	00h	56
2Ch	TRANS_FRA0	3, 4	00h	69
2Dh	TRANS_FRA1	3, 4	00h	69
2Eh	TRANS_FRA2	3, 4	00h	69
34h	TRANS_OFF	3, 4	00h	70
28h	TRANSM0	3, 4	00h	68
29h	TRANSM1	3, 4	00h	68
1Bh	WD_MODE	1, 4	00h	56
01h	X_ACC_EN	0	00h	48

Reset 0 = hardware reset; Reset 1 = HFC reset; Reset 2 = PCM reset; Reset 3 = S2M reset; Reset 4 = soft reset (= reset 1+2+3)

2.7 Timer / watchdog

The HFC-S2M includes a timer and a watchdog with interrupt capability.
The timer counts F0IO pulses. So the timer counter is incremented every 125µs. The watchdog counter is incremented every 2ms.
The timer values for timer and watchdog can be selected by the TIMER_WD register. 16 different timer and watchdog values can be selected.
The watchdog can be manually resetted by setting bit 7 of the WD_MODE register. Furthermore the watchdog is resetted at every access to the HFC-S2M if bit 5 of the WD_MODE register is set.

2.8 FIFOs

There are up to 64 FIFOs with 64 HDLC-controllers in the HFC-S2M. The HDLC circuits are located on the S2M device side of the HFC-S2M. So always plain data is stored in the FIFO. Zero insertion and deletion is done in HDLC mode:

– if the data goes to the S2M or PCM device in send FIFOs and
– when the HDLC data comes from the S2M device or PCM bus in receive operation.

There are a send and a receive FIFO for each B-channel and for each D-channel. A FIFO can be selected for access by writing its number in the FIFO select register (FIFO#).

The FIFOs are realized as ring buffers in the external SRAM. To control them there are some counters. The size of the counters is depinding on the setting of the FIFO size (see also: FIFO size setup on page 37). Z1 is the FIFO input counter and Z2 is the FIFO output counter.
Each counter points to a byte position in the SRAM. On a FIFO input operation Z1 is incremented. On an output operation Z2 is incremented.

After every pulse on the F0IO signal two HDLC-bytes are written into the S2M interface (FIFOs with even No.) and two HDLC-bytes are read from the S2M interface (FIFOs with odd No.).
D-channel data is handled in a similar way but only 2 bits are processed.

> ☝ **important!**
> Instead of the S2M interface also PCM bus is selectable for each B-channel (see CON_HDLC register).

If Z1 = Z2 the FIFO is empty.

Additionally there are two counters F1 and F2 for every FIFO channel (5Bit for B-channel, 4Bit for D-channel). They count the HDLC-frames in the FIFOs and form a ring buffer as Z1 and Z2 do, too.

F1 is incremented when a complete frame has been received and stored in the FIFO. F2 is incremented when a complete frame has been read from the FIFO.

If F1 = F2 there is no complete frame in the FIFO.

When the RESET line is active or software reset is active Z1, Z2, F1 and F2 are all initialized to all 1s.

The access to a FIFO is selected by writing the FIFO number into the FIFO select register (FIFO#).

☝ **important!**

FIFO change, FIFO reset and F1/F2 incrementation
Changing the FIFO, reseting the FIFO or incrementing the frame counters causes a short BUSY period of the HFC-S2M. This means an access to FIFO control registers is NOT allowed until BUSY status is reset (bit 0 of STATUS register). This has a maximum duration of 25 clock cycles (2µs). Status, interrupt and control registers can be read and written at any time.

☝ **important!**
The counter state Z_{MIN} of the Z-counters follows counter state Z_{MAX} in the B- and D-channel FIFOs. Please note that Z_{MIN} and Z_{MAX} are depending on the FIFO number (see also: FIFO size setup on page 37).

2.8.1 FIFO channel operation

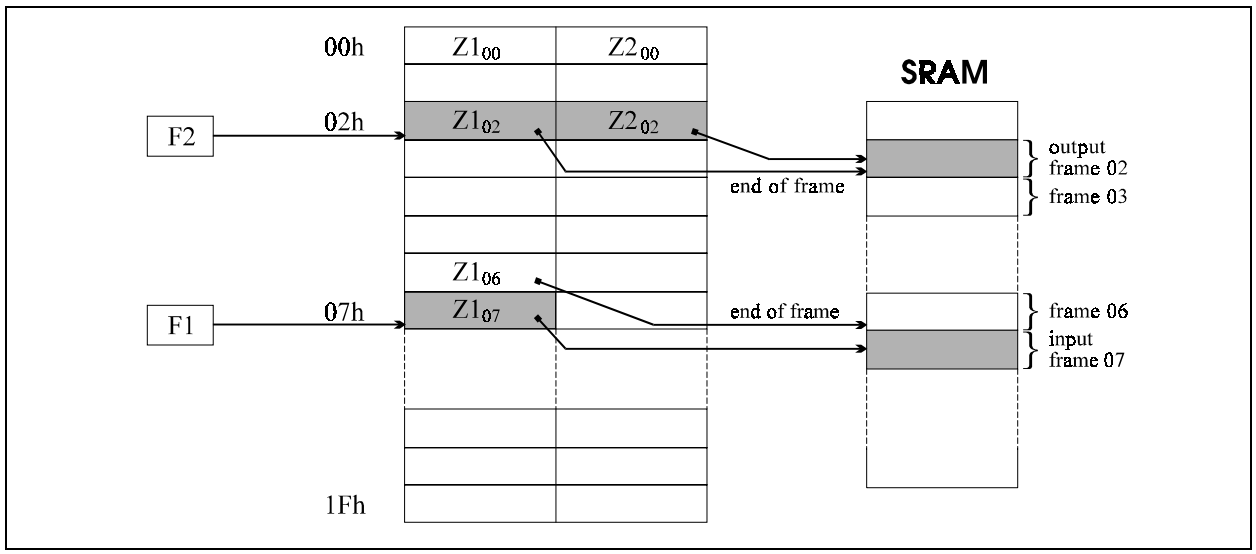

Figure 4: FIFO Organisation (shown for B-channel, similar for D-channel)

2.8.1.1 Send channels (B1, B2 and D transmit)

The send channels send data from the host bus interface to the FIFO and the HFC-S2M converts the data into HDLC code and tranfers it from the FIFO into the S2M or/and the PCM bus interface write registers.

The HFC-S2M checks Z1 and Z2. If Z1=Z2 (FIFO empty) the HFC-S2M generates a HDLC-Flag (01111110) and sends it to the S2M device. In this case Z2 is not incremented. If also F1=F2 only HDLC flags are sent to the S2M interface and all counters remain unchanged. If the frame counters are unequal F2 is incremented and the HFC-S2M tries to send the next frame to the output device. After the end of a frame (Z2 reaches Z1) it automatically generates the 16 bit CRC checksum and adds the ending flag. If there is another frame in the FIFO (F1≠F2) the F2 counter is incremented.

With every byte being sent from the host bus side to the FIFO Z1 is incremented automatically. If a complete frame has been sent F1 must be incremented to send the next frame. If the frame counter F1 is incremented also the Z-counters may change because Z1 and Z2 are functions of F1 and F2. So there are Z1(F1), Z2(F1), Z1(F2) and Z2(F2) (see Figure 4).
Z1(F1) is used for the frame which is just written from the PC-bus side. Z2(F2) is used for the frame which is just beeing transmitted to the S2M device side of the HFC-S2M. Z1(F2) is the end of frame pointer of the current output frame.

In the send channels F1 is only changed from the PC interface side if the software driver wants to say “end of send frame”. Then the current value of Z1 is stored, F1 is incremented and Z1 is used as start address of the next frame. Z1(F2) and Z2(F2) can not be accessed.

2.8.1.2 Automatically D-channel frame repetition

The D-channel send FIFO has a special feature. If the S2M interface signals a D-channel contention before the CRC is sent the Z2 counter is set to the starting address of the current frame and the HFC-S2M tries to repeat the frame automatically.

> ☝ **important!**
> The HFC-S2M begins to transmit the bytes from a FIFO at the moment the FIFO is changed or the F1 counter is incremented. Also changing to the FIFO that is already selected starts the transmission. So by selecting the same FIFO again transmission can be started.

2.8.1.3 FIFO full condition in send channels

Due to the limited number of registers in the HFC-S2M the driver software must maintain a list of frame start and end addresses to calculate actual FIFO size and check FIFO full condition. Because there are a maximum of 32/16 frame counter values and the start address of a frame is the incremented value of the end address of the last frame the memory table must have only 32/16 values of 16 bits instead of 64/32.

Remember that an increment of Z-value Z_{MAX} is Z_{MIN} in the B- and D-channels!

There are two different FIFO full conditions. The first one is met when the FIFO contents comes up to 31 frames (128k or 512k RAM) or 15 frames (32k RAM). There is no possibility for the HFC-S2M to manage more frames even if the frames are very small.

The second limitation is the size of the FIFO (see also: FIFO size setup on page 37).

2.8.1.4 Receive Channels (B1, B2 and D receive)

The receive channels receive data from the S2M or PCM bus interface read registers. The data is converted from HDLC into plain data and sent to the FIFO. The data can then be read via the host bus interface.

The HFC-S2M checks the HDLC data coming in. If it finds a flag or more than 5 consecutive 1s it does not generate any output data. In this case Z1 is not incremented. Proper HDLC data being received is converted by the HFC-S2M into plain data. After the ending flag of a frame the HFC-S2M checks the HDLC CRC checksum. If it is correct one byte with all 0s is inserted behind the CRC data in the FIFO named STAT. This last byte of a frame in the FIFO is different from all 0s if there is no correct CRC field at the end of the frame.

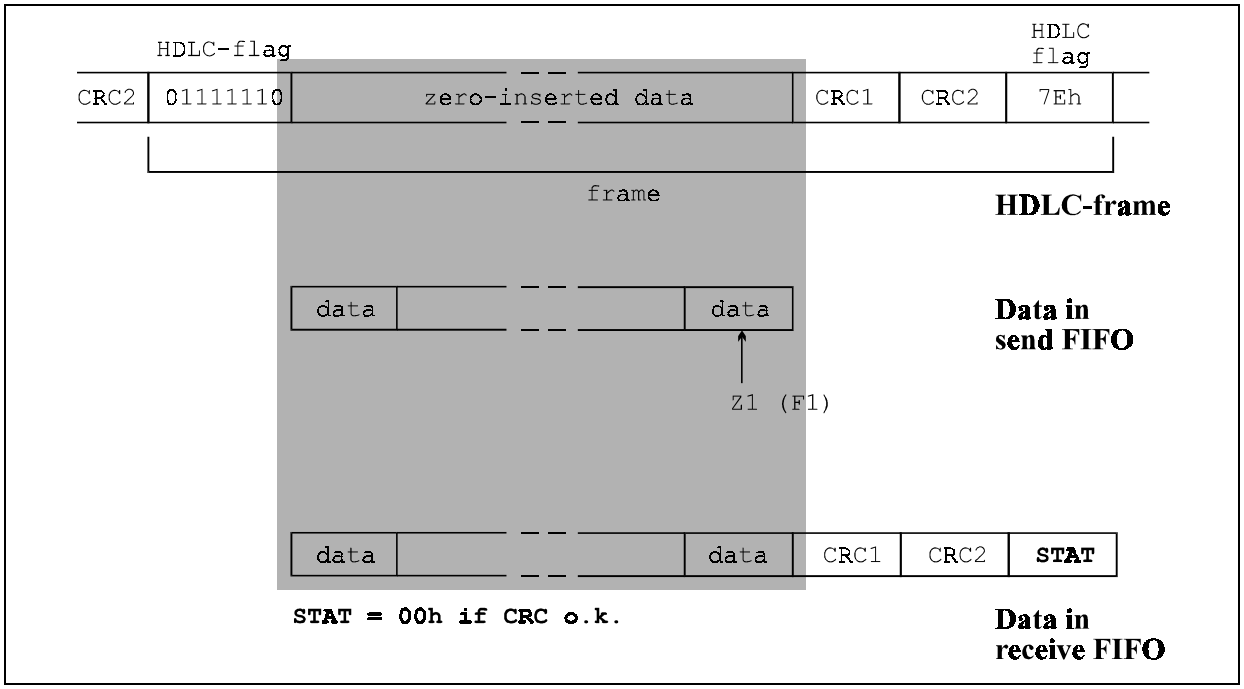

Figure 5: FIFO Data Organisation

The ending flag of a HDLC-frame can also be the starting flag of the next frame.

After a frame is received completely F1 is incremented by the HFC-S2M automatically and the next frame can be received.

After reading a frame via the host bus interface F2 must be incremented. If the frame counter F2 is incremented also the Z-counters may change because Z1 and Z2 are functions of F1 and F2. So there are Z1(F1), Z2(F1), Z1(F2) and Z2(F2) (see Figure 4).
Z1(F1) is used for the frame which is just received from the S2M device side of the HFC. Z2(F2) is used for the frame which is just beeing transmitted to the host bus interface. Z1(F2) is the end of frame pointer of the current output frame.

To calculate the length of the current receive frame the software has to evaluate Z1-Z2+1. When Z2 reaches Z1 the complete frame has been read.

In the receive channels F2 must be incremented from the host interface side after the software detects an end of receive frame (Z1=Z2) and F1≠F2. Then the current value of Z2 is stored, F2 is incremented and Z2 is copied as start address of the next frame. If Z1 = Z2 and F1 = F2 the FIFO is totally empty. Z1(F1) can not be accessed.

☝ **important!**
Before reading a FIFO a change FIFO operation (see also: FIFO# register) must be done even if the desired FIFO is already selected. The change FIFO operation is required to update the internal buffer of the HFC-S2M. Otherwise the first four bytes of the FIFO will be taken from the internal buffer and may be invalid.

2.8.1.5 FIFO full condition in receive channels

Because the ISDN-B-channels and the ISDN-D-channels have no hardware based flow control there is no possibility to stop input data if a receive FIFO is full.

So there is no FIFO full condition implemented in the HFC-S2M. The HFC-S2M assumes that the FIFOs are so deep that the host processor hard- and software is able to avoid any overflow of the receive FIFOs. Overflow conditions are again more than 31 input frames (15 frames for 32k RAM) or a real overflow of the FIFO because of excessive data.

Because HDLC procedures only know a window size of 7 frames no more than 7 frames are sent without software intervention. Due to the great size of the FIFOs of the HFC-S2M it is easy to poll the HFC-S2M even in large time intervalls without having to fear a FIFO overflow condition.

However to avoid any undetected FIFO overflows the software driver should check the number of frames in the FIFO which is F1-F2. An overflow exists if the number (F1-F2) is less than the number in the last reading even if there was no reading of a frame in between.

After a detected FIFO overflow condition this FIFO must be reset by setting the FIFO reset bit in the INC_RES_F register.

2.8.1.6 FIFO reset

All counters Z1, Z2, F1 and F2 of all FIFOs are initialized to all 1s after a RESET.

Then the result is Z1 = Z2 = Z_{MAX} and F1 = F2 = 1Fh (0Fh for 32k RAM).

The same initialisation is done if the bit 3 or bit 4 in the CIRM register is set (soft reset).

Individual FIFOs can be reset by bit 1 of INC_RES_F register.

2.8.2 Transparent mode of HFC-S2M

You can switch off HDLC operation for each B-channel independently by bit 1 in CON_HDLC register. If this bit is set data in the FIFO is sent directly to the S2M or PCM bus interface and data from the S2M or PCM bus interface is sent directly to the FIFO.

Be sure to switch into transparent mode only if F1=F2. Being in transparent mode the Fx counters remain unchanged. Z1 and Z2 are the input and output pointers respectively. Because F1=F2 both Z-counters are always accessable and have valid data.

Because always one Z-counter is changed by the HFC-S2M and only 8 bits of a counter can be read at a time the counter should be read twice to check for a counter incrementation between low and high byte accesses.

If a send FIFO channel changes to FIFO empty condition no CRC is generated and the last data byte written into the FIFO is repeated until there is new data.

In receive channels there is no check on flags or correct CRCs and no status byte is added.

The byte bounderies are not arbitrary like in HDLC mode where byte synchronisation is achieved with HDLC-flags. The data is just the same as it comes from the S2M or PCM bus interface or is sent to this.

Send and receive transparent data can be handled in two ways. The usual way is tranporting B-channel data with the LSB first as it is usual in HDLC mode. The second way is sending the bytes in reverse bit order as it is usual for PWM data. So the first bit is the MSB. The bit order can be reversed by setting bit 7 of the FIFO# register when the FIFO is selected.

2.8.3 FIFO size setup

The HFC-S2M can be operated with 32K x 8, 128k x 8 and 512k x 8 external SRAM. Bits[1..0] of the R_SIZE register must be set accordingly to the RAM size. The following sections show how the FIFO size can be varied for the different RAM sizes.

2.8.3.1 32k x 8 external RAM

Bits[1..0] of the R_SIZE register must be set to 00b if a 32k x 8 RAM is connected to the HFC-S2M. The F-counter range is 00h..0Fh.

F_MODE register		FIFO No.	Z_{MAX}	Z_{MIN}	FIFO size in bytes
bits [1..0]	bits [5..4]				
00	00	63..0	1FFh	80h	384

F_MODE register		FIFO No.	Z_{MAX}	Z_{MIN}	FIFO size in bytes
bits [1..0]	bits [5..4]				
10	00	31..0	0FFh	80h	128
10	00	63..32	1FFh	00h	512
10	01	47..0	0FFh	80h	128
10	01	63..48	3FFh	00h	1024
10	10	55..0	0FFh	80h	128
10	10	63..56	7FFh	00h	2048
10	11	59..0	0FFh	80h	128
10	11	63..60	FFFh	00h	4096

F_MODE register		FIFO No.	Z_{MAX}	Z_{MIN}	FIFO size in bytes
bits [1..0]	bits [5..4]				
11	00	31..0	0FFh	00h	256
11	00	63..32	1FFh	00h	512
11	01	15..0	1FFh	00h	512
11	01	31..16	3FFh	00h	1024
11	10	7..0	3FFh	00h	1024
11	10	15..8	7FFh	00h	2048
11	11	3..0	7FFh	00h	2048
11	11	7..4	FFFh	00h	4096

2.8.3.2 128k x 8 external RAM

Bits[1..0] of the R_SIZE register must be set to 01b if a 128k x 8 RAM is connected to the HFC-S2M. The F-counter range is 00h..1Fh.

F_MODE register		FIFO No.	Z_{MAX}	Z_{MIN}	FIFO size in bytes
bits [1..0]	bits [5..4]				
00	00	63..0	07FFh	C0h	1856

F_MODE register		FIFO No.	Z_{MAX}	Z_{MIN}	FIFO size in bytes
bits [1..0]	bits [5..4]				
10	00	31..0	03FFh	C0h	832
10	00	63..32	07FFh	00h	2048
10	01	47..0	03FFh	C0h	832
10	01	63..48	0FFFh	00h	4096
10	10	55..0	03FFh	C0h	832
10	10	63..56	1FFFh	00h	8192
10	11	59..0	03FFh	C0h	832
10	11	63..60	3FFFh	00h	16384

F_MODE register		FIFO No.	Z_{MAX}	Z_{MIN}	FIFO size in bytes
bits [1..0]	bits [5..4]				
11	00	31..0	03FFh	00h	1024
11	00	63..32	07FFh	00h	2048
11	01	15..0	07FFh	00h	2048
11	01	31..16	0FFFh	00h	4096
11	10	7..0	0FFFh	00h	4096
11	10	15..8	1FFFh	00h	8192
11	11	3..0	1FFFh	00h	8192
11	11	7..4	3FFFh	00h	16384

2.8.3.3 512k x 8 external RAM

Bits[1..0] of the R_SIZE register must be set to 10b if a 512k x 8 RAM is connected to the HFC-S2M. The F-counter range is 00h..1Fh.

F_MODE register		FIFO No.	Z_{MAX}	Z_{MIN}	FIFO size in bytes
bits [1..0]	bits [5..4]				
00	00	63..0	1FFFh	C0h	8000

F_MODE register		FIFO No.	Z_{MAX}	Z_{MIN}	FIFO size in bytes
bits [1..0]	bits [5..4]				
10	00	31..0	0FFFh	C0h	3904
10	00	63..32	1FFFh	00h	8192
10	01	47..0	0FFFh	C0h	3904
10	01	63..48	3FFFh	00h	16384
10	10	55..0	0FFFh	C0h	3904
10	10	63..56	7FFFh	00h	32768
10	11	59..0	0FFFh	C0h	3904
10	11	63..60	FFFFh	00h	65536

F_MODE register		FIFO No.	Z_{MAX}	Z_{MIN}	FIFO size in bytes
bits [1..0]	bits [5..4]				
11	00	31..0	0FFFh	00h	4096
11	00	63..32	1FFFh	00h	8192
11	01	15..0	1FFFh	00h	8192
11	01	31..16	3FFFh	00h	16384
11	10	7..0	3FFFh	00h	16384
11	10	15..8	7FFFh	00h	32768
11	11	3..0	7FFFh	00h	32768
11	11	7..4	FFFFh	00h	65536

2.9 Correspondency between FIFOs, CHANNELs and SLOTs

For the data processing of the HFC-S2M you must distinguish between FIFOs, CHANNELs and SLOTs. The FIFOs are buffers between the universal bus interface and the data interfaces PCM and/or S2M. The HDLC controllers are located on the non host bus side of the FIFOs.
The CHANNELs are either mapped to the data channels on the S2M interface (then the CHANNEL number is the same as the S2M channel number) or they can be connected to arbitrary time slots on the PCM interface. SLOTs are 8 bit time slots on the PCM interface.

The following values (registers) characterise FIFOs, CHANNELs and SLOTs:
FIFO: FIFO#
CHANNEL: CHANNEL#
SLOT: SLOT#

Even numbers (LSB = '0') always belong to a transmit FIFO, transmit CHANNEL or transmit SLOT.
Odd numbers (LSB = '1') always belong to a receive FIFO, receive CHANNEL or receive SLOT.

In Simple Mode (F_MODE register bits 3..2 = '00', SM) the CHANNEL number equals the FIFO number.
In Channel Select Mode (F_MODE register bit 2 = '1', CSM) FIFOs can be associated with arbitrary CHANNELs.
In FIFO Sequence Mode (F_MODE register bit 3 = '1', FSM) FIFOs can be additionally configured to handle more than one CHANNEL.

FIFOs are selected by writing their number in the FIFO# register. All FIFOs are disabled after initialization (reset). By setting at least one of the CON_HDLC register bits 4..1 to '1' the selected FIFO is enabled.

The connection between a FIFO and a CHANNEL can be established by the CHANNEL# register for each FIFO if Channel Select Mode is enabled (F_MODE register bit 2 = '1', CSM). Otherwise the CHANNEL number equals the FIFO number.

The data flow between the HFC part (FIFOs), S2M interface and PCM interface can be selected by the CON_HDLC register (bits 7..5) for each FIFO.

For each PCM SLOT the data source and destination CHANNEL can be selected by the timeslot assigner (SLOT_CFG register). This can be done by writing the desired CHANNEL number to SLOT_CFG register bits 5..0.

Data of a CHANNEL can furthermore be looped over the PCM interface (and the timeslot assigner).

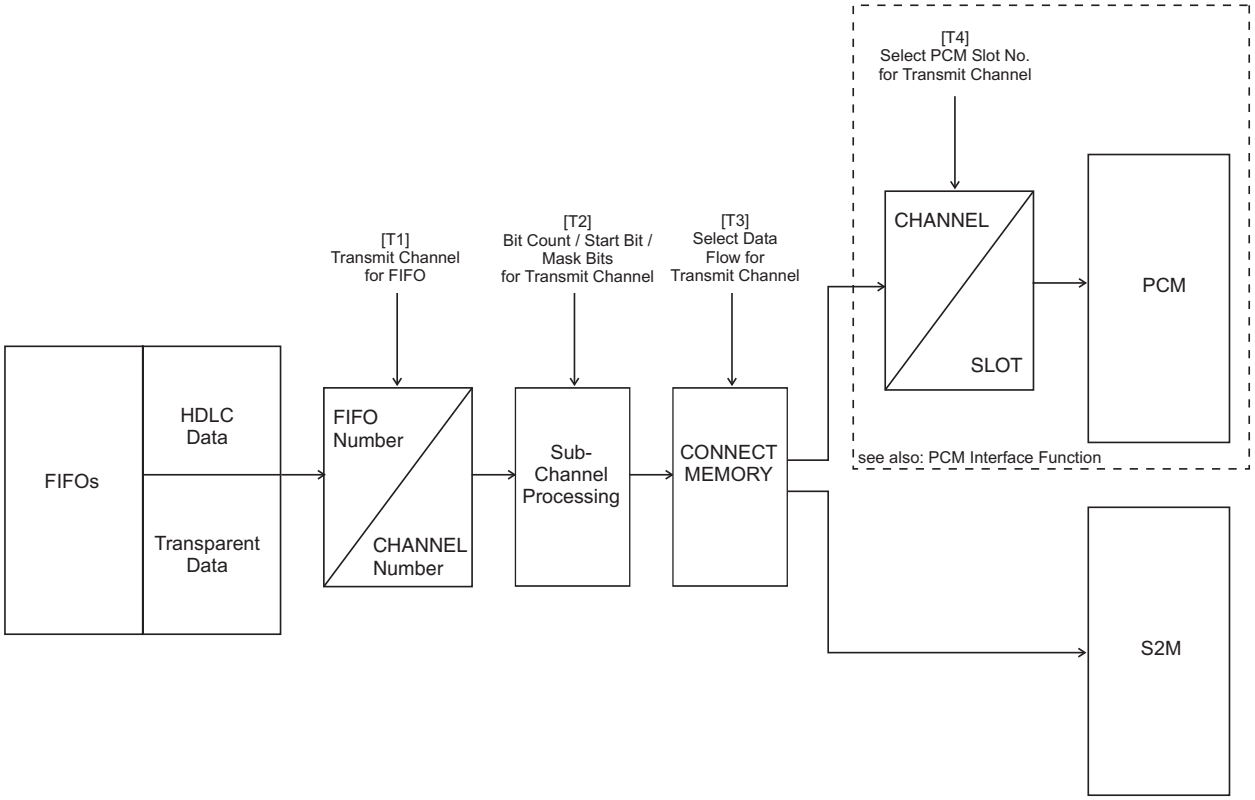

Figure 6: FIFOs, CHANNELs and SLOTs in Transmit Direction

[T1] In Simple Mode (SM) the CHANNEL number is the same as the FIFO number. If Channel Select Mode (CSM) is enabled the transmit CHANNEL for a FIFO can be selected by

1) writing the FIFO number (0..63) in the FIFO# register
2) writing the desired CHANNEL number (0..63) to the CHANNEL# register bits 5..0

Please note that transmit CHANNELs are even numbered (bit 0 of CHANNEL# register = '0').

[T2] The bit values for the not processed bits of the transmit CHANNEL are read from the CH_MASK register. The processed bits are taken from the FIFO (see also: Subchannel Processing). Please note that more than one FIFO can transmit data to the same CHANNEL. This is useful to combine subchannels and transmit them in one ISDN channel.

[T3] Data can either be transmitted to the S2M interface or the PCM interface.

1) write the FIFO number (0..63) in the FIFO# register
2) write the desired connection to the CON_HDLC register bits 7..5

The CON_HDLC register bits 7..5 settings must be the same for corresponding receive and transmit FIFOs.

[T4] A PCM SLOT can be connected to a CHANNEL.
The PCM SLOT number can be selected by
1) writing the desired SLOT number (0..255) to the SLOT# register. Please note that transmit SLOTs are always even numbered (bit 0 of SLOT# register = '0').
2) writing the desired CHANNEL number to the SLOT_CFG register bits 5..0 (bit 0 is for transmit/receive CHANNEL select and must be '0' for transmit CHANNELs)

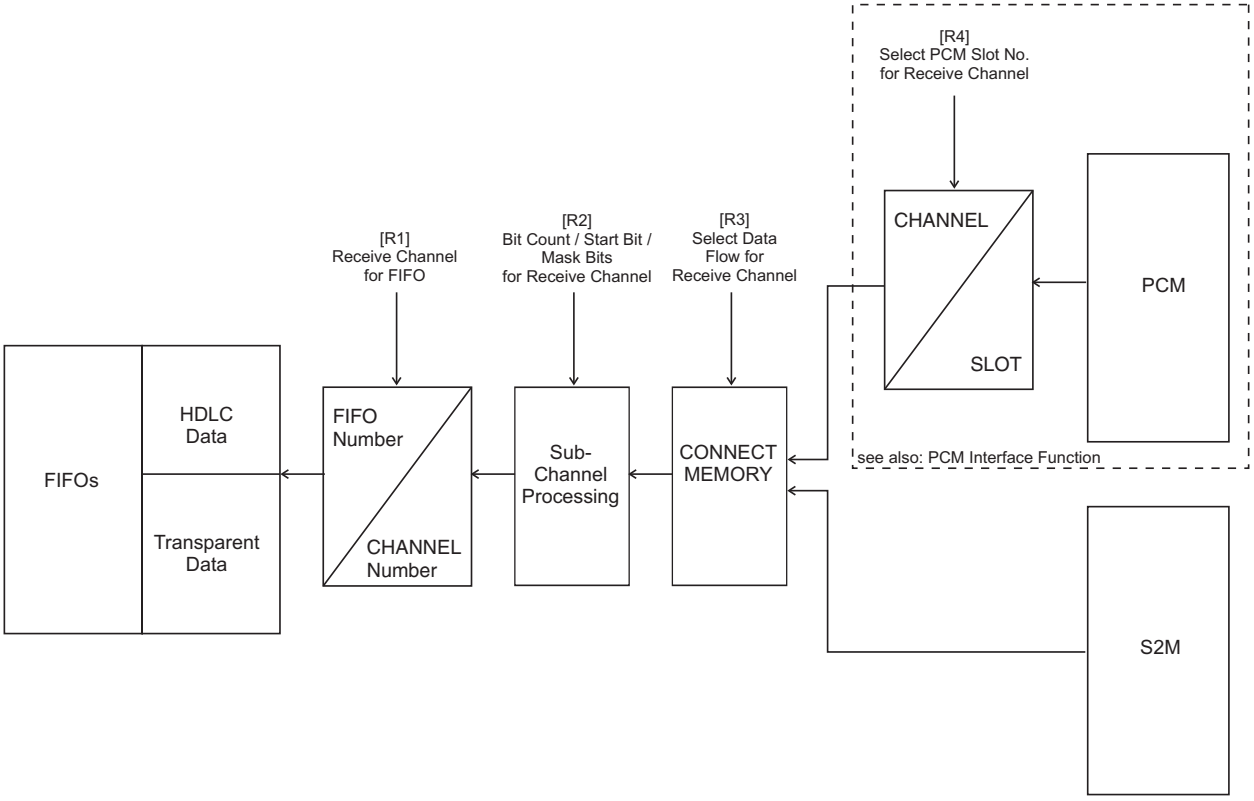

Figure 7: FIFOs, CHANNELs and SLOTs in Receive Direction

[R1] In Simple Mode (SM) the CHANNEL number is the same as the FIFO number. If Channel Select Mode (CSM) is enabled the transmit CHANNEL for a FIFO can be selected by
1) writing the FIFO number (0..63) in the FIFO# register
2) writing the desired CHANNEL number (0..63) to the CHANNEL# register bits 5..0
Please note that receive CHANNELs are odd numbered (bit 0 of CHANNEL# register = '1').

[R2] The bit values of the not processed bits of the receive CHANNEL are ignored. The processed bits are taken from the CHANNEL (see also: Subchannel Processing). Please note that more than one FIFO can receive data from the same CHANNEL (e.g. bits 1..0 are processed by FIFO 1 and bits 3..2 by FIFO 3). This is useful to split subchannels that have been combined to be transmitted in one ISDN channel.

[R3] Data can either be received from the S2M interface or the PCM interface.
1) write the FIFO number (0..63) in the FIFO# register
2) write the desired connection to the CON_HDLC register bits 7..5
The CON_HDLC register bits 7..5 settings must be the same for corresponding receive and transmit FIFOs.

[R4] A PCM SLOT can be connected to a CHANNEL.
The PCM SLOT number can be selected by
1) writing the desired SLOT number (0..255) to the SLOT# register. Please note that transmit SLOTs are always even numbered (bit 0 of SLOT# register = '0').
2) writing the desired CHANNEL number to the SLOT_CFG register bits 5..0 (bit 0 is for transmit/receive CHANNEL select and must be '1' for receive CHANNELs)

2.10 Subchannel Processing

The following example shows how subchannel processing can be configured by the HDLC_PAR register.

Example:

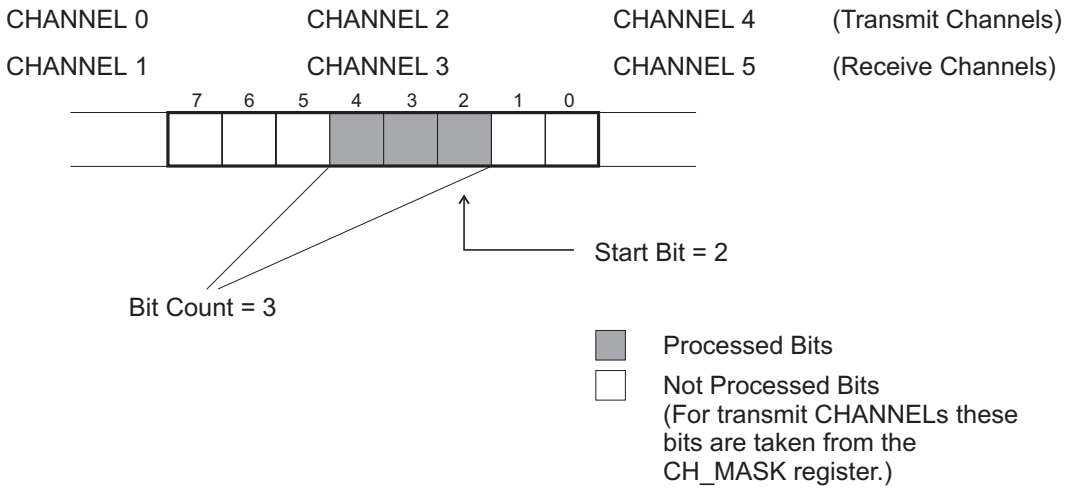

Figure 8: Example for Subchannel Processing

The start bit can be selected by bits 5..3 of the HDLC_PAR register. The number of bits to process can be selected by bits 2..0 of the HDLC_PAR register. By default (HDLC_PAR = 00h) all 8 bits are processed. In the given example the start bit is bit 2 and the number of bits to process is 3. The not processed bits are set to the value given in the CH_MASK register. Please note that the HDLC_PAR register settings can be different for each channel.

2.11 PCM Interface Function

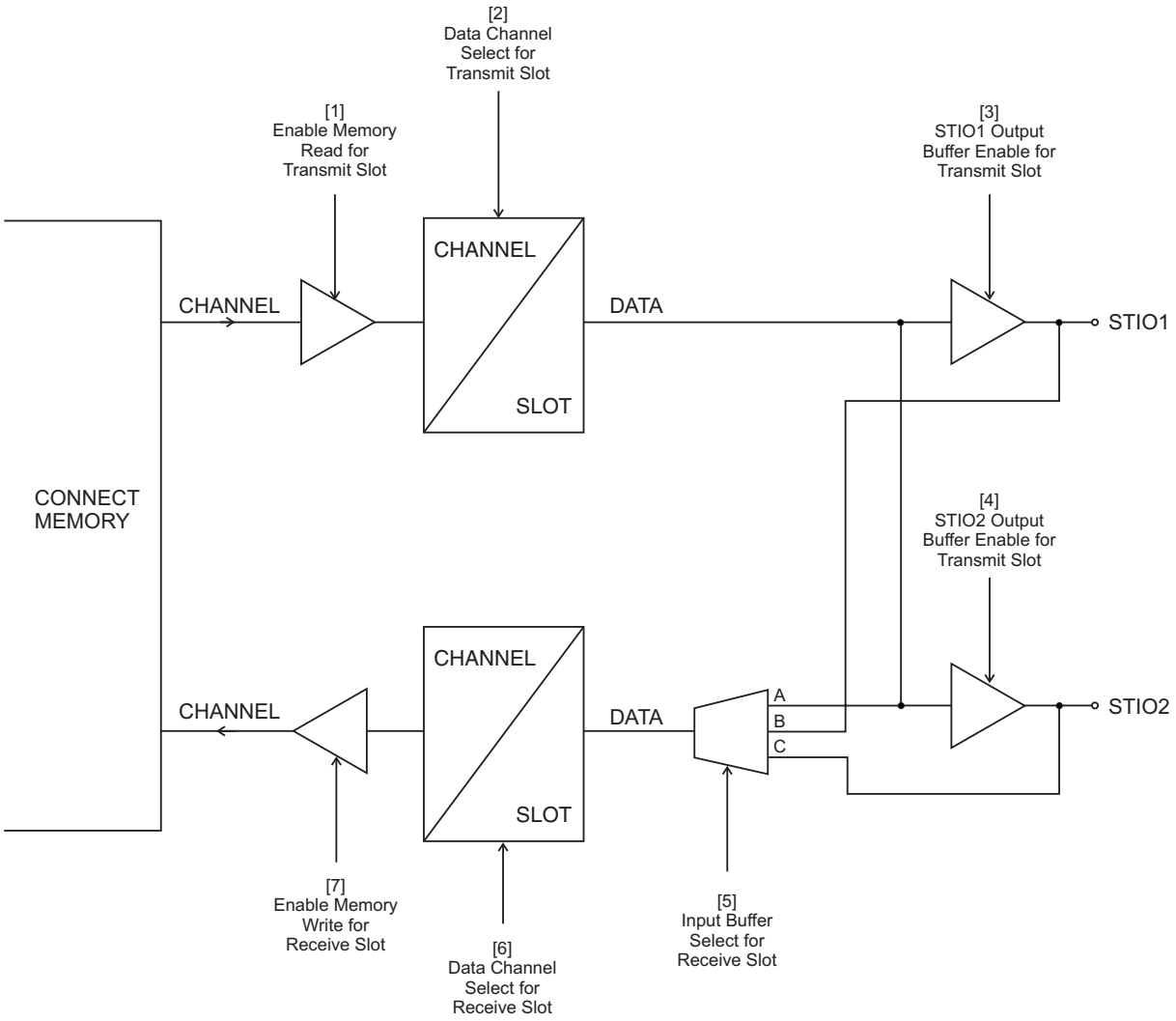

Figure 9: PCM Interface Function Block Diagram

For Even Slot Numbers (Bit 0 of SLOT# Register = '0')		
Number	**Function**	**SLOT_CFG [SLOT#] Register Bits**
[1]	Enable Memory Read for Transmit Slot	Bits[7:6] ≠ '00' (Enable Memory Read)
[2]	Data Channel Select for Transmit Slot	Bits[5:0] are for selection of channel 0..63.
[3]	STIO1 Output Buffer Enable for Transmit Slot	Bits[7:6] = '10' (STIO1 Output Buffer Enable)
[4]	STIO2 Output Buffer Enable for Transmit Slot	Bits[7:6] = '11' (STIO2 Output Buffer Enable)

For Odd Slot Numbers (Bit 0 of SLOT# Register = '1')		
Number	**Function**	**SLOT_CFG [SLOT#] Register Bits**
[5]	Input Buffer Select for Receive Slot	Bits[7:6] = '01' (Loop PCM Internally [MUX Input A]) Bits[7:6] = '10' (Data In From STIO2 [MUX Input C]) Bits[7:6] = '11' (Data In From STIO1 [MUX Input B])
[6]	Data Channel Select for Receive Slot	Bits[5:0] are for selection of channel 0..63.
[7]	Enable Memory Write for Receive Slot	Bits[7:6] ≠ '00' (Enable Memory Write)

2.12 Multiparty audio conferences

The HFC-S2M is able to handle up to 7 different multiparty audio conferences. The number of PCM time slots that can be combined to one conference is only limited by the number of channels (64 / 2 = 32). Each time slot can only be part of one conference.
To add a PCM time slot to a conference the slot number must be written to the SLOT# register. If the time slot has not yet been linked to a data channel this can be done by writing the data channel number and the channels source/destination (input/output pins) to the SLOT_CFG register. For a conference either STIO1 or STIO2 should be used as inputs for receive channels. Afterwards the conference number must be written to the CONFER register. Noise suppression threshold and input attenuation level can be configured independently for each time slot.
To remove a time slot from a conference the time slot must be selected by writing its number to the SLOT# register. Then 00h must be written to the CONFER register.

Example:

// Add PCM time slot #16 to audio conference #1

- write 16 to register SLOT# (10h) // select time slot #16
- write 90h to register SLOT_CFG (D0h) // link slot #16 to data channel 16, data input pin // is STIO2
- write 01h to CONFER (D1h) // add selected time slot (#16) to conference #1, // no noise suppression, input attenuation level // is 0 dB

// Remove PCM time slot #16 from audio conference #1

- write 16 to register SLOT# (10h) // select time slot #16
- write 00h to CONFER (D1h) // write a zero as conference number to remove // the selected time slot (#16) from conference

2.13 Resets

The HFC-S2M has 4 different software resets. The FIFO registers, PCM registers and S2M registers can be resetted independently by bits 4..6 of the CIRM register. The reset bit(s) must be cleared by software.

Reset Name	Reset Number	Description
hard reset	0	hardware reset (RESET pin activated)
HFC-reset	1	Reset for all FIFO registers of the HFC-S2M.
PCM-reset	2	Reset for all PCM registers of the HFC-S2M.
S2M reset	3	Reset for all S2M registers of the HFC-S2M.
soft reset	4	Reset for FIFO, PCM and S2M registers of the HFC-S2M. Soft reset is the same as reset 1+2+3.

Table 4: Resets of the HFC-S2M

Information about which registers are resetted by the different resets can be found in the register list table on page 28.

2.14 EEPROM

The external EEPROM is required for PCMCIA mode and ISA-PnP mode and optional for PCI mode and processor mode. It is used to store the configuration data for PCMCIA, PCI or ISA-PnP. After RESET (hard reset or EE Reload (bit 7 of CIRM register)) the HFC-S2M copies a constant number of bytes from the EEPROM to the external SRAM. The bytes which are not used by the configuration data can be filled with vendor defined data. This data (and the configuration data as well) can be read by RAM accesses to the HFC-S2M. The table below shows how many bytes are copied in the different modes. The address where the configuration data starts is given in the column "Starting Address in SRAM".

Mode	Number of Bytes Copied	Starting Address in SRAM
ISA-PnP mode	512	1A00h with 32K SRAM 2A00h with 128K and 512K SRAM
PCMCIA mode	512	
PCI mode	128	
processor mode	512	

2.15 RAM accesses

The external SRAM of the HFC-S2M can be accessed by the host. The desired RAM address must be written in the RAM_ADR_x registers first. Data can then be read/written by reading/writing the RAM_DATA register.

3 Register bit description

3.1 Register bit description of interrupt, status and control registers

Name	Addr.	Bits	r/w	Function
CIRM	00h	2..0	w	unused, must be '0'
		3	w	soft reset, similar as hardware reset; the selected I/O address is kept in ISA PnP mode. The reset is active until the bit is cleared. '0' deactivate reset (reset default) '1' activate reset
		4	w	HFC-reset Sets all FIFO registers to their initial values. The reset is active until the bit is cleared. '0' deactivate reset (reset default) '1' activate reset
		5	w	PCM-reset Sets all PCM registers to their initial values. The reset is active until the bit is cleared. '0' deactivate reset (reset default) '1' activate reset
		6	w	S2M-reset Sets all S2M registers to their initial values. The reset is active until the bit is cleared. '0' deactivate reset (reset default) '1' activate reset
		7	w	EE Reload '1' reload EEPROM to SRAM This bit must be cleared by software.
X_ACC_EN	01h	0	w	external access enable '0' disable (reset default) '1' enable
		1	w	RAM priority for data access '0' normal priority (reset default) '1' low priority
		7..2		unused

Name	Addr.	Bits	r/w	Function
INT_CTRL	13h	0	w	'0' FIFO interrupts disable (reset default) '1' FIFO interrupts enable
		2..1	w	unused, must be '0'
		3	w	global interrupt signal enable '0' disable (reset default) '1' enable
		4	w	polarity of interrupt signal '0' low active signal '1' high active signal
		7..5	w	unused, must be '0'
INT_OVIEW	10h	7..0	r	Indicates in which FIFO block of 8 FIFOs an interrupt has occured. The exact FIFO can be determined by reading the INT_FIF_x register that belongs to the specified FIFO block. '0000 0001' interrupt has occured in FIFO 0..7 '0000 0010' interrupt has occured in FIFO 8..15 '0000 0100' interrupt has occured in FIFO 16..23 '0000 1000' interrupt has occured in FIFO 24..31 '0001 0000' interrupt has occured in FIFO 32..39 '0010 0000' interrupt has occured in FIFO 40..47 '0100 0000' interrupt has occured in FIFO 48..55 '1000 0000' interrupt has occured in FIFO 56..63 Reading any INT_FIF_x register clears these bits.
INT_MISC	11h	0	r	'1' state of state machine has changed
		1	r	timer interrupt '1' timer elapsed
		2	r	processing/non processing transition interrupt status '1' The HFC-S2M has changed from processing to non processing state (every 125us, F0=8kHz).
		3	r	DTMF detection interrupt '1' DTMF detection has been finished. The results can be read from the DTMF registers.
		4	r	1 S INT '1' 1 S INT
		7..5	r	unused, '0'
INT_MISC_M	11h	0	w	state of state machine changed interrupt mask bit
		1	w	timer elapsed interrupt mask bit
		2	w	processing/nonprocessing transition interrupt mask bit (every 125u, F0=8khz)
		3	w	DTMF detection interrupt mask bit
		4	w	1 S INT mask bit
		7..5	w	unused, must be '0'

Name	Addr.	Bits	r/w	Function
INT_FIF_0	C8h	7..0	r	Specifies the number(s) of the FIFO(s) in which an interrupt has occured. 00h no interrupt in FIFO 0..7 01h interrupt occured in FIFO 0 : : 80h interrupt occured in FIFO 7 Reading this register clears the bits.
INT_FIF_1	C9h	7..0	r	00h no interrupt in FIFO 8..15 01h interrupt occured in FIFO 8 : : 80h interrupt occured in FIFO 15 Reading this register clears the bits.
INT_FIF_2	CAh	7..0	r	00h no interrupt in FIFO 16..23 01h interrupt occured in FIFO 16 : : 80h interrupt occured in FIFO 23 Reading this register clears the bits.
INT_FIF_3	CBh	7..0	r	00h no interrupt in FIFO 24..31 01h interrupt occured in FIFO 24 : : 80h interrupt occured in FIFO 31 Reading this register clears the bits.
INT_FIF_4	CCh	7..0	r	00h no interrupt in FIFO 32..39 01h interrupt occured in FIFO 32 : : 80h interrupt occured in FIFO 39 Reading this register clears the bits.
INT_FIF_5	CDh	7..0	r	00h no interrupt in FIFO 40..47 01h interrupt occured in FIFO 40 : : 80h interrupt occured in FIFO 47 Reading this register clears the bits.
INT_FIF_6	CEh	7..0	r	00h no interrupt in FIFO 48..55 01h interrupt occured in FIFO 48 : : 80h interrupt occured in FIFO 55 Reading this register clears the bits.
INT_FIF_7	CFh	7..0	r	00h no interrupt in FIFO 56..63 01h interrupt occured in FIFO 56 : : 80h interrupt occured in FIFO 63 Reading this register clears the bits.

Name	Addr.	Bits	r/w	Function
R_SIZE	0Ch	1..0	w	RAM size '00' 32k x 8 '01' 128k x 8 '10' 512k x 8
		7..2		unused
F_MODE	0Dh	1..0	w	FIFO mode (see also: FIFO size setup on page 37)
		3..2	w	'00' Simple Mode enable (SM) (FIFO number = CHANNEL number, reset default) '01' Channel Select Mode enable (CSM) '10' FIFO Sequence Mode enable (FSM) '11' CSM + FSM enable
		5..4	w	FIFO size (see also: FIFO size setup on page 37)
		7..6	w	unused
INC_RES_F	0Eh	0	w	increment F-counter of selected FIFO ('1'=increment)
		1	w	reset selected FIFO ('1'=reset FIFO)
		2	w	reset LOST ('1'=reset LOST)
		7..3	w	unused, should be '0'
RAM_ADR_0	08h	7..0	w	Address bits 7..0 for RAM accesses
RAM_ADR_1	09h	7..0	w	Address bits 15..8 for RAM accesses
RAM_ADR_2	0Ah	7..0	w	Address bits 23..16 for RAM accesses
RAM_DATA	C0h	7..0	r	SRAM data
FIFO#	0Fh	5..0	w	select FIFO#
		6	w	unused, should be '0'
		7	w	bit order '0' normal bit order (reset default) '1' reverse bit order

Name	Addr.	Bits	r/w	Function
FIF_DATA [FIFO#]	80h	7..0	r/w	FIFO data register (access 1 byte) read/write one byte data from/to the FIFO selected in the FIFO# register and increment Z-counter
	81h	7..0	r/w	FIFO data register (access 2 bytes) read/write two bytes data from/to the FIFO selected in the FIFO# register and increment Z-counter
	82h	7..0	r/w	FIFO data register (access 3 bytes) read/write three bytes data from/to the FIFO selected in the FIFO# register and increment Z-counter
	83h	7..0	r/w	FIFO data register (access 4 bytes) read/write four bytes data from/to the FIFO selected in the FIFO# register and increment Z-counter
	84h	7..0	r/w	FIFO data register (alternate, access 1 byte) read/write one byte data from/to the FIFO selected in the FIFO# register without incrementing Z-counter
	85h	7..0	r/w	FIFO data register (alternate, access 2 bytes) read/write two bytes data from/to the FIFO selected in the FIFO# register without incrementing Z-counter
	86h	7..0	r/w	FIFO data register (alternate, access 3 bytes) read/write three bytes data from/to the FIFO selected in the FIFO# register without incrementing Z-counter
	87h	7..0	r/w	FIFO data register (alternate, access 4 bytes) read/write three bytes data from/to the FIFO selected in the FIFO# register without incrementing Z-counter
FIF_F1 [FIFO#]	0Ch	7..0	r	FIFO input HDLC frame counter (F1) Up to 31 HDLC frames (15 with 32k RAM) can be stored in each FIFO.
FIF_F2 [FIFO#]	0Dh	7..0	r	FIFO output HDLC frame counter (F2) Up to 31 HDLC frames (15 with 32k RAM) can be stored in each FIFO.

Name	Addr.	Bits	r/w	Function
FIF_Z1L [FIFO#]	04h	7..0	r	FIFO input counter Z1 (low byte) The min/max values of the Z-counters can be found in the chapter FIFO size setup on page 37.
FIF_Z1H [FIFO#]	05h	7..0	r	FIFO input counter Z1 (high byte) The min/max values of the Z-counters can be found in the chapter FIFO size setup on page 37.
FIF_Z2L [FIFO#]	06h	7..0	r	FIFO output counter Z2 (low byte) The min/max values of the Z-counters can be found in the chapter FIFO size setup on page 37.
FIF_Z2H [FIFO#]	07h	7..0	r	FIFO output counter Z2 (high byte) The min/max values of the Z-counters can be found in the chapter FIFO size setup on page 37.

Name	Addr.	Bits	r/w	Function
CON_HDLC [FIFO#]	FAh	0	w	inter frame fill '0' write HDLC flags as inter frame fill (reset default) '1' write all '1's as inter frame fill
		1	w	HDLC mode/transparent mode select '0' HDLC mode (reset default) '1' transparent mode select **if bits are '0000' the FIFO is disabled to reduce RAM accesses** (reset default)
		4..2	w	transparent mode interrupt select
		7..5	w	select B-channel (D-channel) data flow for selected FIFO destination ← source bit 5: '0' B-HFC ← B-S2M '1' B-HFC ← B-PCM bit 6: '0' B-S2M ← B-HFC '1' B-S2M ← B-PCM bit 7: '0' B-PCM ← B-HFC '1' B-PCM ← B-S2M **CON_HDLC register bits[7:5] must be the same for corresponding receive and transmit FIFOs.**

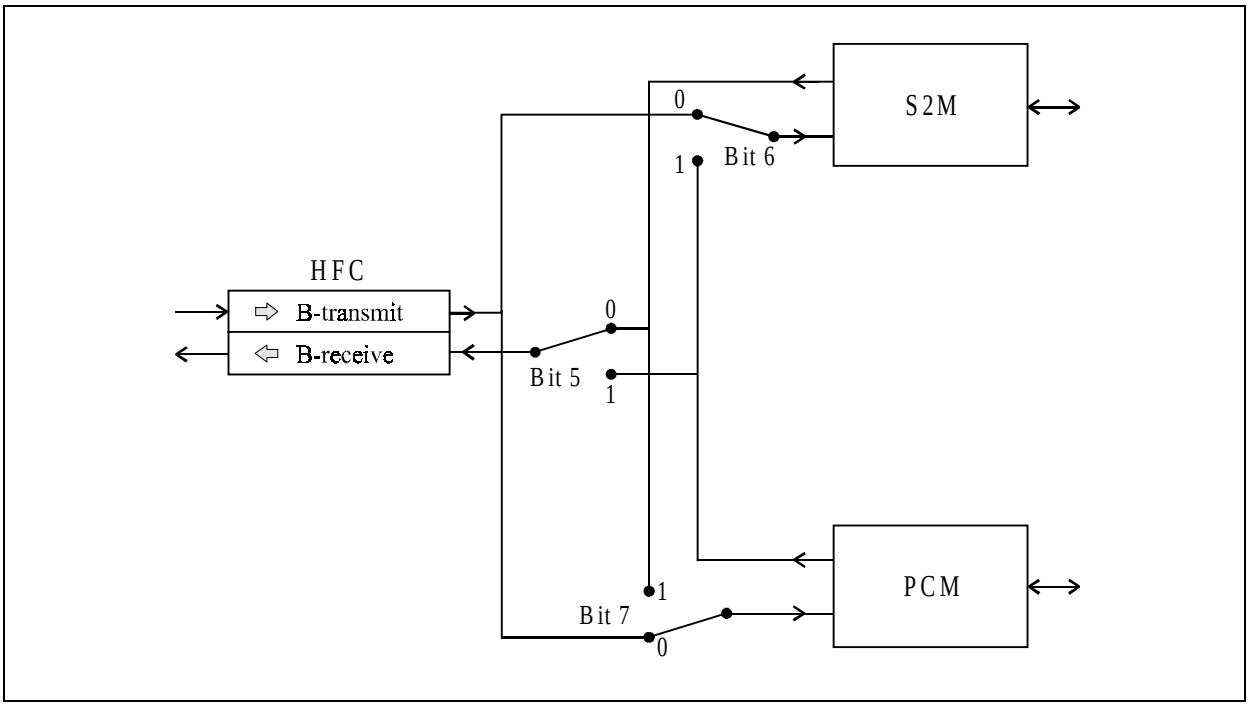

Figure 10: Function of CON_HDLC register bits 7..5

Name	Addr.	Bits	r/w	Function
HDLC_PAR [FIFO#]	FBh	2..0	w	bit count for HDLC and transparent mode (number of bits to process) 000b process 8 bits (64kbit/s) (reset default) 001b process 1 bit : : 111b process 7 bits (56kbit/s)
		5..3	w	start bit for HDLC and transparent mode 000b start processing with bit 0 (reset default) : : 111b start processing with bit 7
		6	w	FIFO loop '0' normal operation (reset default) '1' repeat current frame
		7	w	invert data enable/disable '0' normal read/write data (reset default) '1' invert data

☝ **important!**
For B-channels the HDLC_PAR register must be set to 00h. To use 56kbit/s restricted mode the HDLC_PAR register must be set to 07h for B-channels.

Name	Addr.	Bits	r/w	Function
CHANNEL# [FIFO#]	FCh	5..0	w	link selected FIFO to ISDN channel 0..63 This register is only active in Channel Select Mode (CSM) (see also: F_MODE register)
		7..6	w	unused, should be '0'
F_INIT	0Bh	5..0	w	number of first FIFO sequence. Only used in FIFO sequence mode (FSM) (see also register F_MODE).
		7..6		unused
F_SEQ [FIFO#]	FDh	5..0	w	next FIFO in FIFO sequence
		6	w	unused, should be '0'
		7	w	set as end of chain
CH_MASK [FIFO#]	F4h	7..0	w	Bit value for not processed bits of a channel. All not processed bits of a channel are set to the value defined in this register.
INT_MASK [FIFO#]	FFh	0	w	interrupt mask for selected FIFO
		7..1	w	unused

Name	Addr.	Bits	r/w	Function
TIMER_WD	1Ah	3..0	w	timer event on 2^n * 125 µs 0h 125µs 1h 250 µs : : Eh 2 s Fh 4 s
		7..4	w	watchdog event on 2^n * 16 * 125 µs 0h 2 ms 1h 4 ms : : Fh 65.5 s
WD_MODE	1Bh	4..0	w	unused, must be '0'
		5	w	'1' watchdog is resetted after every access to the chip
		6	w	unused, must be '0'
		7	w	'1' watchdog reset This bit is automatically cleared.

Name	Addr.	Bits	r/w	Function
CHIP_ID	16h	3..0	r	**ISA-PnP mode:** IRQ assigned by the PnP BIOS Bits [2:0] of the CIRM register must be set to the value corresponding to the hardware connected IRQ lines. **PCI mode:** characteristics of PCI buffers '0000' 5V PCI signaling environment '0001' 3.3V PCI signaling environment
		7..4	r	Chip identification '1110' HFC-S2M
STATUS	1Ch	0	r	BUSY/NOBUSY status '1' the HFC-S2M is BUSY after initialising Reset FIFO, increment F or change FIFO '0' the HFC-S2M is not busy, all accesses are allowed
		1	r	processing/non processing status '1' the HFC-S2M is in processing phase (every 125µs) '0' the HFC-S2M is not in processing phase
		2	r	unused, '0'
		3	r	LOST interrupt (frames have been lost) This means the HFC-S2M did not process all data in 125µs. So data may be corrupted. Bit 2 of the INC_RES_F register must be set to recover from this situation.
		4	r	timer status '0' timer not elapsed '1' timer elapsed
		5	r	TE/NT state machine interrupt state '1' state of state machine has changed
		6	r	any miscellaneous interrupt all masked miscellaneous interrupts are "ored"
		7	r	any FIFO frame interrupt all masked FIFO frame interrupts are "ored"

Reading the STATUS register clears no bit.

3.2 Register bit description of PCM bus section

Name	Addr.	Bits	r/w	Function
SLOT#	10h	0	w	select transmit/receive slot '0' transmit slot '1' receive slot
		7..1	w	Select of slot-no. on PCM bus Use selected slot number for all slot number depending registers. Depending on the PCM data rate settings in the MST_MODE1 register 32, 64 or 128 slots are available.
SLOT_CFG [SLOT#]	D0h	0	w	transmit/receive data channel select '0' transmit data channel '1' receive data channel
		5..1	w	data channel select
		7..6	w	**for receive slots:** '00' input data is ignored '01' loop MST internally '10' data in from STIO2 '11' data in from STIO1 **for transmit slots:** '00' disable output buffers '01' transmit data internally, output buffers disabled '10' output buffer enable for STIO1 '11' output buffer enable for STIO2 See also PCM Interface Function on page 44.
CONFER [SLOT#]	D1h	2..0	w	conference number (0..7) (see also: Multiparty audio conferences on page 46)
		4..3	w	noise suppression threshold '00' no noise suppression '01' fifth step, first segment '10' ninth step, first segment '11' sixteenth step, first segment
		6..5	w	input attenuation level '00' 0 dB '01' 3 dB '10' 6 dB '11' 9 dB
		7		unused, must be '0'

Name	Addr.	Bits	r/w	Function
CONF_EN	18h	0	w	conference enable '0' disable (reset default) '1' enable
		6..1	w	unused, should be '0'
		7	w	'0' A-Law '1' µ-Law
MST_MODE0	14h	0	w	PCM bus mode '0' slave (reset default) (C4IO and F0IO are inputs) '1' master (C4IO and F0IO are outputs)
		1	w	polarity of C4- and C20-clock '0' F0IO is sampled on negative clock transition '1' F0IO is sampled on positive clock transition
		2	w	polarity of F0-signal '0' F0 positive pulse '1' F0 negative pulse
		3	w	duration of F0-signal '0' F0 active for one C4-clock (244ns) (reset default) '1' F0 active for two C4-clocks (488ns)
		7..4	w	MST_ADR select register accessible at address 15h '0000' 0h SLOT_SEL0 register accessible at 15h '0001' 1h SLOT_SEL1 register accessible at 15h '0010' 2h SLOT_SEL2 register accessible at 15h '0011' 3h SLOT_SEL3 register accessible at 15h '0100' 4h SLOT_SEL4 register accessible at 15h '0101' 5h SLOT_SEL5 register accessible at 15h '0110' 6h SLOT_SEL6 register accessible at 15h '0111' 7h SLOT_SEL7 register accessible at 15h '1001' 9h MST_MODE1 register accessible at 15h '1010' Ah MST_MODE2 register accessible at 15h '1100' Ch SHAPE0_L register accessible at 15h '1101' Dh SHAPE0_H register accessible at 15h '1110' Eh SHAPE1_L register accessible at 15h '1111' Fh SHAPE1_H register accessible at 15h
SLOT_SEL0	15h	MST_MODE0 bits [7:4] = '0000' (0h)		
		6..0	w	Slot number -1 for F_Q0 e.g. 00h means slot 1
		7	w	Shape select '0' use shape from SHAPE0 registers '1' use shape from SHAPE1 registers
		By setting all bits to '1' the slot is disabled.		

Name	Addr.	Bits	r/w	Function
SLOT_SEL1	15h	MST_MODE0 bits [7:4] = '0001' (1h)		
		6..0	w	Slot number -1 for F_Q1 e.g. 00h means slot 1
		7	w	Shape select '0' use shape from SHAPE0 registers '1' use shape from SHAPE1 registers
		By setting all bits to '1' the slot is disabled.		
SLOT_SEL2	15h	MST_MODE0 bits [7:4] = '0010' (2h)		
		6..0	w	Slot number -1 for F_Q2 e.g. 00h means slot 1
		7	w	Shape select '0' use shape from SHAPE0 registers '1' use shape from SHAPE1 registers
		By setting all bits to '1' the slot is disabled.		
SLOT_SEL3	15h	MST_MODE0 bits [7:4] = '0011' (3h)		
		6..0	w	Slot number -1 for F_Q3 e.g. 00h means slot 1
		7	w	Shape select '0' use shape from SHAPE0 registers '1' use shape from SHAPE1 registers
		By setting all bits to '1' the slot is disabled.		
SLOT_SEL4	15h	MST_MODE0 bits [7:4] = '0100' (4h)		
		6..0	w	Slot number -1 for F_Q4 e.g. 00h means slot 1
		7	w	Shape select '0' use shape from SHAPE0 registers '1' use shape from SHAPE1 registers
		By setting all bits to '1' the slot is disabled.		
SLOT_SEL5	15h	MST_MODE0 bits [7:4] = '0101' (5h)		
		6..0	w	Slot number -1 for F_Q5 e.g. 00h means slot 1
		7	w	Shape select '0' use shape from SHAPE0 registers '1' use shape from SHAPE1 registers
		By setting all bits to '1' the slot is disabled.		
SLOT_SEL6	15h	MST_MODE0 bits [7:4] = '0110' (6h)		
		6..0	w	Slot number -1 for F_Q6 e.g. 00h means slot 1
		7	w	Shape select '0' use shape from SHAPE0 registers '1' use shape from SHAPE1 registers
		By setting all bits to '1' the slot is disabled.		

Name	Addr.	Bits	r/w	Function
SLOT_SEL7	15h	MST_MODE0 bits [7:4] = '0111' (7h)		
		6..0	w	Slot number -1 for F_Q7 e.g. 00h means slot 1
		7	w	Shape select '0' use shape from SHAPE0 registers '1' use shape from SHAPE1 registers
		By setting all bits to '1' the slot is disabled.		

☝ **important!**
The value that must be written to the SLOT_SEL registers to select slot 0 is depending on the PCM data rate:
PCM30: 1Fh must be written to the desired SLOT_SEL register to select slot 0
PCM64: 3Fh must be written to the desired SLOT_SEL register to select slot 0
PCM128: 7Fh must be written to the desired SLOT_SEL register to select slot 0

Please note that slot 0 for PCM128 can only be used with SHAPE0.

Name	Addr.	Bits	r/w	Function
MST_MODE1	15h	MST_MODE0 bits [7:4] = '1001' (9h)		
		0	w	'0' CODEC enable signals on F_Q[7:0] '1' SHAPE1/2 signals on F1_0, F1_1 F_Q[7:0] are counter outputs for up to 128 CODECs.
		1	w	unused, must be '0'
		3..2	w	DPLL adjust speed '00' C4IO clock is adjusted in the last time slot of MST frame 4 times by one half clock cycle '01' C4IO clock is adjusted in the last time slot of MST frame 3 times by one half clock cycle '10' C4IO clock is adjusted in the last time slot of MST frame twice by one half clock cycle '11' C4IO clock is adjusted in the last time slot of MST frame once by one half clock cycle
		5..4	w	PCM data rate '00' 2MBit/s '01' 4MBit/s '10' 8Mbit/s '11' unused
		6	w	MST test loop When set MST output data is looped to the MST inputs.
		7	w	unused, must be '0'
MST_MODE2	15h	MST_MODE0 bits [7:4] = '1010' (Ah)		
		1..0	w	unused, must be '0'
		2	w	select PCM DPLL sync source '0' S/T receive frame (only in TE mode and state F7) '1' SYNC_I input 8 kHz
		3	w	select SYNC_O output '0' S/T receive frame 8 kHz (only in TE mode and state F7) '1' SYNC_I is connected to SYNC_O
		5..4	w	unused, must be '0'
		6	w	This bit is only valid if bit 7 is set. '0' PCM frame time is reduced as selected by bits 3..2 of the MST_MODE1 register '1' PCM frame time is increased as selected by bits 3..2 of the MST_MODE1 register
		7	w	'0' normal operation '1' enable PCM PLL adjust if no sync source is available

Name	Addr.	Bits	r/w	Function
SHAPE0_L	15h	MST_MODE0 bits [7:4] = '1100' (Ch)		
		7..0	w	CODEC enable signal SHAPE 0 (low byte) Bit 0 defines the level for the rest of the period.
SHAPE0_H	15h	MST_MODE0 bits [7:4] = '1101' (Dh)		
		7..0	w	CODEC enable signal SHAPE 0 (high byte)
SHAPE1_L	15h	MST_MODE0 bits [7:4] = '1110' (Eh)		
		7..0	w	CODEC enable signal SHAPE 1 (low byte) Bit 0 defines the level for the rest of the period.
SHAPE1_H	15h	MST_MODE0 bits [7:4] = '1111' (Fh)		
		7..0	w	CODEC enable signal SHAPE 1 (high byte)

The following figure shows how the CODEC enable signal shapes can be configured.

In the example F1_0 uses SHAPE 0 (SHAPE0_L and SHAPE0_H register) and time slot 0, F1_1 uses SHAPE 1 (SHAPE1_L and SHAPE1_H register) and time slot 1.
The register values are as follows:

SHAPE0_L F8h (bit 0 defines the level for the rest of the period)
SHAPE0_H 03h

SHAPE1_L 1Fh (bit 0 defines the level for the rest of the period)
SHAPE1_H F0h

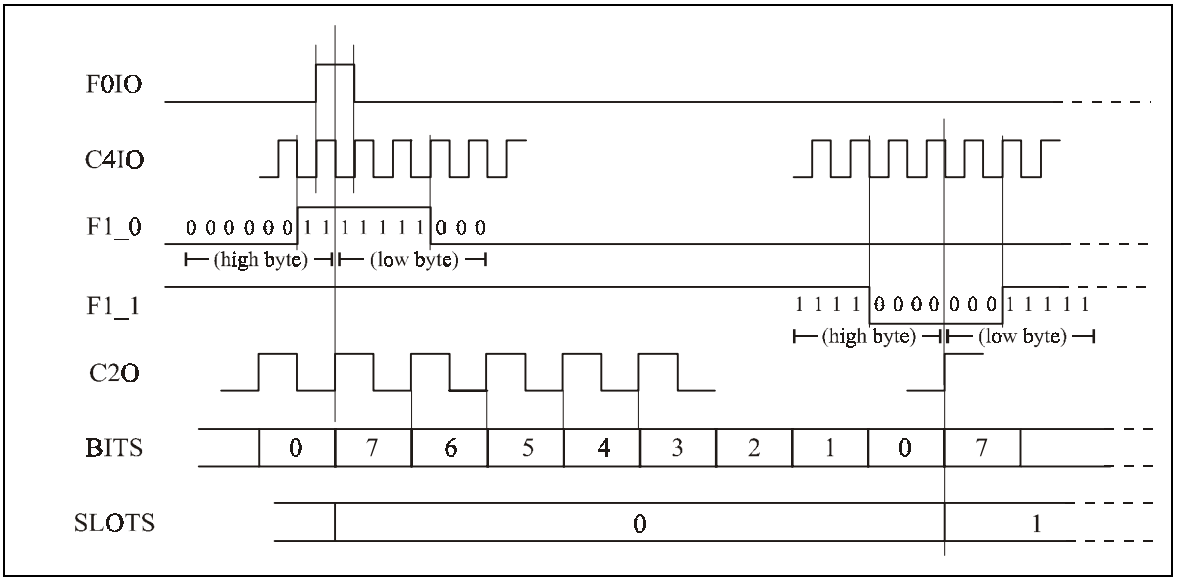

Figure 11: Configuring CODEC enable signal shapes

Name	Addr.	Bits	r/w	Function
F0_CNT_L	18h	7..0	r	F0IO pulse count 16 bit 125µs time counter (low byte) This register should be read first to "lock" the value of the F0_CNT_H register until F0_CNT_H has also been read.
F0_CNT_H	19h	7..0	r	F0IO pulse count 16 bit 125µs time counter (high byte)

3.3 Register bit description of S2M section

Name	Addr.	Bits	r/w	Function
LOS0	22h	7..0	w	LOS alarm will be active if the incoming data stream has no transitions in (LOS 0+1)*16 consecutive pulse. Maximum time is: 256 * 16 * 488ns = 2ms.
LOS1	23h	7..0	w	LOS alarm will be cleared if the incoming data stream has LOS1+1 transitions in LOS0 time interval. After LOS alarm is cleared a new LOS0 time interval will be started.
RECEIVE0	24h	1..0	w	Receive Code '00' NRZ '10' AMI Code 'x1' HDB3 Code
		2	w	FULLHALF '0' half bauded (normal operation) '1' full bauded
		3	w	CMI code '0' CMI off '1' CMI on
		4	w	inverted CMI code, only valid if CMI is on '0' CMI code '1' inverted CMI code
		5	w	Polarity of Clock, if Data Clock Input is used '0' Clock not inverted '1' Clock inverted
		6	w	Polarity of Data Input '0' Data not inverted '1' Data inverted
		7	w	AIS alarm '0' according to ETS 300233 '1' according to ITU-T G.775
REC_STAT0	24h	1..0	r	'00' not synchronized '01' FAS found '10' NFAS found after FAS '11' synchronized (FAS - NFAS - FAS found)
		2	r	frame sync. (FAS - NFAS - FAS found)
		3	r	LOS - Loss of Receive Signal detected.
		5..4	r	counter for MFA (multi frame alignment) '01' MFA pattern found '10' MFA reached (2 consecutive MFA patterns found)
		6	r	AIS - Receiving Alarm Indication Signal
		7	r	NMF '1' no Multiframe Synchronisation found for 400ms

Diagnostic:

Name	Addr.	Bits	r/w	Function
REC_STAT1	25h	0	r	SI_FAS
		1	r	SI_NFAS
		2	r	A_BIT
		3	r	CRC_OK
		4	r	SI_NFAS
		5	r	unused
		6	r	E1
		7	r	E2
REC_STAT2	26h	3..0	r	SA6 (4:1)
		6..4	r	unused
		7	r	SACSC
REC_STAT3	27h	4..0	r	SA (8:4)
		7..5	r	unused

Name	Addr.	Bits	r/w	Function
REC_FRAME	25h	0	w	Receive Transparent Mode '0' normal operation '1' no synchronisation of input data
		1	w	Automatic Force Resynchronisation '0' normal operation '1' after loss of synchronisation search for Multiframe sync pattern is initiated again (only valid in CRC Multiframe format)
		2	w	Automatic Error Recovery '0' normal operation '1' If there are more than 914 CRC errors in one second the receiver will search for new basic- and multiframing. (only valid in multiframing synchronous state)
		3	w	Service Word Condition Disable '0' Loss of synchronisation if there are three or four (depending on 'Loss of Sync Condition') consecutive incorrect service words. '1' When in synchronous state incorrect service words have no influence
		4	w	Loss of Sync Condition '0' Loss of synchronisation if there are 3 consecutive incorrect FAS or service words '1' Loss of synchronisation if there are 4 consecutive incorrect FAS or service words.
		5	w	Extended CRC4 to NON CRC4 '0' according to ITU-T G.706 '1' according to ITU-T G.706 except that the synchroniser will still search the multiframing even if the 400 ms is expired
		6	w	Multiframe Force Resynchronisation '1' initiate the resynchronisation of CRC Mutliframe alignment without influencing doubleframe synchronous state. If 'Automatic Force Synchronisation' is enabled and multiframe alignment can not be regained, a new search of doubleframe is initiated. (only valid in CRC Multiframe format)
		7	w	Force Resynchronisation '1' initiate resynchronisation of receive frame

Name	Addr.	Bits	r/w	Function
TRANSM0	28h	1..0	w	Transmit Code '00' NRZ '10' AMI Code 'x1' HDB3 Code
		2	w	FULLHALF '0' half bauded (normal operation) '1' full bauded
		3	w	CMI code '0' CMI off '1' CMI on
		4	w	inverted CMI code, only valid if CMI is on '0' CMI code '1' inverted CMI code
		5	w	Polarity of Clock for data clock output (if enable) '0' Clock not inverted '1' Clock inverted
		6	w	Polarity of Data output '0' Data not inverted '1' Data inverted
		7	w	X_Enable Transmit '0' output buffer not enabled '1' output buffer enabled
TRANSM1	29h	0	w	Polarity of Mark '0' normal operation '1' inverted Clock impulses (only valid with CMI Code)
		1	w	TxD-Exchange '0' normal operation '1' exchanged Data output lines
		2	w	Generate AIS output signal (continous 1s)
		6..3		unused, should be '0'
		7	w	Error Counter Mode '0' normal counter operation after reaching maximum counter starts at 0 again '1' Every second the error counter will be reseted automatically after it is latched. The latched state should be read within the next second. During updating reading should be avoided.

Name	Addr.	Bits	r/w	Function	
TRANS_FRA0	2Ch	0	w	Transparent SI Bit in Service Word '0' SI Bit will be generates internally '1' input data will be used internal information of TRANS_FRA1 will be ignored	
		1	w	Transparent SI Bit in FAS '0' SI Bit will be generated internally '1' channel 0 data will be used internal information of TRANS_FRA1 will be ignored	
		2	w	Transparent Remote Alarm '0' Remote Alarm Bit will be generates internally '1' channel 0 data will be used internal inform ation of TRANS_FRA1 will be ignored	
		7..3	w	Transparent SA4..SA8 Bit '0' SA4..SA8 bit will be generates internally '1' channel 0 data will be used	
TRANS_FRA1	2Dh	0	w	Transmit Spare Bit for International USE (FAS)	Only used in Doubleframe Format (bit 1 of TRANS_FRA2 not set)
		1	w	Spare Bit for International Use	
		2	w	Transmit Remote Alarm	
		7..3	w	SA4..SA8 Spare Bits for International Use (Y-Bits, S_n-Bits, S_a-Bits)	
TRANS_FRA2	2Eh	0	w	Transmit Framing Select '0' Doubleframe Format '1' CRC4 Multiframe Format	
		1	w	Time-Slot-0 Transparent Mode '0' normal operation '1' bits of TRANS_FRA1 will be ignored	
		3..2		unused, should be '0'	
		4	w	Automatic Transmission of Submultiframe Status '0' XS13 and XS15 Bits will transmit '1' E-Bits will transmit (only valid in CRC Mutliframe)	
		5	w	Polarity of E-Bit '0' positive E-Bit '1' negative E-Bit	
		6	w	XS13 - Transmit Spare Bit (Frame 13) '0' XS13 is '0' '1' XS13 is '1' (only valid in CRC Mutliframe)	
		7	w	XS15 - Transmit Spare Bit (Frame 15) '0' XS15 is '0' '1' XS15 is '1' (only valid in CRC Mutliframe)	

Name	Addr.	Bits	r/w	Function
RECEIVE_OFF	30h	1..0	w	Receive Elastic Buffer size in number of frames
		2	w	Initialize Buffer (some data may be lost)
		7..3		unused, should be '0'
TRANS_OFF	34h	1..0	w	Transmit Elastic Buffer size in number of frames
		2	w	Initialize Buffer (some data may be lost)
		7..3		unused, should be '0'
STATE / FSM (STATES)	20h	2..0	 r w	F/G State Binary value of actual state (NT: Gx, TE: Fx) Binary value of new state (NT: Gx, TE: Fx) Bit 4 must also be set to load the state.
		3		unused, should be '0'
		4	r/w	Force state '1' loads the prepared state (bit 3..0) and stops the state machine. (reset default) '0' enables the state machine. After writing an invalid state the state machine goes to deactivated state.
		7..5		Unused, should be '0'
E_C_FAS_L	30h	7..0	r	Error count FAS (low byte)
E_C_FAS_H	31h	7..0	r	Error count FAS (high byte)
E_C_VIO_L	32h	7..0	r	Error count VIO (low byte)
E_C_VIO_H	33h	7..0	r	Error count VIO (high byte)
E_C_CRC_L	34h	7..0	r	Error count CRC (low byte)
E_C_CRC_H	35h	7..0	r	Error count CRC (high byte)
E_C_E12_L	36h	7..0	r	Error count E1,2 (low byte)
E_C_E12_H	37h	7..0	r	Error count E1,2 (high byte)
E_C_SA613_L	38h	7..0	r	Error count SA6,1,3 (low byte)
E_C_SA613_H	39h	7..0	r	Error count SA6,1,3 (high byte)
E_C_SA623_L	3Ah	7..0	r	Error count SA6,2,3 (low byte)
E_C_SA623_H	3Bh	7..0	r	Error count SA6,2,3 (high byte)

4 Frame structure

4.1 Allocation of bits 1 to 8 of the frame

Bit number Alternate frames	1	2	3	4	5	6	7	8
frame containing the	S_i	0	0	1	1	0	1	1
frame alignment signal	(note 1)	Frame alignment signal						
frame not containing the	S_i	1	A	S_{a4}	S_{a5}	S_{a6}	S_{a7}	S_{a8}
frame alignment signal	(note 1)	(note 2)	(note 3)	(note 4)				

Table 5: Allocation of bits 1 to 8 of the frame

Note 1: S_i bits used for Cyclic Redundancy Check (CRC) multiframe alignment.
Note 3: This bit is fixed at '1' to assist in avoiding simulations of the frame alignment signal.
Note 3: A = remote alarm indication. In undisturbed operation, set to '0'. In alarm condition, set to '1'.
Note 4: Bits S_{a4} to S_{a8} shall be set to '1' by the TE.

4.2 CRC-4 multiframe structure

Sub-Multiframe (SMF)	Frame number	Bits 1 to 8 of the frame							
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
I	0	C_1	0	0	1	1	0	1	1
	1	0	1	A	S_{a4}	S_{a5}	S_{a6}	S_{a7}	S_{a8}
	2	C_2	0	0	1	1	0	1	1
	3	0	1	A	S_{a4}	S_{a5}	S_{a6}	S_{a7}	S_{a8}
	4	C_3	0	0	1	1	0	1	1
	5	1	1	A	S_{a4}	S_{a5}	S_{a6}	S_{a7}	S_{a8}
	6	C_4	0	0	1	1	0	1	1
	7	0	1	A	S_{a4}	S_{a5}	S_{a6}	S_{a7}	S_{a8}
II	0	C_1	0	0	1	1	0	1	1
	1	1	1	A	S_{a4}	S_{a5}	S_{a6}	S_{a7}	S_{a8}
	2	C_2	0	0	1	1	0	1	1
	3	1	1	A	S_{a4}	S_{a5}	S_{a6}	S_{a7}	S_{a8}
	4	C_3	0	0	1	1	0	1	1
	5	E	1	A	S_{a4}	S_{a5}	S_{a6}	S_{a7}	S_{a8}
	6	C_4	0	0	1	1	0	1	1
	7	E	1	A	S_{a4}	S_{a5}	S_{a6}	S_{a7}	S_{a8}

Table 6: CRC-4 multiframe structure

S_{a4} to S_{a8} = Spare bits
A = Remote alarm indication
C_1 to C_4 = CRC-4 bits
E = CRC-4 error indication bits