Cologne Chip

HFC - S USB
ISDN 2BDS0

ISDN HDLC FIFO controller
with
S/T interface
and
USB interface

July 2001

Revision History

Date	Remarks
Jul. 2001	Changes made on: External circuitries, ISDN USB terminal adapter sample circuitry.
Jan. 2001	Sections added: Auxiliary port circuitry. Information added to: Microprocessor access, PCM/GCI/IOM2 timing, EEPROM circuitry. Changes made on: ISDN USB terminal adapter sample circuitry.
Nov. 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

Timing diagrams

Features

- single chip ISDN-S/T-controller with B- and D-channel HDLC support
- integrated S/T interface
- independent read and write HDLC-channels for 2 ISDN B-channels, one ISDN D-channel and one PCM timeslot (or E-channel)
- B1- and B2-channel transparent mode independently selectable
- integrated FIFOs for B1, B2, D and PCM (or E)
- FIFO size: 128 bytes per channel and direction; up to 7 HDLC frames per FIFO
- 56 kbit/s restricted mode for U.S. ISDN lines selectable by software
- full I.430 ITU S/T ISDN support in TE and NT mode
- PCM128 / PCM64 / PCM30 interface configurable to interface MITEL ST™ bus (MVIP™), Siemens IOM2™ or GCI™ for interface to U-chip or external CODECs
- integrated full speed 12 MBps USB interface (USB specification 1.1 compliant)
- no microcontroller, no firmware required
- integrated auxiliary port (USB bridge)
- CMOS technology
- PQFP 48 case

1 General description

The HFC-S USB is an ISDN S/T HDLC basic rate controller for single-chip USB applications.

The S/T interface, HDLC-controllers, FIFOs and the USB interface are integrated in the HFC-S USB. It only requires an external EEPROM to store the USB configuration data if the default data from the internal ROM is not used. The USB protocol is implemented in hardware. Code development for a microcontroller is not necessary. A PCM128 / PCM64 / PCM30 interface is also implemented which can be connected to many telecom serial busses. CODECs are usually connected to this interface. All ISDN channels (2B+1D) and the PCM interface are served fully duplex by the 8 integrated FIFOs.

The integrated 8 bit auxiliary port enables the HFC-S USB to be used as USB bridge.

HDLC controllers are implemented in hardware so there is no need to implement HDLC on the host computer.

1.1 Applications

- ISDN USB terminal adapters

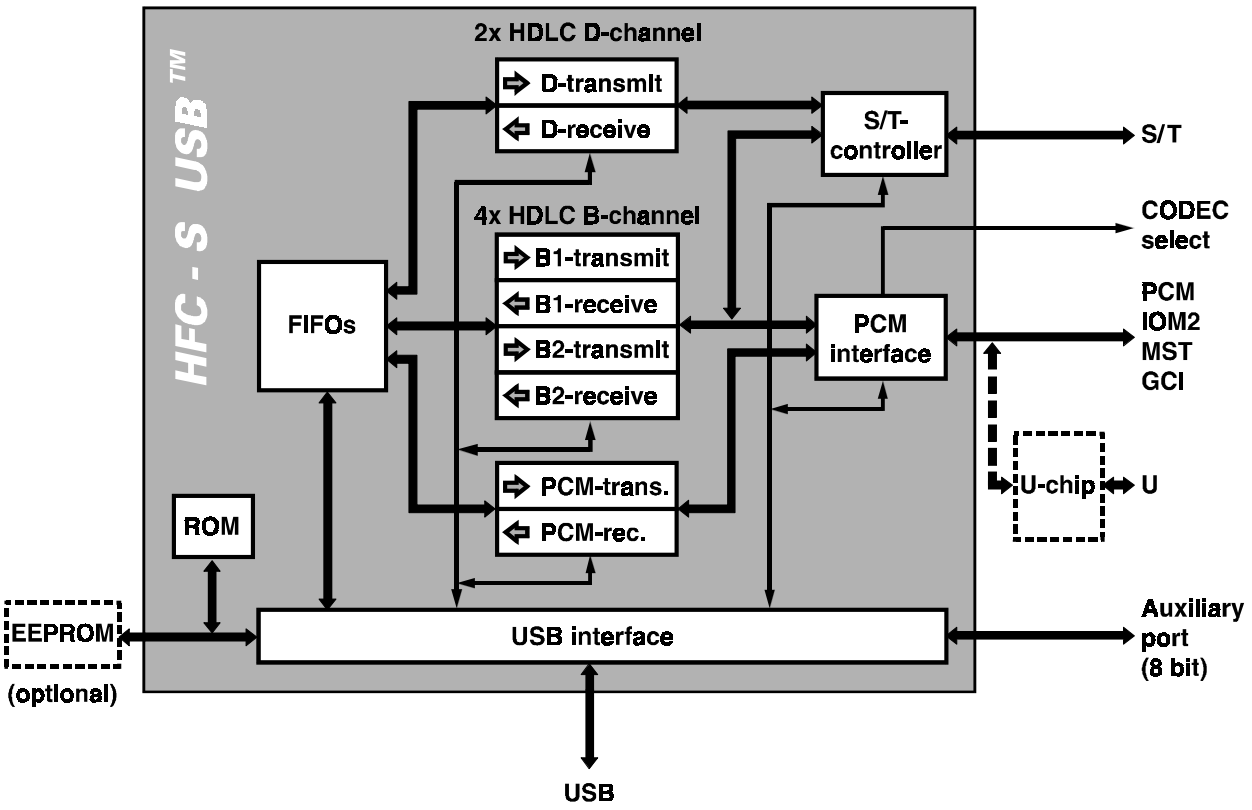


Figure 1: HFC-S USB block diagram

1.2 Mode description

The HFC-S USB has 2 different bus modes, which can be selected like shown in the table below. Depending on the selected mode the function of several pins is different (see: Pin description).

MODE	Selected mode
GND	Passive USB Mode (mode 1)
VDD	Active USB Mode (mode 2)

Table 1: Mode selection

1.2.1 Passive USB Mode (Mode 1)

The Passive USB Mode (mode 1) is used for passive USB terminal adapters. An external processor is not required in this mode.
The microprocessor interface is disabled in this mode and can be used as auxiliary port instead (e.g. for LEDs).

1.2.2 Active USB Mode (Mode 2)

In Active USB Mode (mode 2) the USB interface and ISDN interface of the HFC-S USB are controlled by an external microprocessor.
The microprocessor interface is enabled. The auxiliary port can not be used in this mode.
The data bus is PORT_D[7:0].
A0 (pin 47) is the address bus. The higher address (A0 = '1') is used for register selection and the lower address (A0 = '0') is used for data read/write.

2 Pin description

2.1 Pin description for Passive USB Mode (mode 1)

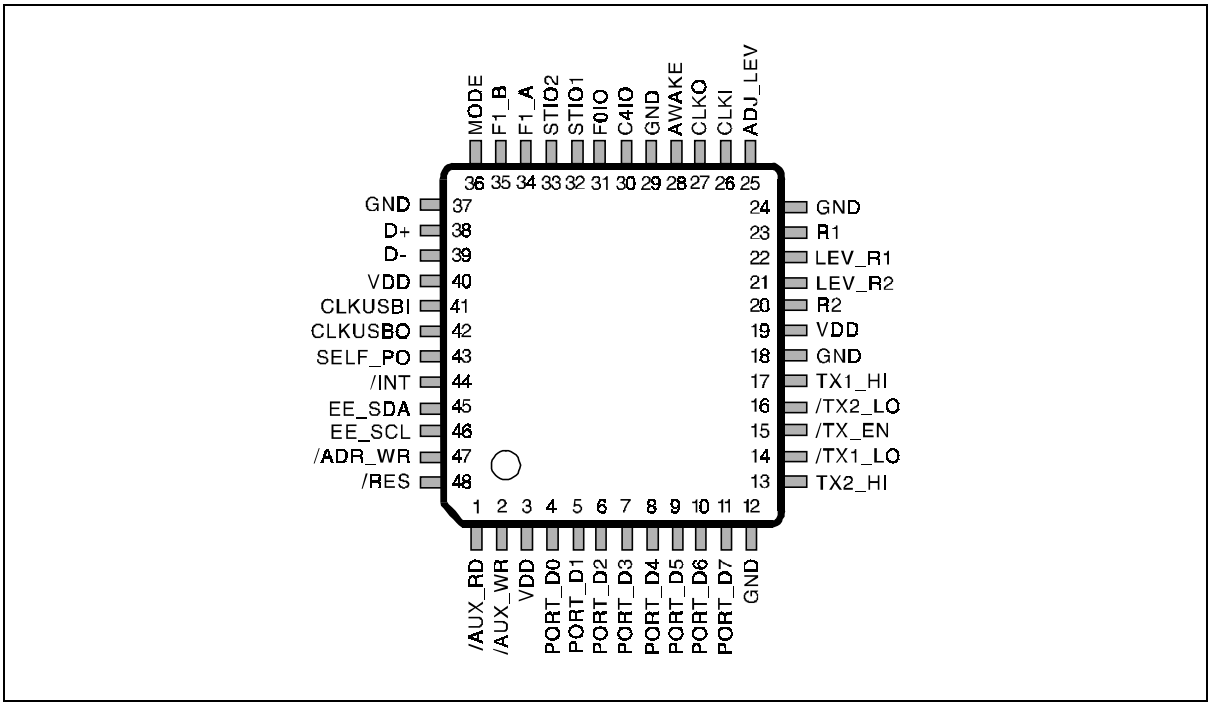


Figure 2: Pin Connection for Passive USB Mode

2.1.1 USB interface signals

For further information please refer to the USB Universal Serial Bus Specification.

Pin No.	Pin Name	Input Output	Function
38	D+	I/O	USB interface data pin +
39	D-	I/O	USB interface data pin -
41	CLKUSBI	I	48 MHz clock input or 48 MHz crystal
42	CLKUSBO	O	48 MHz clock output or 48 MHz crystal
43	SELF_PO	I	Self powered or bus powered indication ('0' = bus powered)

2.1.2 S/T interface transmit signals

Pin No.	Pin Name	Input Output	Function
13	TX2_HI	O	Transmit output 2
14	/TX1_LO	O	GND driver for transmitter 1
15	/TX_EN	O	Transmit enable
16	/TX2_LO	O	GND driver for transmitter 2
17	TX1_HI	O	Transmit output 1

2.1.3 S/T interface receive signals

20	R2	I	Receive data 2
21	LEV_R2	I	Level detect for R2
22	LEV_R1	I	Level detect for R1
23	R1	I	Receive data 1
25	ADJ_LEV	O	Levelgenerator
28	AWAKE	I	Awake input pin for external awake circuitry

2.1.4 PCM bus interface signals

30	C4IO	I/O [u)]	4.096 MHz / 8.192 MHz / 16.384 MHz clock PCM/GCI/IOM2 bus clock master: output PCM/GCI/IOM2 bus clock slave: input (reset default)
31	F0IO	I/O [u)]	Frame synchronisation, 8kHz pulse for PCM/GCI/IOM2 bus frame synchronisation PCM/GCI/IOM2 bus master: output PCM/GCI/IOM2 bus slave: input (reset default)
32	STIO1	I/O [u)]	PCM/GCI/IOM2 bus data line I Slotwise programmable as input or output
33	STIO2	I/O [u)]	PCM/GCI/IOM2 bus data line II Slotwise programmable as input or output
34	F1_A	O	enable signal for external CODEC A or C2IO clock (bit clock) Programmable as positive (reset default) or negative pulse.
35	F1_B	O	enable signal for external CODEC B Programmable as positive (reset default) or negative pulse.

[u)] internal pull up

2.1.5 Auxiliary port and D-interface signals

Pin No.	Pin Name	Input Output	Function
4	PORT_D0	I/O	AUX data bit 0
5	PORT_D1	I/O	AUX data bit 1
6	PORT_D2	I/O	AUX data bit 2
7	PORT_D3	I/O	AUX data bit 3
8	PORT_D4	I/O	AUX data bit 4
9	PORT_D5	I/O	AUX data bit 5
10	PORT_D6	I/O	AUX data bit 6
11	PORT_D7	I/O	AUX data bit 7
47	/ADR_WR	I/O	AUX address write
2	/AUX_WR	I/O	AUX write
1	/AUX_RD	I/O	AUX read
46	EE_SCL	O [u)]	Clock of external EEPROM (only during reset) This pin must be connected to GND if no EEPROM is connected to the HFC-S USB.
45	EE_SDA	I/O [u)]	Serial data of external EEPROM (only during reset)
36	MODE	I	Mode selection (only during reset) Connect to GND for Passive USB Mode.

e) internal pull up

2.1.6 Miscellaneous pins

44	/INT	O	Interrupt request for external processor (low active)
48	/RES	I [u)]	Reset (low active)

u) internal pull up

2.1.7 Oscillator

Pin No.	Pin Name	Input Output	Function
26	CLKI	I	24.576 MHz clock input or 24.576 MHz crystal
27	CLKO	O	24.576 MHz clock output or 24.576 MHz crystal

2.1.8 Power supply

Pin No.	Pin Name	Function
3, 19, 40	VDD	VDD (+3.3V ± 10%)
12, 18, 24, 29, 37	GND	GND

2.2 Pin description for Active USB Mode (mode 2)

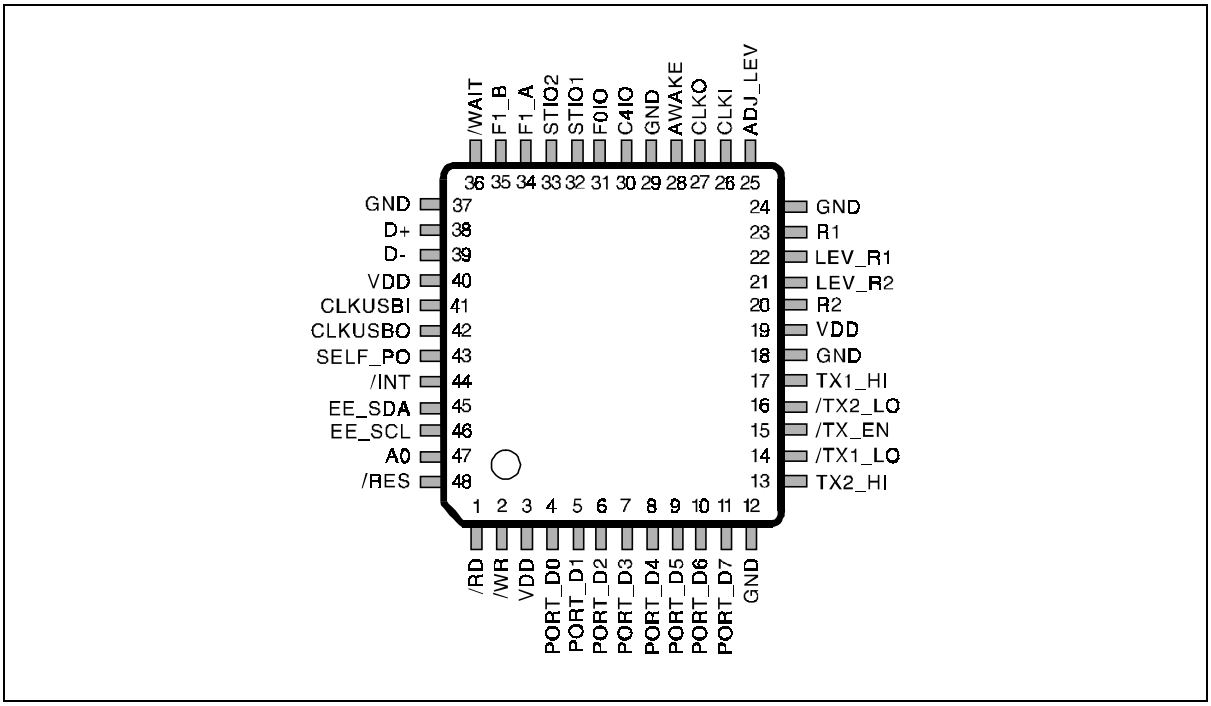


Figure 3: Pin Connection for Active USB Mode

2.2.1 USB interface signals

For further information please refer to the USB Universal Serial Bus Specification.

Pin No.	Pin Name	Input Output	Function
38	D+	I/O	USB interface data pin +
39	D-	I/O	USB interface data pin -
41	CLKUSBI	I	48 MHz clock input or 48 MHz crystal
42	CLKUSBO	O	48 MHz clock output or 48 MHz crystal
43	SELF_PO	I	Self powered or bus powered indication ('0' = bus powered)

2.2.2 S/T interface transmit signals

Pin No.	Pin Name	Input Output	Function
13	TX2_HI	O	Transmit output 2
14	/TX1_LO	O	GND driver for transmitter 1
15	/TX_EN	O	Transmit enable
16	/TX2_LO	O	GND driver for transmitter 2
17	TX1_HI	O	Transmit output 1

2.2.3 S/T interface receive signals

20	R2	I	Receive data 2
21	LEV_R2	I	Level detect for R2
22	LEV_R1	I	Level detect for R1
23	R1	I	Receive data 1
25	ADJ_LEV	O	Levelgenerator
28	AWAKE	I	Awake input pin for external awake circuitry

2.2.4 PCM bus interface signals

30	C4IO	I/O [u)]	4.096 MHz / 8.192 MHz / 16.384 MHz clock PCM/GCI/IOM2 bus clock master: output PCM/GCI/IOM2 bus clock slave: input (reset default)
31	F0IO	I/O [u)]	Frame synchronisation, 8kHz pulse for PCM/GCI/IOM2 bus frame synchronisation PCM/GCI/IOM2 bus master: output PCM/GCI/IOM2 bus slave: input (reset default)
32	STIO1	I/O [u)]	PCM/GCI/IOM2 bus data line I Slotwise programmable as input or output
33	STIO2	I/O [u)]	PCM/GCI/IOM2 bus data line II Slotwise programmable as input or output
34	F1_A	O	enable signal for external CODEC A or C2IO clock (bit clock) Programmable as positive (reset default) or negative pulse.
35	F1_B	O	enable signal for external CODEC B Programmable as positive (reset default) or negative pulse.

[u)] internal pull up

2.2.5 Auxiliary port and D-interface signals

Pin No.	Pin Name	Input Output	Function
4	PORT_D0	I/O	Data bus (bit 0)
5	PORT_D1	I/O	Data bus (bit 1)
6	PORT_D2	I/O	Data bus (bit 2)
7	PORT_D3	I/O	Data bus (bit 3)
8	PORT_D4	I/O	Data bus (bit 4)
9	PORT_D5	I/O	Data bus (bit 5)
10	PORT_D6	I/O	Data bus (bit 6)
11	PORT_D7	I/O	Data bus (bit 7)
47	A0	I	Address bit 0 from external processor (selects between register selection (A0 = '1') and data read/write (A0 = '0'))
2	/WR	I	Write signal from external processor
1	/RD	I	Read signal from external processor
46	EE_SCL	O [u)]	Clock of external EEPROM (only during reset) This pin must be connected to GND if no EEPROM is connected to the HFC-S USB.
45	EE_SDA	I/O [u)]	Serial data of external EEPROM (only during reset)
36	/WAIT MODE	O [e)] I [e)]	Wait signal for external processor (low active) Mode selection (only during reset)

u) internal pull up
e) external pull up required

2.2.6 Miscellaneous pins

44	/INT	O	Interrupt request for external processor (low active)
48	/RES	I [u)]	Reset (low active)

u) internal pull up

2.2.7 Oscillator

Pin No.	Pin Name	Input Output	Function
26	CLKI	I	24.576 MHz clock input or 24.576 MHz crystal
27	CLKO	O	24.576 MHz clock output or 24.576 MHz crystal

2.2.8 Power supply

Pin No.	Pin Name	Function
3, 19, 40	VDD	VDD (+3.3V ± 10%)
12, 18, 24, 29, 37	GND	GND

3 Functional description

3.1 USB interface

A full speed 12MBps USB interface is integrated in the HFC-S USB. It is compliant to USB specification 1.1. The USB interface does not use an internal microcontroller. So code development is obsolete and power consumption is reduced to a minimum.

3.1.1 Register access by USB interface

The internal registers of the HFC-S USB are accessed by USB vendor specific device requests by the host. The register address and (for write accesses) the data to be written must be passed in the setup packet parameters like shown in the table below (see also: Universal Serial Bus Specification Revision 1.1, chapter 9.3).

Offset	Field	Size	Value	Description
0	*bmRequestType*	1	40h for writing data C0h for reading data	direction=host-to-device, type=vendor, recipient=device direction=device-to-host, type=vendor, recipient=device
1	*bRequest*	1	HFC_REG_WR or HFC_REG_RD	specific request for register write (HFC_REG_WR) or register read access (HFC_REG_RD).
2	*wValue (low byte)*	1	data	For write commands this field contains the byte-sized value to be written to the register. This value is ignored in read commands.
3	*wValue (high byte)*	1	ignored	All registers of the HFC-S USB have 8 bits so the high byte is ignored.
4	*wIndex (low byte)*	1	register address	For read and write commands this field must contain the register address.
5	*wIndex (high byte)*	1	ignored	All registers of the HFC-S USB have an one byte address so the high byte is ignored.
6	*wLength*	2	0000h for write, 0001h for read	Only read accesses return data.

Table 1: Setup packet parameters for register access

Name	Value	Description
HFC_REG_WR	0000h	*bRequest* value for register write access.
HFC_REG_RD	0001h	*bRequest* value for register read access.

3.1.2 USB configuration data

The external EEPROM is optional. If no EEPROM is available, EE_SCL must be connected to GND. Without EEPROM the HFC-S USB returns the configuration data from the internal ROM. Communication class descriptors (EE_SDA='0') are stored there as well as generic descriptors (EE_SDA='1'). The tables below show the values for both descriptor types.

DEVICE descriptor

Offset	Field	Size	ROM Value		Description
			generic	comm. class	
0	*bLength*	1	12h		Size of this descriptor in bytes
1	*bDescriptorType*	1	01h		DEVICE Descriptor Type
2	*bcdUSB*	2	0110h		USB Specification Release Number in Binary-Coded Decimal. The HFC-S USB device and its descriptors are compliant with the USB Specification 1.10.
4	*bDeviceClass*	1	FFh	02h	Class code (assigned by the USB).
5	*bDeviceSubClass*	1	FFh	00h	Subclass code (assigned by the USB).
6	*bDeviceProtocol*	1	FFh	00h	Protocol code (assigned by the USB).
7	*bMaxPacketSize0*	1	08h		Maximum packet size for endpoint zero (only 8, 16, 32, or 64 are valid)
8	*idVendor*	2	0959h		Vendor ID (assigned by the USB for Cologne Chip AG)
10	*idProduct*	2	2BD0h		Product ID (assigned by the manufacturer)
12	*bcdDevice*	2	0100h		Device release number in binary-coded decimal (1.0)
14	*iManufacturer*	1	01h		Index of string descriptor describing manufacturer *)
15	*iProduct*	1	01h		Index of string descriptor describing product *)
16	*iSerialNumber*	1	01h		Index of string descriptor describing the device's serial number *)
17	*bNumConfigurations*	1	01h		Number of possible configurations

*) All strings indices (*iManufacturer*, *iProduct*, *iSerialNumber*) point to the same descriptor. So always the same string is displayed at startup.

CONFIGURATION descriptor

Offset	Field	Size	ROM Value		Description
			generic	comm. class	
0	*bLength*	1	09h		Size of this descriptor in bytes
1	*bDescriptorType*	1	02h		CONFIGURATION Descriptor Type
2	*wTotalLength*	2	009Dh		Total length of data returned for this configuration. Includes the combined length of all descriptors returned for this configuration.
4	*bNumInterfaces*	1	02h		Number of interfaces supported by this configuration
5	*bConfigurationValue*	1	01h		Value to use as an argument to the SetConfiguration() request to select this configuration
6	*iConfiguration*	1	00h		Index of string descriptor describing this configuration (no string defined)
7	*bmAttributes*	1	A0h		Configuration characteristics D7: Reserved (must be set to one) D6: Self-powered (not self-powered) D5: Remote Wakeup D4...0: Reserved (reset to zero)
8	*MaxPower*	1	20h		Maximum power consumption of the USB device from the bus in this specific configuration when the device is fully operational. Expressed in 2mA units (64mA in this case).

INTERFACE descriptor 0

Offset	Field	Size	ROM Value		Description
			generic	comm. class	
0	*bLength*	1	09h		Size of this descriptor in bytes
1	*bDescriptorType*	1	04h		INTERFACE Descriptor Type
2	*bInterfaceNumber*	1	00h		Number of interface.
3	*bAlternateSetting*	1	00h		Value used to select alternate setting for the interface identified in the prior field
4	*bNumEndpoints*	1	00h		Number of endpoints used by this interface (excluding endpoint zero).
5	*bInterfaceClass*	1	FFh	02h	Class code (assigned by the USB).
6	*bInterfaceSubClass*	1	FFh	80h	Subclass code (assigned by the USB).
7	*bInterfaceProtocol*	1	FFh	FFh	Protocol code (assigned by the USB). This field is set to FFH, so the device uses a vendor-specific protocol for this interface.
8	*iInterface*	1	00h		Index of string descriptor describing this interface (no string defined)

INTERFACE descriptor 1 alternate 0

Offset	Field	Size	ROM Value		Description
			generic	comm. class	
0	*bLength*	1	09h		Size of this descriptor in bytes
1	*bDescriptorType*	1	04h		INTERFACE Descriptor Type
2	*bInterfaceNumber*	1	01h		Number of interface.
3	*bAlternateSetting*	1	00h		Value used to select alternate setting for the interface identified in the prior field
4	*bNumEndpoints*	1	00h		Number of endpoints used by this interface (excluding endpoint zero).
5	*bInterfaceClass*	1	FFh	0Ah	Class code (assigned by the USB).
6	*bInterfaceSubClass*	1	FFh	00h	Subclass code (assigned by the USB).
7	*bInterfaceProtocol*	1	FFh	FFh	Protocol code (assigned by the USB). This field is set to FFH, so the device uses a vendor-specific protocol for this interface.
8	*iInterface*	1	00h		Index of string descriptor describing this interface (no string defined)

INTERFACE descriptor 1 alternate 1

Offset	Field	Size	ROM Value		Description
			generic	comm. class	
0	*bLength*	1	09h		Size of this descriptor in bytes
1	*bDescriptorType*	1	04h		INTERFACE Descriptor Type
2	*bInterfaceNumber*	1	01h		Number of interface.
3	*bAlternateSetting*	1	01h		Value used to select alternate setting for the interface identified in the prior field
4	*bNumEndpoints*	1	08h		Number of endpoints used by this interface (excluding endpoint zero).
5	*bInterfaceClass*	1	FFh	0Ah	Class code (assigned by the USB).
6	*bInterfaceSubClass*	1	FFh	00h	Subclass code (assigned by the USB).
7	*bInterfaceProtocol*	1	FFh	FFh	Protocol code (assigned by the USB). This field is set to FFH, so the device uses a vendor-specific protocol for this interface.
8	*iInterface*	1	00h		Index of string descriptor describing this interface (no string defined)

ENDPOINT descriptors for isochronous transfer

Offset	Field	Size	ROM Value	Description
0	*bLength*	1	07h	Size of this descriptor in bytes
1	*bDescriptorType*	1	05h	ENDPOINT Descriptor Type
2	*bEndpointAddress*	1	05h	OUT endpoint 5
3	*bmAttributes*	1	01h	01h = Isochronous
4	*wMaxPacketSize*	2	0010h	Maximum packet size of this endpoint is 16 bytes.
6	*bInterval*	1	01h	Interval for polling endpoint (1ms). For isochronous endpoints this field must be set to 1.

Offset	Field	Size	ROM Value	Description
0	*bLength*	1	07h	Size of this descriptor in bytes
1	*bDescriptorType*	1	05h	ENDPOINT Descriptor Type
2	*bEndpointAddress*	1	85h	IN endpoint 5
3	*bmAttributes*	1	01h	01h = Isochronous
4	*wMaxPacketSize*	2	0010h	Maximum packet size of this endpoint is 16 bytes.
6	*bInterval*	1	01h	Interval for polling endpoint (1ms). For isochronous endpoints this field must be set to 1.

Offset	Field	Size	ROM Value	Description
0	*bLength*	1	07h	Size of this descriptor in bytes
1	*bDescriptorType*	1	05h	ENDPOINT Descriptor Type
2	*bEndpointAddress*	1	06h	OUT endpoint 6
3	*bmAttributes*	1	01h	01h = Isochronous
4	*wMaxPacketSize*	2	0010h	Maximum packet size of this endpoint is 16 bytes.
6	*bInterval*	1	01h	Interval for polling endpoint (1ms). For isochronous endpoints this field must be set to 1.

Offset	Field	Size	ROM Value	Description
0	*bLength*	1	07h	Size of this descriptor in bytes
1	*bDescriptorType*	1	05h	ENDPOINT Descriptor Type
2	*bEndpointAddress*	1	86h	IN endpoint 6
3	*bmAttributes*	1	01h	01h = Isochronous
4	*wMaxPacketSize*	2	0010h	Maximum packet size of this endpoint is 16 bytes.
6	*bInterval*	1	01h	Interval for polling endpoint (1ms). For isochronous endpoints this field must be set to 1.

Offset	Field	Size	ROM Value	Description
0	*bLength*	1	07h	Size of this descriptor in bytes
1	*bDescriptorType*	1	05h	ENDPOINT Descriptor Type
2	*bEndpointAddress*	1	07h	OUT endpoint 7
3	*bmAttributes*	1	01h	01h = Isochronous
4	*wMaxPacketSize*	2	0010h	Maximum packet size of this endpoint is 16 bytes.
6	*bInterval*	1	01h	Interval for polling endpoint (1ms). For isochronous endpoints this field must be set to 1.

Offset	Field	Size	ROM Value	Description
0	*bLength*	1	07h	Size of this descriptor in bytes
1	*bDescriptorType*	1	05h	ENDPOINT Descriptor Type
2	*bEndpointAddress*	1	87h	IN endpoint 7
3	*bmAttributes*	1	01h	01h = Isochronous
4	*wMaxPacketSize*	2	0010h	Maximum packet size of this endpoint is 16 bytes.
6	*bInterval*	1	01h	Interval for polling endpoint (1ms). For isochronous endpoints this field must be set to 1.

Offset	Field	Size	ROM Value	Description
0	*bLength*	1	07h	Size of this descriptor in bytes
1	*bDescriptorType*	1	05h	ENDPOINT Descriptor Type
2	*bEndpointAddress*	1	08h	OUT endpoint 8
3	*bmAttributes*	1	01h	01h = Isochronous
4	*wMaxPacketSize*	2	0010h	Maximum packet size of this endpoint is 16 bytes.
6	*bInterval*	1	01h	Interval for polling endpoint (1ms). For isochronous endpoints this field must be set to 1.

Offset	Field	Size	ROM Value	Description
0	*bLength*	1	07h	Size of this descriptor in bytes
1	*bDescriptorType*	1	05h	ENDPOINT Descriptor Type
2	*bEndpointAddress*	1	88h	IN endpoint 8
3	*bmAttributes*	1	01h	01h = Isochronous
4	*wMaxPacketSize*	2	0010h	Maximum packet size of this endpoint is 16 bytes.
6	*bInterval*	1	01h	Interval for polling endpoint (1ms). For isochronous endpoints this field must be set to 1.

INTERFACE descriptor 1 alternate 2

Offset	Field	Size	ROM Value		Description
			generic	comm. class	
0	*bLength*	1	09h		Size of this descriptor in bytes
1	*bDescriptorType*	1	04h		INTERFACE Descriptor Type
2	*bInterfaceNumber*	1	01h		Number of interface.
3	*bAlternateSetting*	1	02h		Value used to select alternate setting for the interface identified in the prior field
4	*bNumEndpoints*	1	08h		Number of endpoints used by this interface (excluding endpoint zero).
5	*bInterfaceClass*	1	FFh	0Ah	Class code (assigned by the USB).
6	*bInterfaceSubClass*	1	FFh	00h	Subclass code (assigned by the USB).
7	*bInterfaceProtocol*	1	FFh	FFh	Protocol code (assigned by the USB). This field is set to FFH, so the device uses a vendor-specific protocol for this interface.
8	*iInterface*	1	00h		Index of string descriptor describing this interface (no string defined)

ENDPOINT descriptors for interrupt transfer

Offset	Field	Size	ROM Value	Description
0	*bLength*	1	07h	Size of this descriptor in bytes
1	*bDescriptorType*	1	05h	ENDPOINT Descriptor Type
2	*bEndpointAddress*	1	01h	OUT endpoint 1
3	*bmAttributes*	1	03h	03h = Interrupt
4	*wMaxPacketSize*	2	0010h	Maximum packet size of this endpoint is 16 bytes.
6	*bInterval*	1	01h	Interval for polling endpoint (1ms).

Offset	Field	Size	ROM Value	Description
0	*bLength*	1	07h	Size of this descriptor in bytes
1	*bDescriptorType*	1	05h	ENDPOINT Descriptor Type
2	*bEndpointAddress*	1	81h	IN endpoint 1
3	*bmAttributes*	1	03h	03h = Interrupt
4	*wMaxPacketSize*	2	0010h	Maximum packet size of this endpoint is 16 bytes.
6	*bInterval*	1	01h	Interval for polling endpoint (1ms).

Offset	Field	Size	ROM Value	Description
0	*bLength*	1	07h	Size of this descriptor in bytes
1	*bDescriptorType*	1	05h	ENDPOINT Descriptor Type
2	*bEndpointAddress*	1	02h	OUT endpoint 2
3	*bmAttributes*	1	03h	03h = Interrupt
4	*wMaxPacketSize*	2	0010h	Maximum packet size of this endpoint is 16 bytes.
6	*bInterval*	1	01h	Interval for polling endpoint (1ms).

Offset	Field	Size	ROM Value	Description
0	*bLength*	1	07h	Size of this descriptor in bytes
1	*bDescriptorType*	1	05h	ENDPOINT Descriptor Type
2	*bEndpointAddress*	1	82h	IN endpoint 2
3	*bmAttributes*	1	03h	03h = Interrupt
4	*wMaxPacketSize*	2	0010h	Maximum packet size of this endpoint is 16 bytes.
6	*bInterval*	1	01h	Interval for polling endpoint (1ms).

Offset	Field	Size	ROM Value	Description
0	*bLength*	1	07h	Size of this descriptor in bytes
1	*bDescriptorType*	1	05h	ENDPOINT Descriptor Type
2	*bEndpointAddress*	1	03h	OUT endpoint 3
3	*bmAttributes*	1	03h	03h = Interrupt
4	*wMaxPacketSize*	2	0010h	Maximum packet size of this endpoint is 16 bytes.
6	*bInterval*	1	01h	Interval for polling endpoint (1ms).

Offset	Field	Size	ROM Value	Description
0	*bLength*	1	07h	Size of this descriptor in bytes
1	*bDescriptorType*	1	05h	ENDPOINT Descriptor Type
2	*bEndpointAddress*	1	83h	IN endpoint 3
3	*bmAttributes*	1	03h	03h = Interrupt
4	*wMaxPacketSize*	2	0010h	Maximum packet size of this endpoint is 16 bytes.
6	*bInterval*	1	01h	Interval for polling endpoint (1ms).

Offset	Field	Size	ROM Value	Description
0	*bLength*	1	07h	Size of this descriptor in bytes
1	*bDescriptorType*	1	05h	ENDPOINT Descriptor Type
2	*bEndpointAddress*	1	04h	OUT endpoint 4
3	*bmAttributes*	1	03h	03h = Interrupt
4	*wMaxPacketSize*	2	0010h	Maximum packet size of this endpoint is 16 bytes.
6	*bInterval*	1	01h	Interval for polling endpoint (1ms).

Offset	Field	Size	ROM Value	Description
0	*bLength*	1	07h	Size of this descriptor in bytes
1	*bDescriptorType*	1	05h	ENDPOINT Descriptor Type
2	*bEndpointAddress*	1	84h	IN endpoint 4
3	*bmAttributes*	1	03h	03h = Interrupt
4	*wMaxPacketSize*	2	0010h	Maximum packet size of this endpoint is 16 bytes.
6	*bInterval*	1	01h	Interval for polling endpoint (1ms).

STRING descriptor 0

Offset	Field	Size	ROM Value	Description
0	*bLength*	1	04h	Size of this descriptor in bytes
1	*bDescriptorType*	1	03h	STRING Descriptor Type
2	*wLANGID[0]*	2	0409h	LANGID code zero

STRING descriptor 1

Offset	Field	Size	ROM Value	Description
0	*bLength*	1	18h	Size of this descriptor in bytes
1	*bDescriptorType*	1	03h	STRING Descriptor Type
2	*bString*	22		UNICODE encoded string "ISDN USB TA"

3.1.3 Writing the USB configuration EEPROM

The EEPROM can be programmed in an existing USB device. The EEPROM Programming Spec. is only available on special request to avoid destruction of configuration information by not authorized programs or software viruses.

3.1.4 Standard device requests

The USB Specification 1.1. makes a difference for standard device requests in Address state and in Configured state. In Address state a device should not accept some standard device requests. Even in Address state the HFC-S USB reacts as being in Configured state. Furthermore the HFC-S USB behavior is independent of a selected configuration or interface value. In both states a Get Configuration returns the configuration value set by Set Configuration and Get Interface returns the interface value set by Set interface. The HFC-S USB behaves to all standard device requests (chapter 9.4 of the USB Specification 1.1, except 9.4.8 Set Descriptor and 9.4.11 Synch Frame which are not supported) in Address state as if in Configured state.

3.2 Microprocessor interface

The HFC-S USB has an integrated 8 bit microprocessor interface. The processor interface is enabled in Active USB Mode (mode 2) only. Pin A0 is the address input. The data bus is PORT_D[7:0]. The inputs /RD and /WR are used to control read and write operations.

3.2.1 Register access by microprocessor interface

The HFC-S USB has 2 addresses in Active USB Mode (mode 2). The lower address (A0 = '0') is used for data read/write. The higher address (A0 = '1') is write only and is used for register selection. Registers are selected by first setting A0 to '1' and then writing the address of the desired register to the data bus PORT_D[7:0]. All following accesses to the HFC-S USB with A0 = '0' are read/write operations to this register.

3.3 USB bridge

The HFC-S USB has an integrated 8 bit auxiliary port with multiplexed address/data bus which can be used as USB bridge in mode 1 (see also Mode description on page 8). A microprocessor is not required to use this USB bridge. The host can easily write an address (P_ADR_W register) to the auxiliary port and then read/write data (P_DATA register) from/to this address. The device connected to USB bridge of the HFC-S USB is passive.
The registers P_ADR_W (address write) and P_DATA (data read/write) are used by the host to control the auxiliary port.
PORT_D[7:0] is the multiplexed address/data bus.
The active (low cycle) time of the read and write control signals /AUX_RD and /AUX_WR can be adjusted by the CIRM register (see also Auxiliary port access on page 57).

3.4 FIFOs

There is a transmit and a receive FIFO with HDLC-controller for each of the two B-channels, for the D-channel and for the PCM interface in the HFC-S USB. Each FIFO has 128 bytes length in each direction. Up to 7 frames can be stored in each FIFO.

The HDLC circuits are located on the S/T device side of the HFC-S USB. So always plain data is stored in the FIFOs. Zero insertion and CRC checksum processing for receive and transmit data is done by the HFC-S USB automatically.

3.4.1 FIFO endpoints and transfer types

The FIFOs can be accessed by isochronous data transfer (endpoints 5..8) or bulk/interrupt data transfer (endpoints 1..4). The table below shows how FIFOs can be read/written. Each FIFO has two endpoints: one for bulk/interrupt data transfer and one for isochronous data transfer. All endpoint numbers are bidirectional. This means by writing data from the host to an endpoint of the HFC-S USB the transmit FIFO is accessed. If the host reads data from an endpoint of the HFC-S USB the corresponding receive FIFO is accessed.

Endpoint	IN / OUT	FIFO	FIFO#	Transfer Type
1	OUT	B1-transmit	0	Bulk / Interrupt
	IN	B1-receive	1	Bulk / Interrupt
2	OUT	B2-transmit	2	Bulk / Interrupt
	IN	B2-receive	3	Bulk / Interrupt
3	OUT	D-transmit	4	Bulk / Interrupt
	IN	D-receive	5	Bulk / Interrupt
4	OUT	PCM-transmit	6	Bulk / Interrupt
	IN	PCM-receive	7	Bulk / Interrupt
5	OUT	B1-transmit	0	Isochronous
	IN	B1-receive	1	Isochronous
6	OUT	B2-transmit	2	Isochronous
	IN	B2-receive	3	Isochronous
7	OUT	D-transmit	4	Isochronous
	IN	D-receive	5	Isochronous
8	OUT	PCM-transmit	6	Isochronous
	IN	PCM-receive	7	Isochronous

Table 2: FIFO endpoints and transfer types

3.4.2 FIFO control bytes

3.4.2.1 FIFO control bytes for receive FIFOs

For the receive FIFOs (IN transfer on enpoints 1..4 or 5..8) the first two data bytes of the first data packet are used as FIFO control bytes (see also Table 3 and Figure 4). Then the data bytes of the FIFO selected by the endpoint number are transmitted.

FIFO Control Bytes for Receive FIFOs (Host Receives Data)																
	Byte 1								Byte 2							
Bit	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
Name	STATES[3:0]				unused		ERR	EoF	F_FILL[7:0]							

Bit Name	Description
F_FILL[7:0]	Bits 7..0 of the F_FILL register (indicate which FIFOs are over threshold)
STATES[3:0]	Bits 3..0 of STATES register (current state of TE/NT state machine)
ERR	'1' A receive data error on an isochronous OUT transfer has occured. This bit is automatically reset after the next ISO-OUT transfer whithout errors.
EoF	'1' end of HDLC frame after data transfer In transparent mode this bit is always '0'.

Table 3: FIFO control bytes for receive FIFOs

The HFC-S USB assumes a data transfer as finished if the data packet size is less than *wMaxPacketSize* (see also USB_SIZE register).
For receive FIFOs the EoF-bit is set to '1' if the HDLC frame ends after the data transfer. If the last data packet has the same length as *wMaxPacketSize* an empty data packet is sent next (see Figure 4).
As you can see in the figure below the FIFO control bytes are only sent in the first data packet of a transfer.

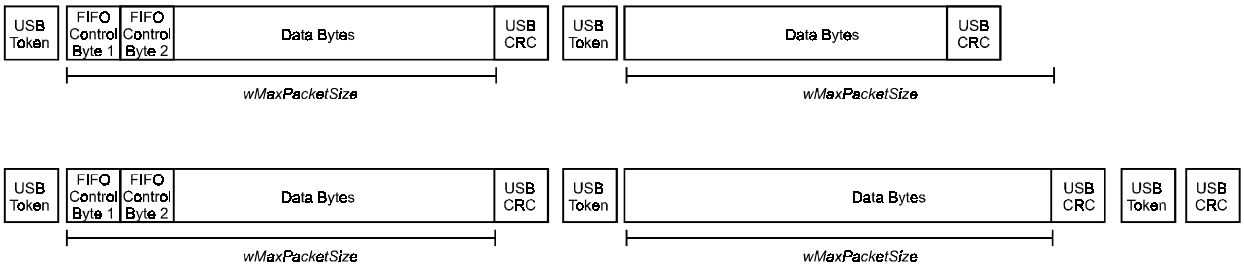


Figure 4: FIFO control bytes for receive FIFOs

important!
The *wMaxPacketSize* in the USB endpoint descriptors must be the same as the size selected in the USB_SIZE register (or USB_SIZE_I register for isochronous transfers).

3.4.2.2 FIFO control bytes for transmit FIFOs

For the transmit FIFOs (OUT transfer on enpoints 1..4 or 5..8) the first data byte of the first data packet of a transfer is used as FIFO control byte (see Table 4 and Figure 5). The host must indicate the end of a HDLC frame in this FIFO control byte if the HDLC frame ends after this transfer.

FIFO Control Byte for Transmit FIFOs (Host Transmits Data)								
Bit	7	6	5	4	3	2	1	0
Name	unused							EoF

Bit Name	Description
EoF	'1' end of HDLC frame after data transfer In transparent mode this bit must be '0'.

Table 4: FIFO control byte for transmit FIFOs

The HFC-S USB assumes a data transfer as finished if the data packet size is less than *wMaxPacketSize* (see also USB_SIZE register).
Figure 5 shows how a complete HDLC frame can be transmitted to the HFC-S USB. The EoF-bit in the FIFO control byte must be set to '1' if the HDLC transmit frame ends after the USB transfer. If the last data packet has the same length as *wMaxPacketSize* an empty data packet must be sent next. Otherwise the HFC-S USB would assume the data transfer (and the HDLC frame) as not yet finished.
As you can see in the figure above the FIFO control byte is only required in the first data packet of a transfer.

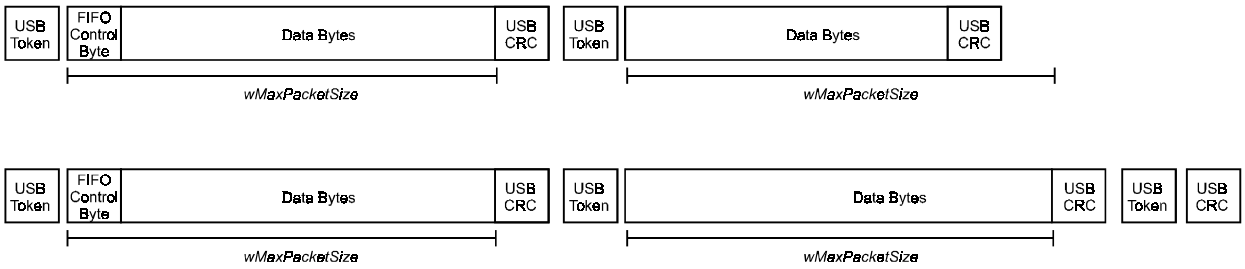


Figure 5: FIFO control byte for transmit FIFOs

☝ **important!**
The *wMaxPacketSize* in the USB endpoint descriptors must be the same as the size selected in the USB_SIZE register (or USB_SIZE_I register for isochronous transfers).

3.4.3 FIFO initialization

After reset all FIFOs are disabled. To enable a FIFO at least one of bits[4:1] of the CON_HDLC register for the corresponding FIFO must be set to '1'.

For D-channel FIFOs the inter frame fill bit (bit 0 of CON_HDLC register) must be set to '1'. The HDLC_PAR register must be set to 02h ('0000 0010').

3.5 Transparent mode of HFC-S USB

You can switch off HDLC operation for each B-channel independently. There is one bit for each B-channel in the CON_HDLC control register. If this bit is set data in the FIFO is sent directly to the S/T or PCM bus interface and data from the S/T or PCM bus interface is sent directly to the FIFO.

The FIFOs should be empty when switching into transparent mode.

If a send FIFO channel changes to FIFO empty condition no CRC is generated and the last data byte in the FIFO memory is repeated until there is new data. If the last data byte which was written to the selected FIFO should be repeated the last byte must be written without increment of Z-counter (FIF_DATA register, address 84h).

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 S/T or PCM bus interface or is sent to this.

Send and receive transparent data can be handled in two ways. The usual way is transporting 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 the corresponding bit in the F_CROSS register.

3.6 Power down considerations

In suspend mode the power consumption must be reduced to a minimum. To avoid current generated by floating inputs in suspend mode the auxiliary port data bus (PORT_D[7:0]) must be put to GND or VDD. If another device is connected to these pins it is useful to connect each pin of the PORT_D data bus to GND over a resistor of about 1MΩ.
If no other device is connected to PORT_D[7:0] it is also possible to set the bus to driving out. In this case bit 4 of the CIRM register must be set.

3.7 Configuring test loops

For electrical tests of layer 1 it is useful to create a S/T test loop for the B1/B2 channel. The test loop described here transmits the data that has been received on the B1 or B2 channel to the same channel on the S/T interface. To configure the test loop the following must be done:

- write **0Fh** to register CLKDEL (**37h**) // Adjust the phase offset between receive and transmit direction (the value depends on the external circuitry).
- write **43h** to register SCTRL (**31h**) // 03h is to enable B1, B2 at the S/T interface for transmission // 40h is for TX_LO setup (capacitive line mode)
- write **00h** to register STATES (**30h**) // Release S/T state machine for activation over the S/T interface by incoming INFO 2 or INFO 4.
- write **03h** to register SCTRL_R (**33h**) // Configure S/T B1 and B2 channel to normal receive operation.
- write **00h** to register FIFO# (**0Fh**) // Select B1 transmit
- write **C4h** to register CON_HDLC (**FAh**) // Configure B1 transmit channel for test loop
- write **01h** to register FIFO# (**0Fh**) // Select B1 receive
- write **C4h** to register CON_HDLC (**FAh**) // Configure B1 receive channel for test loop
- write **02h** to register FIFO# (**0Fh**) // Select B2 transmit
- write **C4h** to register CON_HDLC (**FAh**) // Configure B2 transmit channel for test loop
- write **03h** to register FIFO# (**0Fh**) // Select B2 receive
- write **C4h** to register CON_HDLC (**FAh**) // Configure B2 receive channel for test loop
- write **80h** to register B1_SSL (**20h**) // Enable transmit channel for PCM/GCI/IOM2 bus, pin STIO1 is used as output, use time slot #0.
- write **C0h** to register B1_RSL (**24h**) // Enable receive channel for PCM/GCI/IOM2 bus, pin STIO1 is used as input, use time slot #0.
- write **81h** to register B2_SSL (**21h**) // Enable transmit channel for PCM/GCI/IOM2 bus, pin STIO1 is used as output, use transmission slot #1.
- write **C1h** to register B2_RSL (**25h**) // Enable receive channel for PCM/GCI/IOM2 bus, pin STIO1 is used as input, use time slot #1.
- write **01h** to register MST_MODE0 (**14h**) // Configure HFC-S USB as PCM/GCI/IOM2 bus master.

4 Register description

4.1 Register reference list

4.1.1 Registers by address

Registers by Address		
Name	**Address**	**Page**
CIRM	00h	36
FIF_Z1 []	04h	38
FIF_Z2 []	06h	38
USB_SIZE_I	06h	37
USB_SIZE	07h	36
F_CROSS	0Bh	37
F_THRES	0Ch	38
FIF_F1 []	0Ch	38
F_MODE	0Dh	37
FIF_F2 []	0Dh	38
INC_RES_F []	0Eh	37
FIFO#	0Fh	37
INT_S1	10h	39
INT_S2	11h	40
MST_MODE0	14h	46
MST_MODE1	15h	47
CHIP_ID	16h	44
MST_MODE2	16h	47
F0_CNT_L	18h	47
F0_CNT_H	19h	47
F_USAGE []	1Ah	37
INT_M1	1Ah	41
F_FILL	1Bh	38
INT_M2	1Bh	41
STATUS	1Ch	44
P_ADR_W	1Eh	44
P_DATA	1Fh	44
B1_SSL	20h	45
B2_SSL	21h	45
AUX1_SSL	22h	45
AUX2_SSL	23h	45
B1_RSL	24h	45
B2_RSL	25h	45

Registers by Address		
Name	**Address**	**Page**
AUX1_RSL	26h	45
AUX2_RSL	27h	45
C/I	28h	47
TRxR	29h	48
MON1_D	2Ah	48
MON2_D	2Bh	48
B1_D	2Ch	45
B2_D	2Dh	45
AUX1_D	2Eh	45
AUX2_D	2Fh	45
STATES	30h	49
SCTRL	31h	50
SCTRL_E	32h	50
SCTRL_R	33h	51
SQ_REC	34h	51
SQ_SEND	34h	51
CLKDEL	37h	51
B1_REC	3Ch	52
B1_SEND	3Ch	52
B2_REC	3Dh	52
B2_SEND	3Dh	52
D_REC	3Eh	52
D_SEND	3Eh	52
E_REC	3Fh	52
FIF_DATA []	80h	37
FIF_DATA []	84h	37
CON_HDLC []	FAh	42
HDLC_PAR []	FBh	41

4.1.2 Registers by name

Registers by Name		
Name	**Address**	**Page**
AUX1_D	2Eh	45
AUX1_RSL	26h	45
AUX1_SSL	22h	45
AUX2_D	2Fh	45
AUX2_RSL	27h	45
AUX2_SSL	23h	45
B1_D	2Ch	45
B1_REC	3Ch	52
B1_RSL	24h	45
B1_SEND	3Ch	52
B1_SSL	20h	45
B2_D	2Dh	45
B2_REC	3Dh	52
B2_RSL	25h	45
B2_SEND	3Dh	52
B2_SSL	21h	45
C/I	28h	47
CHIP_ID	16h	44
CIRM	00h	36
CLKDEL	37h	51
CON_HDLC []	FAh	42
D_REC	3Eh	52
D_SEND	3Eh	52
E_REC	3Fh	52
F_CROSS	0Bh	37
F_FILL	1Bh	38
F_MODE	0Dh	37
F_THRES	0Ch	38
F_USAGE []	1Ah	37
F0_CNT_H	19h	47
F0_CNT_L	18h	47
FIF_DATA []	80h	37

Registers by Name		
Name	**Address**	**Page**
FIF_DATA []	84h	37
FIF_F1 []	0Ch	38
FIF_F2 []	0Dh	38
FIF_Z1 []	04h	38
FIF_Z2 []	06h	38
FIFO#	0Fh	37
HDLC_PAR []	FBh	41
INC_RES_F []	0Eh	37
INT_M1	1Ah	41
INT_M2	1Bh	41
INT_S1	10h	39
INT_S2	11h	40
MON1_D	2Ah	48
MON2_D	2Bh	48
MST_MODE0	14h	46
MST_MODE1	15h	47
MST_MODE2	16h	47
P_ADR_W	1Eh	44
P_DATA	1Fh	44
SCTRL	31h	50
SCTRL_E	32h	50
SCTRL_R	33h	51
SQ_REC	34h	51
SQ_SEND	34h	51
STATES	30h	49
STATUS	1Ch	44
TRxR	29h	48
USB_SIZE	07h	36
USB_SIZE_I	06h	37

4.2 FIFO, interrupt, status and control registers

Name	Addr.	Bits	r/w	Function
CIRM	00h	2..0		defines the length of the auxiliary port access: Value Cycle time (/AUX_WR or /AUX_RD low) '000' 2 CLKI Clock '001' 6 CLKI Clocks '010' 10 CLKI Clocks '011' 14 CLKI Clocks '100' 18 CLKI Clocks '101' 22 CLKI Clocks '110' 26 CLKI Clocks '111' 30 CLKI Clocks
		3	w	soft reset The reset is active until the bit is cleared. '0' deactivate reset (reset default) '1' activate reset
		4	w	auxiliary port mode '0' port is tristated when not accessed '1' data out is valid until the next auxiliary port write access is initiated (e.g. for LEDs) (reset default)
		7..5		unused, must be '0'
USB_SIZE	07h	3..0	w	size of USB out transactions for bulk and interrupt transfers '0000' 0 bytes '0001' 8 bytes (reset default) '0010' 16 bytes : : '1111' 120 bytes **The *wMaxPacketSize* of the endpoint descriptor must match with the size selected here.**
		7..4	w	size of USB in transactions for bulk and interrupt transfers '0000' 0 bytes '0001' 8 bytes (reset default) '0010' 16 bytes : : '1111' 120 bytes **The *wMaxPacketSize* of the endpoint descriptor must match with the size selected here.**

Name	Addr.	Bits	r/w	Function
USB_SIZE_I	06h	6..0	w	size of USB transactions for isochronous transfers in bytes 10h = 16 bytes (reset default) **The *wMaxPacketSize* of the endpoint descriptor must match with the size selected here.**
		7	w	unused, should be '0'
F_CROSS	0Bh	Select bit order for FIFO data '0' normal bit order (LSB first, reset default) '1' reverse bit order (MSB first)		
		0	w	B1-transmit
		1	w	B1-receive
		2	w	B2-transmit
		3	w	B2- receive
		4	w	D-transmit
		5	w	D- receive
		6	w	PCM-transmit
		7	w	PCM-receive
F_MODE	0Dh	6..0	w	must be '0'
		7	w	Channel Select Mode enable (CSM)
INC_RES_F [FIFO#]	0Eh	0	w	increment F-counter of selected FIFO ('1'=increment)
		1	w	reset selected FIFO ('1'=reset FIFO)
		7..2	w	unused, should be '0'
FIFO#	0Fh	2..0	w	FIFO select '000' B1-transmit '001' B1-receive '010' B2-transmit '011' B2-receive '100' D-transmit '101' D-receive '110' PCM-transmit '111' PCM-receive
		7..3	w	unused, should be '0'
FIF_DATA [FIFO#]	80h	7..0	w	FIFO data register read/write data from/to the FIFO selected in the FIFO# register and increment Z-counter
	84h	7..0	w	FIFO data register (alternate) read/write data from/to the FIFO selected in the FIFO# register without incrementing Z-counter
F_USAGE [FIFO#]	1Ah	7..0	w	fill level of FIFO in bytes

Name	Addr.	Bits	r/w	Function
FIF_F1 [FIFO#]	0Ch	7..0	r	FIFO input HDLC frame counter (F1) Up to 7 HDLC frames can be stored in each FIFO.
FIF_F2 [FIFO#]	0Dh	7..0	r	FIFO output HDLC frame counter (F2) Up to 7 HDLC frames can be stored in each FIFO.
FIF_Z1 [FIFO#]	04h	7..0	r	FIFO input counter (Z1)
FIF_Z2 [FIFO#]	06h	7..0	r	FIFO output counter (Z2)
F_THRES	0Ch	3..0	w	transmit FIFO (OUT transfer on endpoints 1..8) threshold for B1-transmit, B2-transmit, D-transmit and PCM-transmit (see also F_FILL) '0000' 0 bytes '0001' 8 bytes (reset default) : : '1111' 120 bytes The corresponding bit(s) in the F_FILL register are set if the number of bytes in a transmit FIFO is greater or equal than this value.
		7..4	w	receive FIFO (IN transfer on endpoints 1..8) threshold for B1-receive, B2-receive, D-receive and PCM-receive (see also F_FILL) '0000' 0 bytes '0001' 8 bytes (reset default) : : '1111' 120 bytes The corresponding bit(s) in the F_FILL register are set if the number of bytes in a receive FIFO is greater or equal than this value.
F_FILL	1Bh	'0' Number of bytes in the following FIFOs is lower than the value defined in the F_THRES register. '1' Number of bytes in the following FIFOs is greater or equal than the value defined in the F_THRES register.		
		0	r	B1-transmit
		1	r	B1-receive
		2	r	B2-transmit
		3	r	B2-receive
		4	r	D-transmit
		5	r	D-receive
		6	r	PCM-transmit
		7	r	PCM-receive

Name	Addr.	Bits	r/w	Function
INT_S1	10h	0	r	B1-channel interrupt status in transmit direction '1' a complete frame has been transmitted, the frame counter F2 has been incremented
		1	r	B1-channel interrupt status in receive direction '1' a complete frame has been transmitted, the frame counter F1 has been incremented
		2	r	B2-channel interrupt status in transmit direction '1' a complete frame has been transmitted, the frame counter F2 has been incremented
		3	r	B2-channel interrupt status in receive direction '1' a complete frame has been transmitted, the frame counter F1 has been incremented
		4	r	D-channel interrupt status in transmit direction '1' a complete frame was transmitted, the frame counter F2 was incremented
		5	r	D-channel interrupt status in receive direction '1' a complete frame was transmitted, the frame counter F1 was incremented
		6	r	PCM-channel interrupt status in transmit direction '1' a complete frame was transmitted, the frame counter F2 was incremented
		7	r	PCM-channel interrupt status in receive direction '1' a complete frame was transmitted, the frame counter F1 was incremented

☝ **note!**
The interrupts indicated in the INT_S1 register are frame interrupts which occur in HDLC mode. In transparent mode an interrupt can be generated on a regular basis. Interrupt frequency can be selected in the CON_HDLC register.

Name	Addr.	Bits	r/w	Function
INT_S2	11h	0	r	TE/NT state machine interrupt status '1' state of state machine changed
		1	r	timer interrupt status '1' timer is elapsed
		2	r	processing/non processing transition interrupt status '1' The HFC-S USB has changed from processing to non processing state.
		3	r	GCI I-change interrupt '1' a different I-value on GCI was detected
		4	r	receiver ready (RxR) of monitor channel '1' 2 monitor bytes have been received
		5	r	USB interrupt '1' bit 0 of register 01h has been set to '1' by a USB vendor request
		7..6	r	unused, '0'

☝ **important!**
Reading the INT_S1 or INT_S2 register resets all active read interrupts in the INT_S1 or INT_S2 register respectively. New interrupts may occur during read. These interrupts are reported at the next read of INT_S1 or INT_S2.
All interrupt bits are reported regardless of the mask registers settings (INT_M1 and INT_M2). The mask registers settings only influence the interrupt output condition.
The interrupt output goes inactive during the read of INT_S1 or INT_S2. If interrupts occur during this read the interrupt line goes active immediately after the read is finished. So processors with level or transition triggered interrupt inputs can be connected.

Name	Addr.	Bits	r/w	Function
INT_M1	1Ah	0	w	interrupt mask for channel B1 in transmit direction
		1	w	interrupt mask for channel B1 in receive direction
		2	w	interrupt mask for channel B2 in transmit direction
		3	w	interrupt mask for channel B2 in receive direction
		4	w	interrupt mask for channel D in transmit direction
		5	w	interrupt mask for channel D in receive direction
		6	w	interrupt mask for channel PCM in transmit direction
		7	w	interrupt mask for channel PCM in receive direction
INT_M2	1Bh	0	w	interrupt mask for TE/NT state machine state change
		1	w	interrupt mask for timer
		2	w	interrupt mask for processing/non processing transition
		3	w	interrupt mask for GCI I-change
		4	w	interrupt mask for receiver ready (RxR) of monitor channel
		5	w	interrupt mask for USB interrupt
		6	w	interrupt output is reversed
		7	w	enable interrupt output

For mask bits a '1' enables and a '0' disables interrupt. RESET clears all bits to '0'.

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) '000' process 8 bits (64kbit/s) (reset default) '001' process 1 bit : : '111' process 7 bits (56kbit/s)
		5..3	w	start bit for HDLC and transparent mode '000' start processing with bit 0 (reset default) : : '111' 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.
For D-channels the HDLC_PAR register must be set to 02h.

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 (must be set for D-channel)
		1	w	HDLC mode/transparent mode select '0' HDLC mode (reset default) '1' transparent mode select
		3..2	w	transparent mode interrupt frequency select '00' every 8 bytes '01' every 16 bytes '10' every 32 bytes '11' every 64 bytes
		4	w	must be '0'
		7..5	w	select data flow for selected FIFO destination ← source B1-channel (FIFO0 and 1, see FIFO#): bit 5: '0' FIFO1 ← B1-S/T '1' FIFO1 ← B1-PCM bit 6: '0' B1-S/T ← FIFO0 '1' B1-S/T ← B1-PCM bit 7: '0' B1-PCM ← FIFO0 '1' B1-PCM ← B1-S/T B2-channel (FIFO2 and 3, see FIFO#): bit 5: '0' FIFO3 ← B2-S/T '1' FIFO3 ← B2-PCM bit 6: '0' B2-S/T ← FIFO2 '1' B2-S/T ← B2-PCM bit 7: '0' B2-PCM ← FIFO2 '1' B2-PCM ← B2-S/T D-channel and PCM (FIFO4 and 5, see FIFO#): bit 5: '0' FIFO5 ← D-S/T '1' FIFO5 ← AUX1 bit 6: '0' D-S/T ← FIFO4 '1' D-S/T ← AUX1 bit 7: '0' AUX1 ← FIFO4 '1' AUX1 ← D-S/T E-channel and PCM (FIFO6 and 7, see FIFO#): bit 5: '0' FIFO7 ← E-S/T '1' FIFO7 ← AUX2 bit 6: '0' E-S/T ← FIFO6 '1' E-S/T ← AUX2 bit 7: '0' AUX2 ← FIFO6 '1' AUX2 ← E-S/T **CON_HDLC register bits[7:5] must be the same for corresponding receive and transmit FIFOs.**

if bits 3..1 are '0000' the FIFO is disabled (reset default)

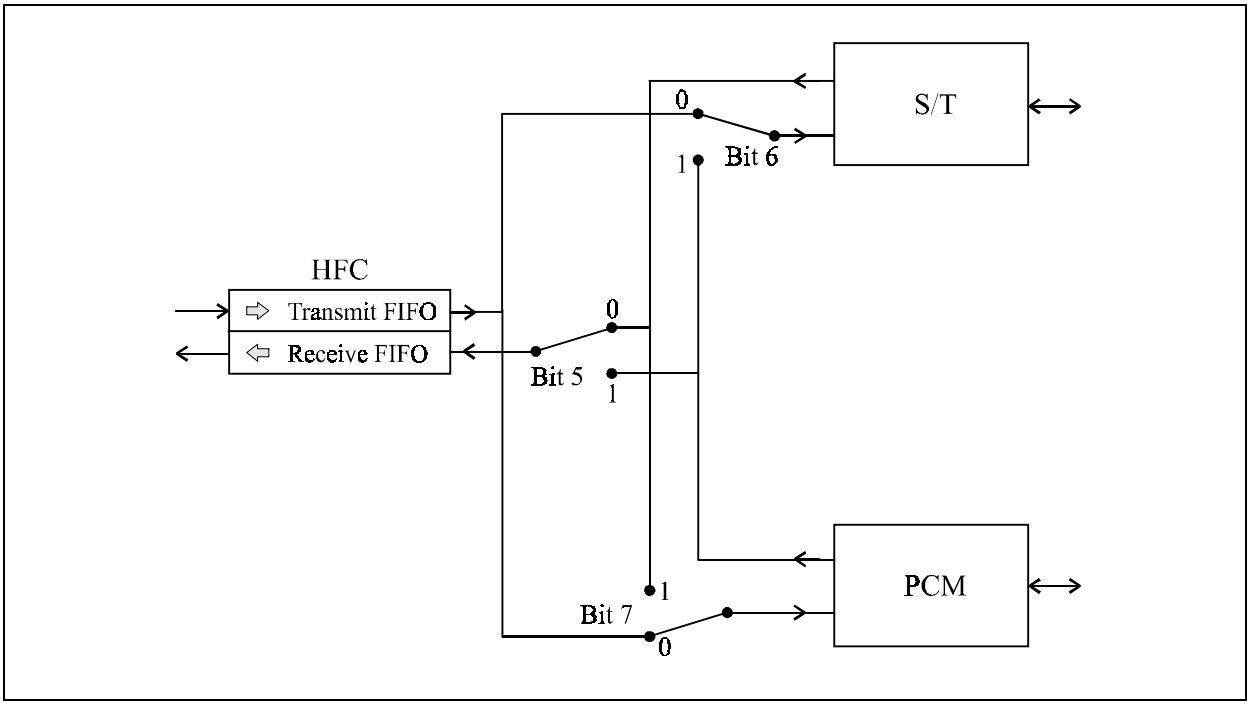


Figure 6: Function of CON_HDLC register bits 7..5

Name	Addr.	Bits	r/w	Function
CHIP_ID	16h	3..0	r	unused, '0'
		7..4	r	Chip identification '0100' HFC-S USB
STATUS	1Ch	0	r	BUSY/NOBUSY status '1' the HFC-S USB is BUSY after initializing reset FIFO, increment F or change FIFO '0' the HFC-S USB is not busy, all accesses are allowed
		1	r	processing/non processing status '1' the HFC-S USB is in processing phase (every 125µs) '0' the HFC-S USB is not in processing phase
		5..2	r	unused, '0'
		6	r	an interrupt (with enabled mask bit) indicated in the INT_S2 register has occured
		7	r	FRAME interrupt with enabled mask bit has occured (any data channel interrupt) All masked B-, D- and PCM-channel interrupts are "ored" (see register INT_S1)

Reading the STATUS register clears no bit.

4.3 Auxiliary port registers

Name	Addr.	Bits	r/w	Function
P_ADR_W	1Eh	7..0	w	Port address write
P_DATA	1Fh	7..0	r/w	Port data

4.4 PCM/GCI/IOM2 bus section registers

Timeslots for transmit direction

Name	Addr.	Bits	r/w	Function
B1_SSL	20h	4..0	w	select PCM/GCI/IOM2 bus transmission slot (0..31, 32..63, 64..95, 96..127, see MST_MODE2 register bits 5..4)
B2_SSL	21h	5	w	unused
AUX1_SSL AUX2_SSL	22h 23h	6	w	select PCM/GCI/IOM2 bus data lines '0' STIO1 output '1' STIO2 output
		7	w	transmit channel enable for PCM/GCI/IOM2 bus '0' disable (reset default) '1' enable

☝ **important!**
Enabling more than one channel on the same slot causes undefined output data.

Timeslots for receive direction

Name	Addr.	Bits	r/w	Function
B1_RSL	24h	4..0	w	select PCM/GCI/IOM2 bus receive slot (0..31, 32..63, 64..95, 96..127, see MST_MODE2 register bits 5..4)
B2_RSL	25h	5	w	unused
AUX1_RSL AUX2_RSL	26h 27h	6	w	select PCM/GCI/IOM2 bus data lines '0' STIO2 is input '1' STIO1 is input
		7	w	receive channel enable for PCM/GCI/IOM2 bus '0' disable (reset default) '1' enable

Data registers

Name	Addr.	Bits	r/w	Function
B1_D *) B2_D *) AUX1_D *) AUX2_D *)	2Ch 2Dh 2Eh 2Fh	0..7	r/w	read/write data registers for selected timeslot data

*) These registers are read/written automatically by the HDLC FIFO controller (HFC) or PCM controller and need not be accessed by the user. To read/write data the FIFO registers should be used.

☝ **note!**
Auxiliary channel handling
To support an automatic CODEC to CODEC connection AUX1_D and AUX2_D can be set into mirror mode. In this case if the data registers AUX1_D and AUX2_D are not overwritten, the transmisson slots AUX1_SSL and AUX2_SSL mirror the data received in AUX1_RSL and AUX2_RSL slots. This is useful for an internal connection between two CODECs. This mirroring is enabled by setting bits 1..0 in MST_MODE1 register

Configuration and status registers

Name	Addr.	Bits	r/w	Function
MST_MODE0	14h	0	w	PCM/GCI/IOM2 bus mode '0' slave (reset default) (C4IO and F0IO are inputs) '1' master (C4IO and F0IO are outputs)
		1	w	polarity of C4- and C2O-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)
		5..4	w	time slot for CODEC-A signal F1_A '00' B1 receive slot '01' B2 receive slot '10' AUX1 receive slot '11' signal C2O → pin F1_A (C2O is 1/2 C4O clock)
		7..4		time slot for CODEC-B signal F1_B '00' B1 receive slot '01' B2 receive slot '10' AUX1 receive slot '11' AUX2 receive slot

The pulse shape and polarity of the CODEC signals F1_A and F1_B is the same as the pulseshape of the F0IO signal. The polarity of C2O can be changed by bit 1.

RESET sets register MST_MODE0, MST_MODE1 and MST_MODE2 to all '0's.

☝ **important!**
If no external clock source is connected to C4IO and F0IO bit 0 of MST_MODE0 must be set for normal operation.

Name	Addr.	Bits	r/w	Function
MST_MODE1	15h	0	w	enable/disable AUX1 channel mirroring '0' disable AUX1 channel data mirroring (reset default) '1' mirror AUX1 receive to AUX1 transmit
		1	w	enable/disable AUX2 channel mirroring '0' disable AUX2 channel data mirroring (reset default) '1' mirror AUX2 receive to AUX2 transmit
		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 (PCM30) '01' 4MBit/s (PCM64) '10' 8MBit/s (PCM128) '11' unused
		6	w	MST test loop When set MST output data is looped to the MST inputs.
		7	w	enable PCM/GCI/IOM2 write slots '0' disable PCM/GCI/IOM2 write slots; slot #2 and slot #3 may be used for normal data '1' enables slot #2 and slot #3 as master, D- and C/I-channel
MST_MODE2	16h	0	w	'1' generate frame signal for OKI™ CODECs on F1_A (see also Timing diagram 5: PCM/GCI/IOM2 timing on page 59)
		1	w	'1' generate frame signal for OKI™ CODECs on F1_B (see also Timing diagram 5: PCM/GCI/IOM2 timing on page 59)
		3..2	w	unused, must be '0'
		5..4	w	PCM/GCI/IOM2 slot select for higher data rates '00' slots 31..0 accessable '01' slots 63..32 accessable '10' slots 95..64 accessable '11' slots 127..96 accessable
		7..6	w	unused, must be '0'
F0_CNT_L	18h	7..0	r	F0IO pulse count 16 bit 125µs time counter (low byte)
F0_CNT_H	19h	7..0	r	F0IO pulse count 16 bit 125µs time counter (high byte)
C/I	28h	3..0	r/w	on read: indication on write: command
		7..4		unused

Name	Addr.	Bits	r/w	Function
TRxR	29h	0	r	'1' Monitor receiver ready (2 monitor bytes have been received)
		1	r	'1' Monitor transmitter ready Writing on MON2_D starts transmisssion and resets this bit.
		5..2	r	reserved
		6	r	STIO2 in
		7	r	STIO1 in
MON1_D	2Ah	7..0	r/w	first monitor byte
MON2_D	2Bh	7..0	r/w	second monitor byte

4.5 S/T section registers

Name	Addr.	Bits	r/w	Function
STATES (read)	30h	3..0	r	binary value of actual state (NT: Gx, TE: Fx)
		4	r	Frame-Sync ('1'=synchronized)
		5	r	'1' timer T2 expired (NT mode only, see also 8.1 S/T interface activation/deactivation layer 1 for finite state matrix for NT on page 73)
		6	r	'1' receiving INFO0
		7	r	'1' in NT mode: transition from G2 to G3 is allowed.
STATES (write)	30h	3..0	w	Set new state xxxx (bit 4 must also be set to load the state).
		4	w	'1' loads the prepared state (bit 3..0) and stops the state machine. This bit needs to be set for a minimum period of 5.21µs and must be cleared by software. (reset default) '0' enables the state machine. After writing an invalid state the state machine goes to deactivated state (G1, F2)
		6..5	w	'00' no operation '01' no operation '10' start deactivation '11' start activation The bits are automatically cleared after activation/deactivation.
		7	w	'0' no operation '1' in NT mode: allows transition from G2 to G3. This bit is automatically cleared after the transition.

☝ **important!**
The S/T state machine is stuck to '0' after a reset.
In this state the HFC-S USB sends no signal on the S/T-line and it is not possible to activate it by incoming INFOx.
Writing a '0' to bit 4 of the STATES register restarts the state machine.
NT mode: The NT state machine does not change automatically from G2 to G3 if the TE side sends INFO3 frames. This transition must be activated each time by bit 7 of the STATES register or by setting bit 0 of the SCTRL_E register.

Name	Addr.	Bits	r/w	Function
SCTRL	31h	0	w	'0' B1 send data disabled (permanent 1 sent in activated states, reset default) '1' B1 data enabled
		1	w	'0' B2 send data disabled (permanent 1 sent in activated states, reset default) '1' B2 data enabled
		2	w	S/T interface mode '0' TE mode (reset default) '1' NT mode
		3	w	D-channel priority '0' high priority 8/9 (reset default) '1' low priority 10/11
		4	w	S/Q bit transmission '0' S/Q bit disable (reset default) '1' S/Q bit and multiframe enable
		5	w	'0' normal operation (reset default) '1' send 96kHz transmit test signal (alternating zeros)
		6	w	TX_LO line setup This bit must be configured depending on the used S/T transformer module and circuitry to match the 400Ω pulse mask test. '0' capacitive line mode (reset default) '1' non capacitive line mode
		7	w	Power down '0' power up, oscillator active (reset default) '1' power down, oscillator stopped This bit is not cleared by a soft reset.
SCTRL_E	32h	0	w	force G2 → G3 automatic transition from G2 → G3 without setting bit 7 of STATES register
		1	w	must be '0'
		2	w	D reset '0' normal operation (reset default) '1' D bits are forced to '1'
		3	w	D_U enable '0' normal operation (reset default) '1' D channel is always send enabled regardless of E receive bit
		4	w	force E='0' (NT mode) '0' normal operation (reset default) '1' E-bit send is forced to '0'
		6..5	w	must be '0'
		7	w	'1' swap B1 and B2-channel in the S/T interface

Name	Addr.	Bits	r/w	Function
SCTRL_R	33h	0	w	B1-channel receive enable
		1	w	B2-channel receive enable '0' B-receive bits are forced to '1' '1' normal operation
		7..2	w	unused
SQ_REC	34h	3..0	r	TE mode: S bits (bit 3 = S1, bit 2 = S2, bit 1 = S3, bit 0 = S4) NT mode: Q bits (bit 3 = Q1, bit 2 = Q2, bit 1 = Q3, bit 0 = Q4)
		4	r	'1' a complete S or Q multiframe has been received Reading SQ_REC clears this bit.
		6..5	r	not defined
		7	r	'1' ready to send a new S or Q multiframe Writing to SQ_SEND clears this bit.
SQ_SEND	34h	3..0	w	TE mode: Q bits (bit 3 = Q1, bit 2 = Q2, bit 1 = Q3, bit 0 = Q4) NT mode: S bits (bit 3 = S1, bit 2 = S2, bit 1 = S3, bit 0 = S4)
		7..4	w	not defined
CLKDEL	37h	3..0	w	TE: 4 bit delay value to adjust the 2 bit time between receive and transmit direction (see also Figure 18). The delay of the external S/T-interface circuit can be compensated. The lower the value the smaller the delay between receive and transmit direction. NT: Data sample point. The lower the value the earlier the input data is sampled. The steps are 163ns.
		6..4	w	NT mode only early edge input data shaping Low pass characteristic of extended bus configurations can be compensated. The lower the value the earlier input data pulse is sampled. No compensation means a value of 6 (110b). Step size is the same as for bits 3-0.
		7	w	unused

☝ **note!**
The register is not initialized with a '0' after reset. The register should be initialized as follows before activating the TE/NT state machine:

TE mode: 0Dh .. 0Fh (0Fh for S/T interface circuitry on page 63)
NT mode: 6Ch

Name	Addr.	Bits	r/w	Function
B1_REC *)	3Ch	7..0	r	B1-channel receive register
B1_SEND *)	3Ch	7..0	w	B1-channel transmit register
B2_REC *)	3Dh	7..0	r	B2-channel receive register
B2_SEND *)	3Dh	7..0	w	B2-channel transmit register
D_REC *)	3Eh	7..0	r	D-channel receive register
D_SEND *)	3Eh	7..0	w	D-channel transmit register
E_REC *)	3Fh	7..0	r	E-channel receive register

*) These registers are read/written automatically by the HDLC FIFO controller (HFC) or PCM controller and need not be accessed by the user. To read/write data the FIFO registers should be used.

5 Electrical characteristics

Absolute maximum ratings

Parameter	Symbol	Rating
Supply voltage	V_{DD}	-0.3V to +7.0V
Input voltage	V_I	-0.3V to V_{CC} + 0.3V
Output voltage	V_O	-0.3V to V_{CC} + 0.3V
Operating temperature	T_{opr}	-10°C to +85°C
Storage temperature	T_{stg}	-40°C to +125°C

Recommended operating conditions

Parameter	Symbol	Condition	MIN.	TYP.	MAX.
Supply voltage	V_{DD}	V_{DD}=3.3V	3.0V	3.3V	3.6V
Operating temperature	T_{opr}		0°C		+70°C
Supply current normal power down	I_{DD}	f_{CLK}=24.576MHz; f_{CLKUSB}=48MHz V_{DD} = 3.3V, running oscillator: oscillator stopped:			

Electrical characteristics for 3.3V power supply

V_{DD} = 3.0V to 3.6V, T_{opr} = 0°C to +70°C

Parameter	Symbol	Condition	TTL level			CMOS level		
			MIN.	TYP.	MAX.	MIN.	TYP.	MAX.
Input LOW voltage	V_{IL}				0.8V			1.0V
Input HIGH voltage	V_{IH}		1.5V			2.0V		
Output LOW voltage	V_{OL}				0.4V			0.4V
Output HIGH voltage	V_{OH}		2.4V			2.4V		
Schmitt trigger, positive-going threshold	VT+				1.3V			2.0V
Schmitt trigger, negative-going threshold	VT-		0.5V			1.0V		

I/O Characteristics

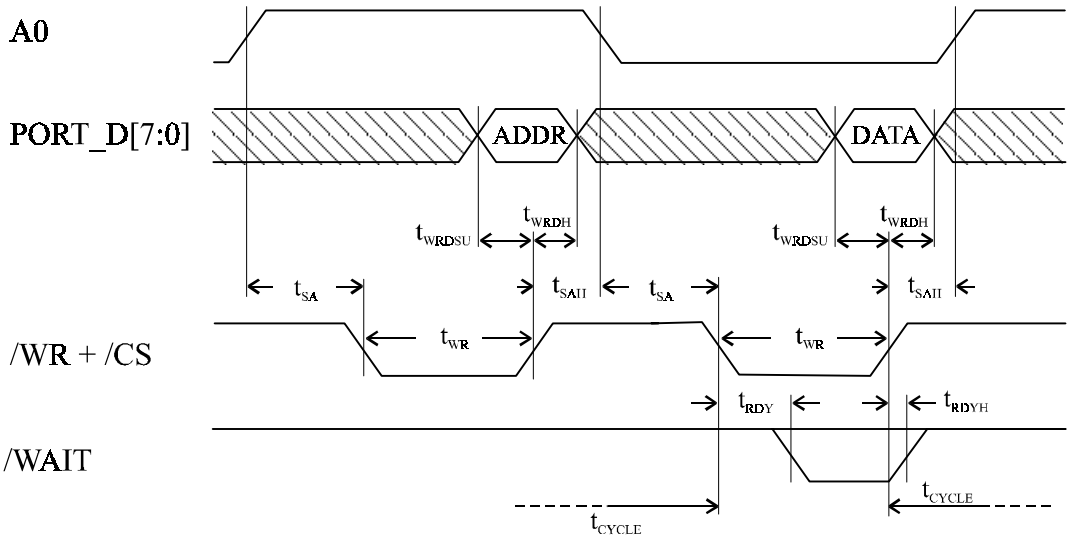
Input	Interface Level
/RD	CMOS
/WR	CMOS
PORT_D0-7	CMOS
CLKI	CMOS
AWAKE	CMOS
C4IO	TTL Schmitt Trigger, internal pull-up resistor
F0IO	CMOS, internal pull-up resistor
STIO1-2	CMOS, internal pull-up resistor
/WAIT MODE	CMOS, internal pull-up resistor
D+	USB Compliant Buffer
D-	USB Compliant Buffer
CLKUSBI	CMOS
SELF_PO	CMOS
EE_SDA	CMOS, internal pull-up resistor
EE_SCL	CMOS, internal pull-up resistor
A0	CMOS
/RES	CMOS Schmitt Trigger, internal pull-up resistor

	Driver Capability	
	Low	**High**
Output	**0.4V**	**V_{DD} - 0.8V**
/AUX_RD	4mA	2mA
/AUX_WR	4mA	2mA
/ADR_WR	4mA	2mA
PORT_D0-7	4mA	2mA
C4IO	8mA	4mA
F0IO	8mA	4mA
STIO1-2	8mA	4mA
F1_A-B	4mA	2mA
/WAIT	4mA	
/INT	4mA	
EE_SDA	1mA	
EE_SCL	1mA	

6 Timing characteristics

6.1 Microprocessor access

6.1.1 Register write access



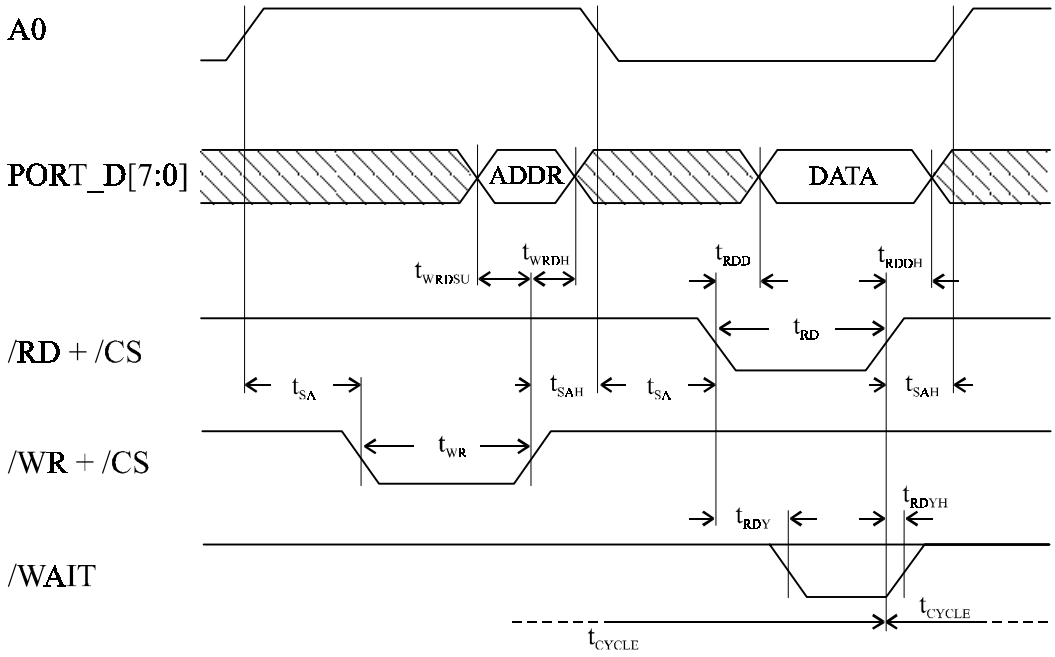
Timing diagram 1: Register write access

SYMBOL	CHARACTERISTICS	MIN.	MAX.
t_{SA}	Address to /WR Low Setup Time	20ns	–
t_{SAH}	Address Hold Time after /WR High	20ns	–
t_{WR}	Write Time	50ns	∞
t_{WRDSU}	Write Data Setup Time to /WR High	30ns	∞
t_{WRDH}	Write Data Hold Time from /WR High	10ns	–
t_{RDY}	Delay Time from /RD or /WR Low to /WAIT Low	3ns	30ns
t_{RDYH}	Delay Time from /RD High or /WR High to /WAIT High	3ns	30ns
t_{CYCLE}	End of Write Data Cycle to Start of Next Read/Write Data Cycle Time	6x t_{CLK}	∞

☝ **hint!**
If the same register as in the last register read/write access is accessed the register address write is not required.

t_{CLK} can be found in Timing diagram 3.

6.1.2 Register read access



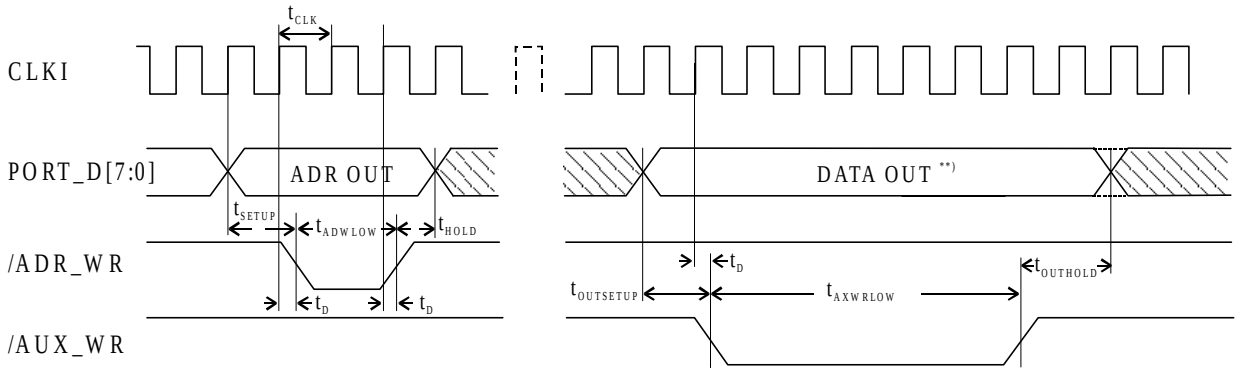
Timing diagram 2: Register read access

SYMBOL	CHARACTERISTICS	MIN.	MAX.
t_{RD}	Read Time	50ns	∞
t_{RDD}	/RD Low to Read Data Out Time	3ns	25ns
t_{RDDH}	/RD High to Data Buffer Turn Off Time	2ns	15ns
t_{SA}	Address to /RD or /WR Low Setup Time	20ns	–
t_{SAH}	Address Hold Time after /RD or /WR High	20ns	–
t_{WR}	Write Time	50ns	∞
t_{WRDSU}	Write Data Setup Time to /WR High	30ns	∞
t_{WRDH}	Write Data Hold Time from /WR High	10ns	–
t_{RDY}	Delay Time from /RD or /WR Low to /WAIT Low	3ns	30ns
t_{RDYH}	Delay Time from /RD High or /WR High to /WAIT High	3ns	30ns
t_{CYCLE}	End of Read Data Cycle to End of Next Read/Write Data Cycle Time	6x t_{CLK}	∞

☝ **hint!**
If the same register as in the last register read/write access is accessed the register address write is not required.

6.2 Auxiliary port access

6.2.1 Auxiliary port write access



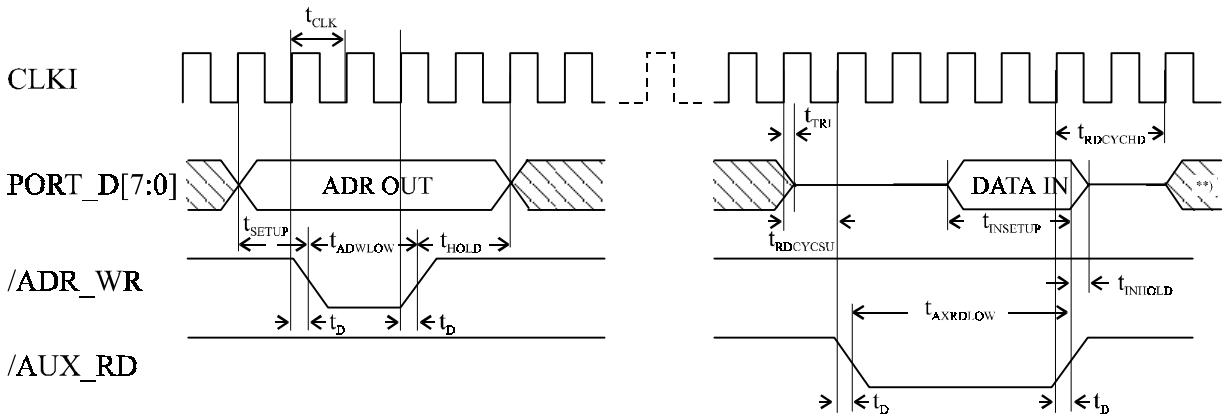
Timing diagram 3: Auxiliary port write access

SYMBOL	CHARACTERISTICS	MIN.	MAX.
t_{CLK}	Clock Period (24.576 MHz)	40.69 ns	
t_{SETUP}	Address Setup Time before /ADR_WR ↓	t_{CLK}	
t_{ADWLOW}	/ADR_WR Low Time	2x t_{CLK}	
t_{HOLD}	Address Hold Time after /ADR_WR ↑	t_{CLK}	
$t_{OUTSETUP}$	Data Out Setup Time before /AUX_WR ↓	t_{CLK}	
t_{AXWLOW}	/AUX_WR Low Time	*)	
$t_{OUTHOLD}$	Data Out Hold Time after /AUX_WR ↑	2x t_{CLK} **)	
t_D	Delay Time between CLKI ↑ and /ADR_WR or /AUX_WR		10 ns

*) configurable (see also: CIRM register bit description)

**) depending on the setting of bit 4 of the CIRM register data out can be valid until the next auxiliary port write access is initiated

6.2.2 Auxiliary port read access

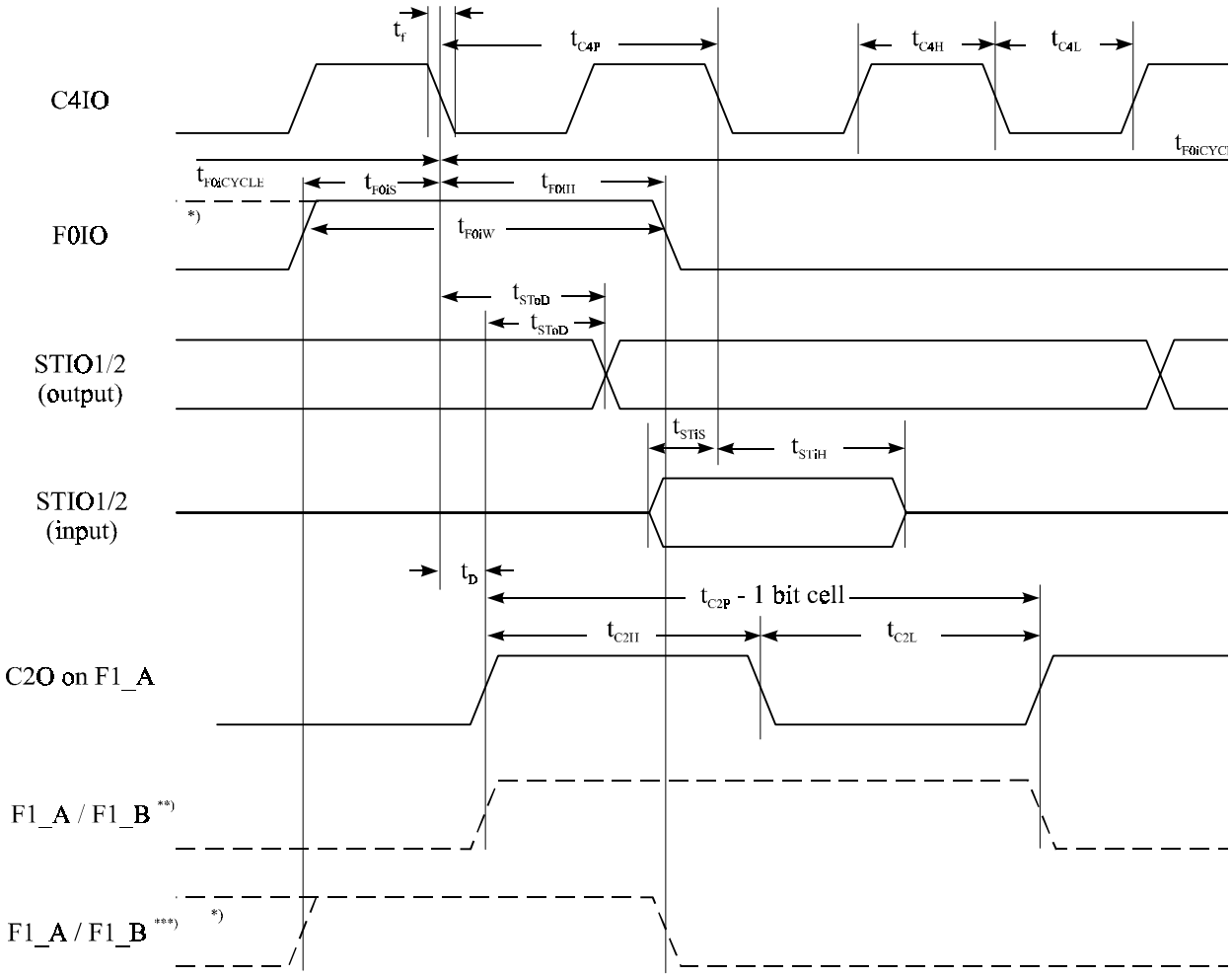


Timing diagram 4: Auxiliary port read access

SYMBOL	CHARACTERISTICS	MIN.	MAX.
t_{CLK}	Clock Period (24.576 MHz)	40.69 ns	
t_{SETUP}	Address Setup Time before /ADR_WR ↓	t_{CLK}	
t_{ADWLOW}	/ADR_WR Low Time	2x t_{CLK}	
t_{HOLD}	Address Hold Time after /ADR_WR ↑	t_{CLK}	
$t_{INSETUP}$	Minimum Data In Setup Time before /AUX_RD ↑	20 ns	
$t_{AXRDLOW}$	/AUX_RD Low Time	*)	
t_{INHOLD}	Data In Hold Time after /AUX_RD ↑	0 ns	
t_D	Delay Time between CLKI ↑ and /ADR_WR or /AUX_RD		10 ns
t_{TRI}	Time Data Floating after CLKI ↑		5 ns
$t_{RDCYCSU}$	Read Cycle Setup Time	t_{CLK}	
$t_{RDCYCHD}$	Output Data Valid after Read Cycle	20 ns **)	

*) configurable (see also: CIRM register bit description)
**) depending on the setting of bit 4 of the CIRM register

6.3 PCM/GCI/IOM2 timing



Timing diagram 5: PCM/GCI/IOM2 timing

*) F0IO starts one C4IO clock earlier if bit 3 in MST_MODE0 register is set. If this bit is set F0IO is also awaited one C4IO clock cycle earlier.

**) If bit 0 (or bit 1) of the MST_MODE2 register is set to '1' a frame signal for OKI™ CODECs is generated on F1_A (or F1_B). The C2O clock on F1_A is not available if bit 0 of the MST_MODE2 register is set.

***) If bit 0 (or bit 1) of the MST_MODE2 register is cleared to '0' F1_A (or F1_B) is a CODEC enable signal with the same pulse shape and timing as the F0IO signal.
If bits 5..4 of MST_MODE0 are '11' F1_A is C2O clock.

6.3.1 Master mode

To configure the HFC-S USB as PCM/GCI/IOM2 bus master bit 0 of the MST_MODE0 register must be set. In this case C4IO and F0IO are outputs.
The PCM bit rate is configured by bits 5..4 of the MST_MODE1 register.

SYMBOL	CHARACTERISTICS		MIN.	TYP.	MAX.
t_C	for 2Mb/s (PCM30)			122.07 ns	
	for 4Mb/s (PCM64)			61.035 ns	
	for 8Mb/s (PCM128)			30.518 ns	
t_{C4P}	Clock C4IO period *)		2 t_C - 26ns	2 t_C	2 t_C + 26ns
t_{C4H}	Clock C4IO High Width *)		t_C - 26ns	t_C	t_C + 26ns
t_{C4L}	Clock C4IO Low Width *)		t_C - 26ns	t_C	t_C + 26ns
t_{C2P}	Clock C2O Period		4 t_C - 52ns	4 t_C	4 t_C + 52ns
t_{C2H}	Clock C2O High Width		2 t_C - 26ns	2 t_C	2 t_C + 26ns
t_{C2L}	Clock C2O Low Width		2 t_C - 26ns	2 t_C	2 t_C + 26ns
t_{F0iW}	F0IO Width	Short F0IO	2 t_C - 6ns	2 t_C	2 t_C + 6ns
		Long F0IO	4 t_C - 6ns	4 t_C	4 t_C + 6ns
t_{STOD}	STIO1/2 Delay fom C4IO ↓ Level 1 Output			10 ns	25 ns
$t_{F0iCYCLE}$	F0IO Cycle Time	1 half clock adjust	124.975 us	125.000 us	125.025 us
		2 half clocks adjust	124.950 us	125.000 us	125.050 us
		3 half clocks adjust	124.925 us	125.000 us	125.075 us
		4 half clocks adjust	124.900 us	125.000 us	125.100 us

All specifications are for f_{CLK} = 24.576 MHz.

*) Time depends on accuracy of CLKI frequency. Because of clock adjustment in the 31st time slot these are the worst case timings when C4IO is adjusted.

6.3.2 Slave mode

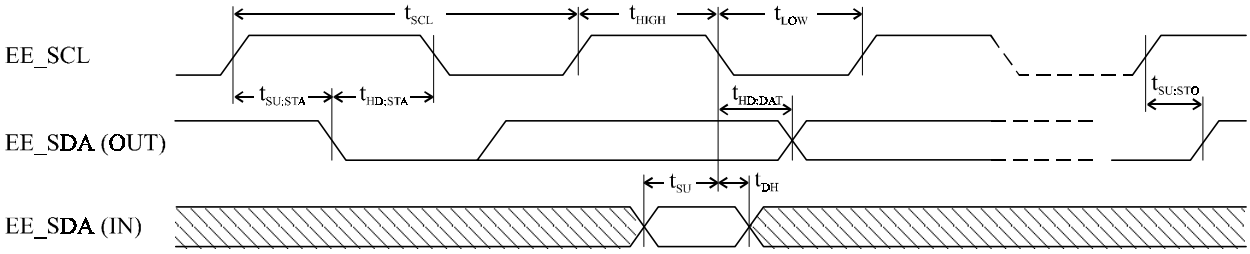
To configure the HFC-S USB as PCM/GCI/IOM2 bus slave bit 0 of the MST_MODE0 register must be cleared. In this case C4IO and F0IO are inputs.

SYMBOL	CHARACTERISTICS	MIN.	TYP.	MAX.
t_C	for 2Mb/s (PCM30)		122.07 ns	
	for 4Mb/s (PCM64)		61.035 ns	
	for 8Mb/s (PCM128)		30.518 ns	
t_{C4P}	Clock C4IO period *)		2 t_C	
t_{C4H}	Clock C4IO High Width	20 ns		
t_{C4L}	Clock C4IO Low Width	20 ns		
t_{C2P}	Clock C2O Period *)		4 t_C	
t_{C2H}	Clock C2O High Width	25 ns		
t_{C2L}	Clock C2O Low Width	25 ns		
t_{F0iS}	F0IO Setup Time to C4IO ↓	20 ns		
t_{F0iH}	F0IO Hold Time after C4IO ↓	20 ns		
t_{F0iW}	F0IO Width	40 ns		
t_{STiS}	STIO2 Setup Time	20 ns		
t_{STiH}	STIO2 Hold Time	20 ns		

All specifications are for f_{CLK} = 24.576 MHz.

*) If the S/T interface is synchronized from C4IO (NT mode) the frequency must be stable to ± 10^{-4}.

6.4 EEPROM access



Timing diagram 6: EEPROM access

SYMBOL	CHARACTERISTICS	TYP.
f_{SCL}	Serial Clock Frequency	93.75 KHz
t_{SCL}	Serial Clock Period	1 / f_{SCL}
$t_{HD:STA}$	Start Condition Hold Time	¾ t_{SCL}
t_{LOW}	Clock Low Period	½ t_{SCL}
t_{HIGH}	Clock High Period	½ t_{SCL}
$t_{SU:STA}$	Start Condition Setup Time	¾ t_{SCL}
$t_{HD:DAT}$	Output Data Change after Clock ↓	10 ns
t_{SU}	Data In Setup Time	100 ns
t_{DH}	Data In Hold Time	100 ns

7 External circuitries

7.1 S/T interface circuitry

In order to comply to the physical requirements of ITU-T recommendation I.430 and considering the national requirements concerning overvoltage protection and electromagnetic compatibility (EMC), the HFC-S USB needs some additional circuitry, which are shown in the following figures.

7.1.1 External receiver circuitry

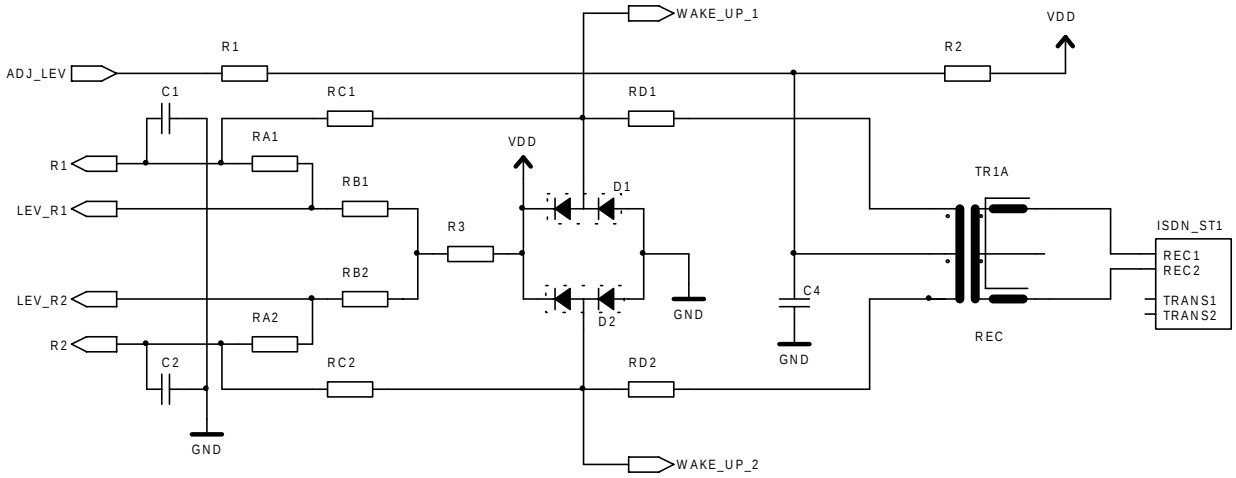


Figure 7: External receiver circuitry

WAKE_UP_1 and WAKE_UP_2 are for connection of the wake up circuitry (see: 7.1.2 External wake-up circuitry).
C1 and C2 are for reduction of high frequency input noise and should be placed as close as possible to the HFC-S USB.

Part list

Part	Value
C1	22p
C2	22p
C4	47n
D1	BAV99
D2	BAV99
RA1	100k
RA2	100k
RB1	33k
RB2	33k
RC1	4k7
RC2	4k7
RD1	4k7
RD2	4k7
R1	3k9
R2	1M
R3	1M8
ISDN_ST1	ISDN Connector
TR1	S/T transformer module (see Table 5 on page 67)

7.1.2 External wake-up circuitry

The wake-up circuitry is optional. It enables the HFC-S USB to wake up by incoming INFOx (non INFO0) signals on the S/T interface. If the wake-up circuitry is not used the AWAKE pin must be grounded.

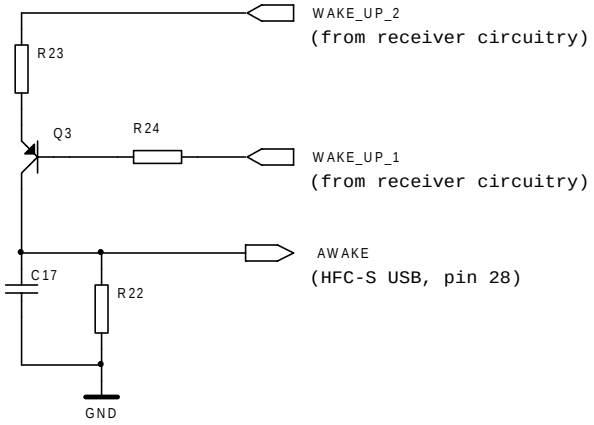


Figure 8: External wake-up circuitry

WAKE_UP_1 and WAKE_UP_2 are inputs from the receiver circuitry (see also: 7.1.1 External receiver circuitry).

Part List

Part	Value
C17	100pF
Q3	BC860C
R22	4M7
R23	10k
R24	100k

☝ **important!**
The remote wake-up feature must be enabled by the USB command SET FEATURE DEVICE_REMOTE_WAKEUP (see USB Specification 1.1, table 9-6). By default it is disabled.

7.1.3 External transmitter circuitry

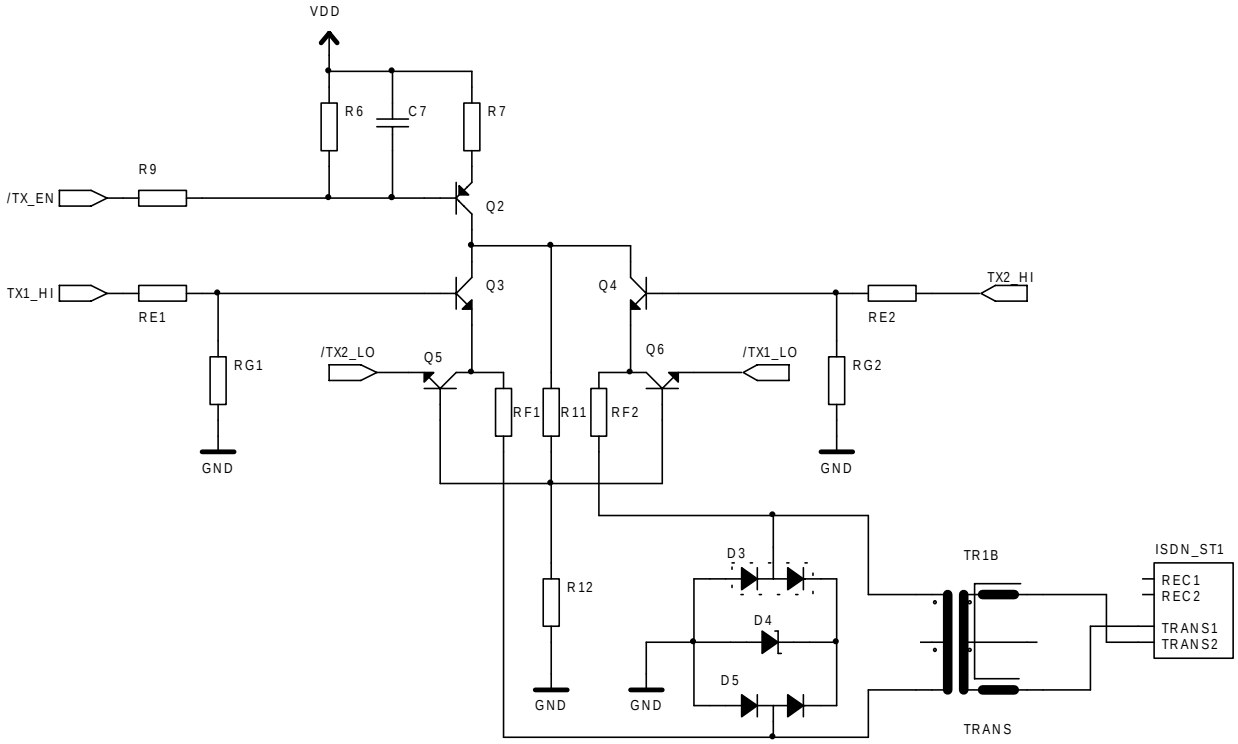


Figure 9: External transmitter circuitry

Part List

Part	Value
C7	470p
D3	BAV99
D5	BAV99
D4	2V7
Q2	BC860CL
Q3	BC850CL
Q4	BC850CL
Q5	BC850CL
Q6	BC850CL
RE1	560
RE2	560
RF1	12 *)

Part	Value
RF2	12 *)
RG1	4k7
RG2	4k7
R6	3k3
R7	50
R9	5k6
R11	1k8
R12	2k2
ISDN_ST1	ISDN Connector
TR1	S/T transformer module (see Table 5 on page 67)

*) value depends on the used S/T transformer module

Information from this list changes more frequently than the datasheets, so it might be out of date. We provide latest information about S/T transformers, modules and manufacturers on our web site www.colognechip.com.

S/T transformer module part number	manufacturer
APC 56624-1 APC 40495S (SMD) S-Hybrid modules with receiver and transmitter circuitry included: APC 5568-3V APC 5568-5V APC 5568DS-3V APC 5568DS-5V	**Advanced Power Components** *United Kingdom* Phone: +44 1634-290588 Fax: +44 1634-290591 http://www.apcisdn.com
FE 8131-55Z	**FEE GmbH** *Singapore* Phone: +65 741-5277 Fax: +65 741-3013 *Bangkok* Phone: +662 718-0726-30 Fax: +662 718-0712 *Germany* Phone: +49 6106-82980 Fax: +49 6106-829898
transformers: PE-64995 PE-64999 PE-65795 (SMD) PE-65799 (SMD) PE-68995 PE-68999 T5006 (SMD) T5007 (SMD) S_0-modules: T5012 T5034 T5038	**Pulse Engineering, Inc.** *United States* Phone: +1-619-674-8100 Fax: +1-619-674-8262 http://www.pulseeng.com
transformers: SM TC-9001 SM ST-9002 SM ST-16311F S_0-modules: SM TC-16311 SM TC-16311A	**Sun Myung** *Korea* Phone: +82-348-943-8525 Fax: +82-348-943-8527 http://www.sunmyung.com

S/T transformer module part number	manufacturer
transformers UT21023 S_0-modules: UT 20795 (SMD) UT 21624 UT 28624 A	**UMEC GmbH** *Germany* Phone: +49 7131-7617-0 Fax: +49 7131-7617-20 *Taiwan* Phone: +886-4-359-009-6 Fax: +886-4-359-012-9 *United States* Phone: +1-310-326-707-2 Fax: +1-310-326-705-8 http://www.umec.de
T 6040... transformers: 3-L4021-X066 3-L4025-X095 3-L5024-X028 3-L4096-X005 3-L5032-X040 S_0-modules: 7-L5026-X010 (SMD) 7-L5051-X014 7-M5051-X032 7-L5052-X102 (SMD) 7-M5052-X110 7-M5052-X114	**VAC GmbH** *Germany* Phone: +49 6181/ 38-0 Fax: +49 6181/ 38-2645 http://www.vacuumschmelze.de
transformers: ST5069 S_0-modules: PT5135 ST5201 ST5202	**Valor Electronics, Inc.** *Asia* Phone: +852 2333-0127 Fax: +852 2363-6206 *North America* Phone: +1 800 31VALOR Fax: +1 619 537-2525 *Europe* Phone: +44 1727-824-875 Fax: +44 1727-824-898 http://www.valorinc.com
543 76 009 00 503 740 010 0 (SMD)	**Vogt electronic AG** *Germany* Phone: +49 8591/ 17-0 Fax: +49 8591/ 17-240 http://www.vogt-electronic.com

Table 5: S/T transformer module part numbers and manufacturers

7.2 Oscillator circuitry for USB clock

7.2.1 Oscillator circuitry with coil

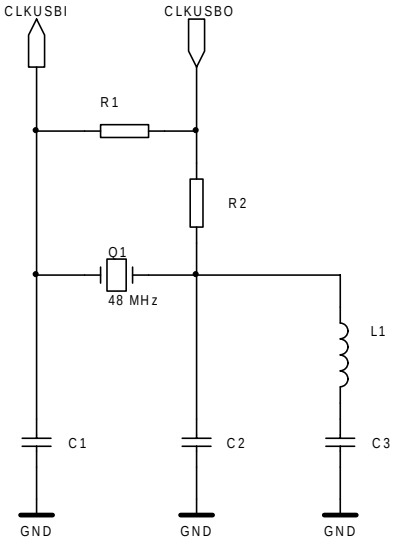


Figure 10: Oscillator circuitry for USB clock

Part List

Name	Value
R1	4.7 M
R2	22..56
C1	22 pF
C2	22 pF
C3	4.7 nF
L1	0.56 µH
Q1	48.000 MHz

L1 and C3 are only needed if Q1 is an overtone quartz.
The values of C1, C2, C3, L1 and R1, R2 depend on the used quartz.

7.2.2 Oscillator circuitry without coil

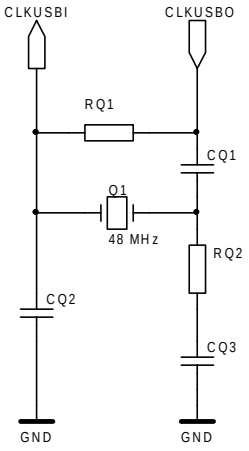


Figure 11: Oscillator circuitry for USB clock

Part List

Name	Value
RQ1	4.7 M
RQ2	680
CQ1	22 pF
CQ2	22 pF
CQ3	4.7 nF
Q1	48.000 MHz

The values of CQ1, CQ2, CQ3, RQ1, RQ2 depend on the used quartz.
This circuitry might not work with all quartzes. Use the circuitry with coil in this case (see above).

7.3 Oscillator circuitry for S/T clock

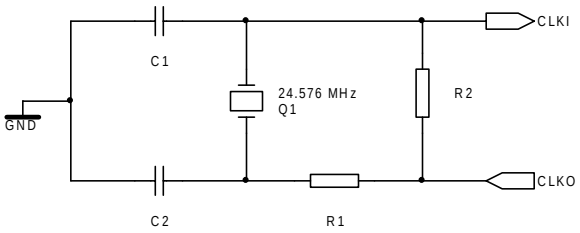


Figure 12: Oscillator circuitry for S/T clock

Part List

Name	Value
R1	330
R2	1 M
C1	47 pF
C2	47 pF
Q1	24.576 MHz quartz

The values of C1, C2 and R1, R2 depend on the used quartz.

For a load-free check of the oscillator frequency the C4O clock of the PCM/GCI/IOM2 bus should be measured (HFC-S USB as master, S/T interface deactivated, 4.096 MHz frequency intented on the C4IO).

7.4 EEPROM circuitry

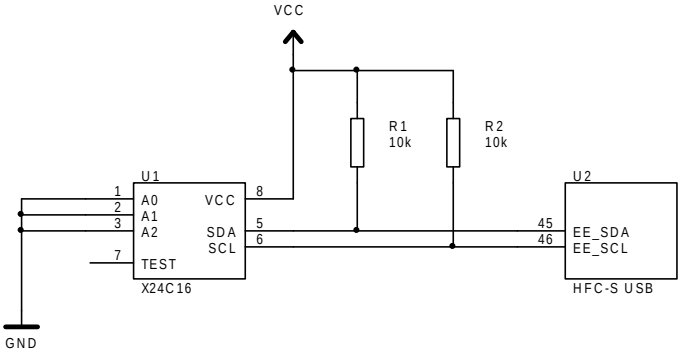


Figure 13: EEPROM circuitry

The external EEPROM is optional. To use the USB configuration data from the internal ROM EE_SCL must be connected to GND like shown in the circuitry below.

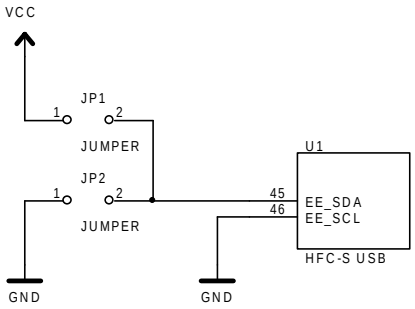


Figure 14: Circuitry to use USB configuration data from internal ROM

JP1 must be closed to select generic descriptors; JP2 must be closed to select communication class descriptors (see also: USB configuration data on page 18).

7.5 Auxiliary port circuitry

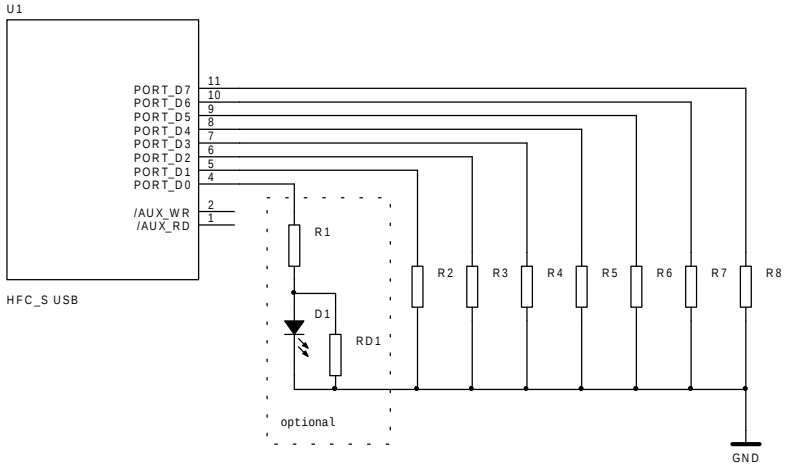


Figure 15: Auxiliary port circuitry

Part List

Name	Value
R1	1M
R2	1M
R3	1M
R4	1M
R5	1M
R6	1M
R7	1M
R8	1M
RD1	620
D1	LED
U1	HFC-S USB

The auxiliary port of the HFC-S USB needs some additional circuitry to reduce power consumption in Suspend Mode. To avoid floating inputs PORT_D[7:0] must be connected to GND over a resistor (R2). Of course this is only possible if the auxiliary port is not used. In the example above PORT_D0 is used for a LED. R1 is the usual resistor to limit the current through the LED. Additional a resistor parallel to the LED (RD1) is required to reach high input level at the auxiliary port pin when the bus is not driving out in Suspend Mode.

☝ **hint!**
To save some resitors all unused PORT_D outputs can be connected to ground over the same resistor. In this case the bits of the auxiliary port data register P_DATA corresponding to these PORT_D output pins must be set to the same value to avoid a short circuit.

7.6 Power supply from USB

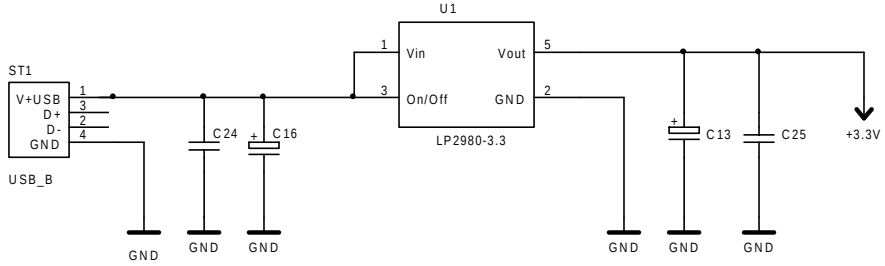


Figure 16: Power supply from USB

Part List

Name	Value
C13	1 uF
C16	2.2 uF
C24	33 nF
C25	33 nF
ST1	USB Connector
U1	LP2980-3.3

7.7 USB connection

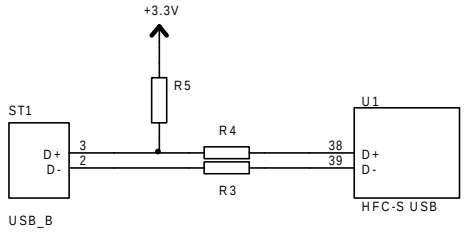


Figure 17: USB connection

Part List

Name	Value
R3	27
R4	27
R5	1k5
ST1	USB Connector
U1	HFC-S USB

8 State matrices for NT and TE

8.1 S/T interface activation/deactivation layer 1 for finite state matrix for NT

State name / State number / INFO sent / Event	Reset	Deactive	Pending activation	Active	Pending deactivation
State number	G0	G1	G2	G3	G4
INFO sent / Event	INFO 0	INFO 0	INFO 2	INFO 4	INFO 0
State machine release (Note 3)	G1	|	|	|	|
Activate request	G2 (Note 1)	G2 (Note 1)	|	|	G2 (Note 1)
Deactivate request	—	|	Start timer T2 G4	Start timer T2 G4	|
Expiry T2 (Note 2)	—	—	—	—	G1
Receiving INFO 0	—	—	—	G2	G1
Receiving INFO 1	—	G2 (Note 1)	—	/	—
Receiving INFO 3	—	/	G3 (Note 1) (Note 4)	—	—

Table 6: Activation/deactivation layer 1 for finite state matrix for NT

— No state change
/ Impossible by the definition of peer-to-peer physical layer procedures or system internal reasons
| Impossible by the definition of the physical layer service

Note 1: Timer 1 (T1) is not implemented in the HFC-S USB and must be implemented in software.

Note 2: Timer 2 (T2) prevents unintentional reactivation. Its value is 32ms (256 x 125µs). This implies that a TE has to recognize INFO 0 and to react on it within this time.

Note 3: After reset the state machine is fixed to G0.

Note 4: Bit 7 of the STATES register must be set to allow this transition.

8.2 Activation/deactivation layer 1 for finite state matrix for TE

Event \ State name		Reset	Sensing	Deactivated	Awaiting signal	Identifying input	Synchronized	Activated	Lost framing
State number		F0	F2	F3	F4	F5	F6	F7	F8
Info sent		INFO 0	INFO 0	INFO 0	INFO 1	INFO 0	INFO 3	INFO 3	INFO 0
State machine release (Note 1)		F2	/	/	/	/	/	/	/
Activate Request	Receiving any signal	—	|	F5	|	|	—	|	—
	Receiving INFO 0	—	|	F4	|	|	—	|	—
Expiry T3 (Note 5)		—	/	—	F3	F3	F3	—	—
Receiving INFO 0		—	F3	—	—	—	F3	F3	F3
Receiving any signal (Note 2)		—	—	—	F5	—	/	/	—
Receiving INFO 2 (Note 3)		—	F6	F6	F6	F6	—	F6	F6
Receiving INFO 4 (Note 3)		—	F7	F7	F7	F7	F7	—	F7
Lost framing (Note 4)		—	/	/	/	/	F8	F8	—

Table 7: Activation/deactivation layer 1 for finite state matrix for TE

— No change, no action
| Impossible by the definition of the layer 1 service
/ Impossible situation

Notes

Note 1: After reset the state machine is fixed to F0.

Note 2: This event reflects the case where a signal is received and the TE has not (yet) determined wether it is INFO 2 or INFO 4.

Note 3: Bit- and frame-synchronisation achieved.

Note 4: Loss of Bit- or frame-synchronisation.

Note 5: Timer 3 (T3) is not implemented in the HFC-S USB and must be implemented in software.

9 Binary organisation of the frames

9.1 S/T frame structure

The frame structures on the S/T interface are different for each direction of transmission. Both structures are illustrated in Figure 18.

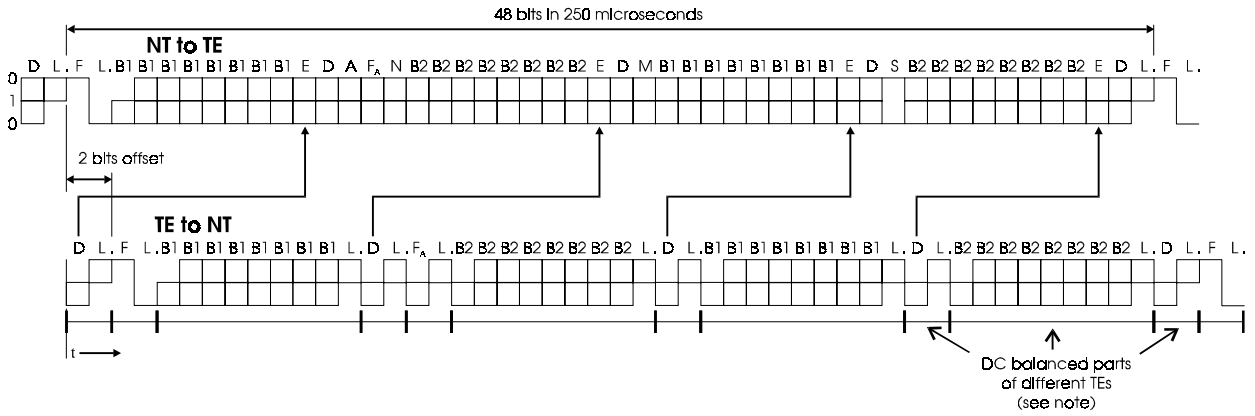


Figure 18: Frame structure at reference point S and T

F	Framing bit	N	Bit set to a binary value N = $\overline{F}_A$ (NT to TE)
L	D.C. balancing bit	B1	Bit within B-channel 1
D	D-channel bit	B2	Bit within B-channel 2
E	D-echo-channel bit	A	Bit used for activation
F_A	Auxiliary framing bit	S	S-channel bit
M	Multiframing bit		

☝ **note!**
Lines demarcate those parts of the frame that are independently d.c.-balanced.
The F_A bit in the direction TE to NT is used as Q bit in every fifth frame if S/Q bit transmission is enabled (see SCTRL register).
The nominal 2-bit offset is as seen from the TE. The offset can be adjusted with the CLKDEL register in TE mode. The corresponding offset at the NT may be greater due to delay in the interface cable and varies by configuration.
HDLC-B-channel data start with the LSB, PCM-B-channel data start with the MSB.

9.2 GCI frame structure

The binary organisation of a single GCI channel frame is described below. C4IO clock frequency is 4096kHz.

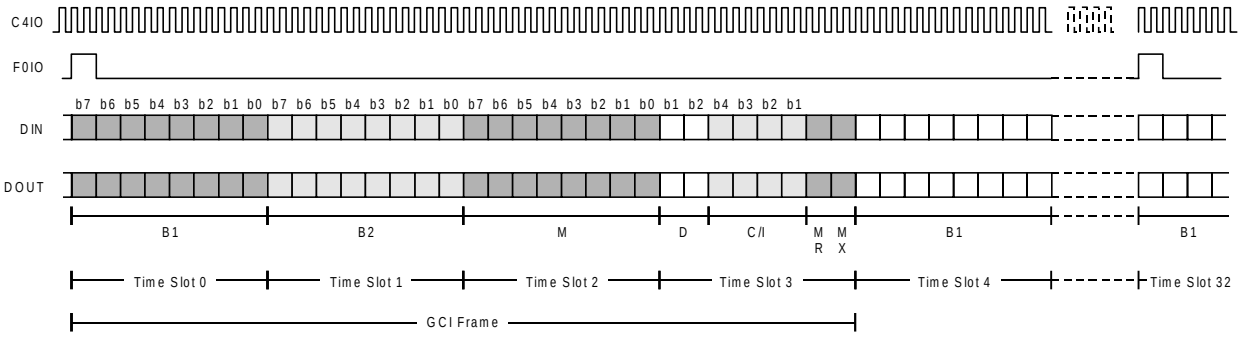


Figure 19: Single channel GCI format

B1 B-channel 1 data
B2 B-channel 2 data
M Monitor channel data
D D-channel data
C/I Command/indication bits for controlling activation/deactivation and for additional control functions
MR Handshake bit for monitor channel
MX Handshake bit for monitor channel

10 Clock synchronisation

10.1 Clock synchronisation in NT-mode

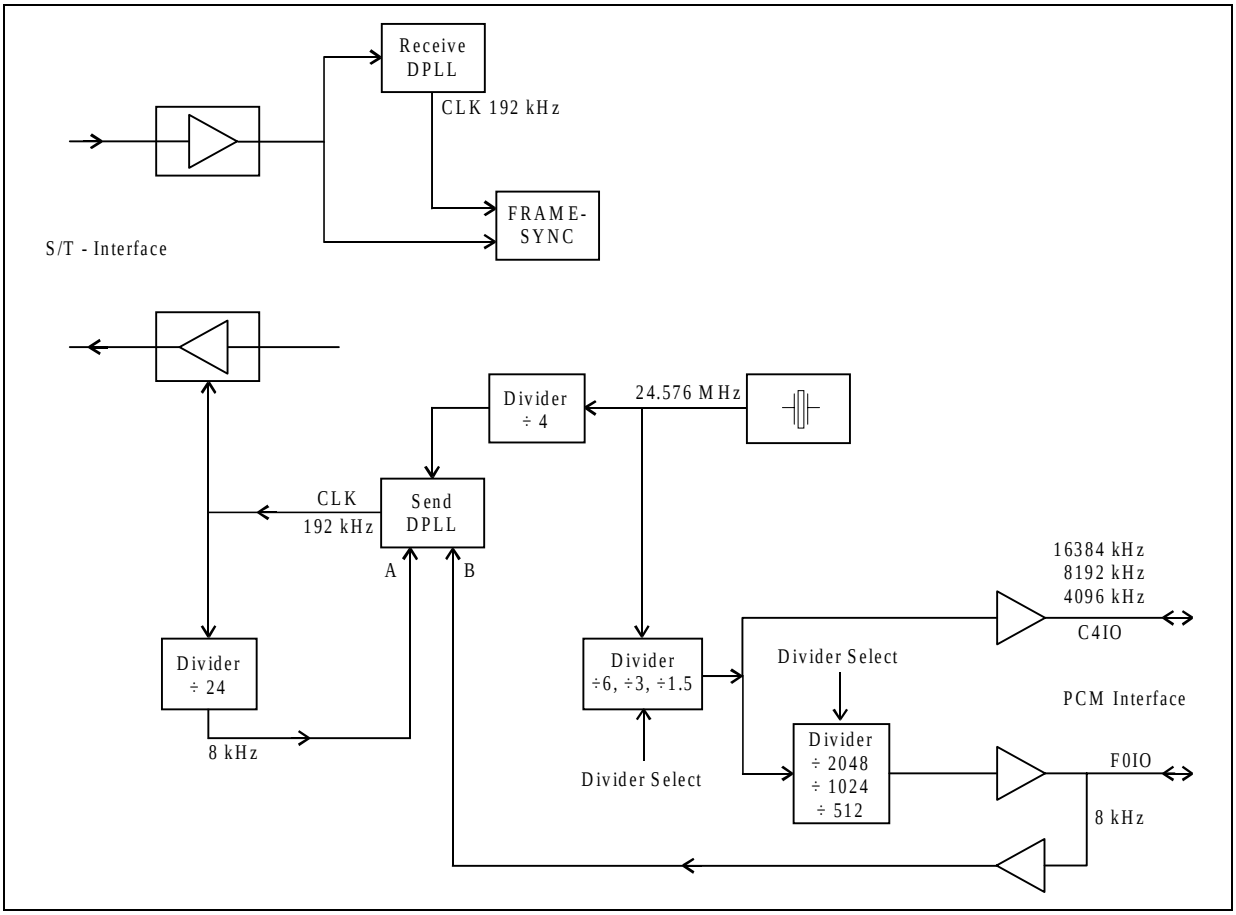


Figure 20: Clock synchronisation in NT-mode

10.2 Clock synchronisation in TE-mode

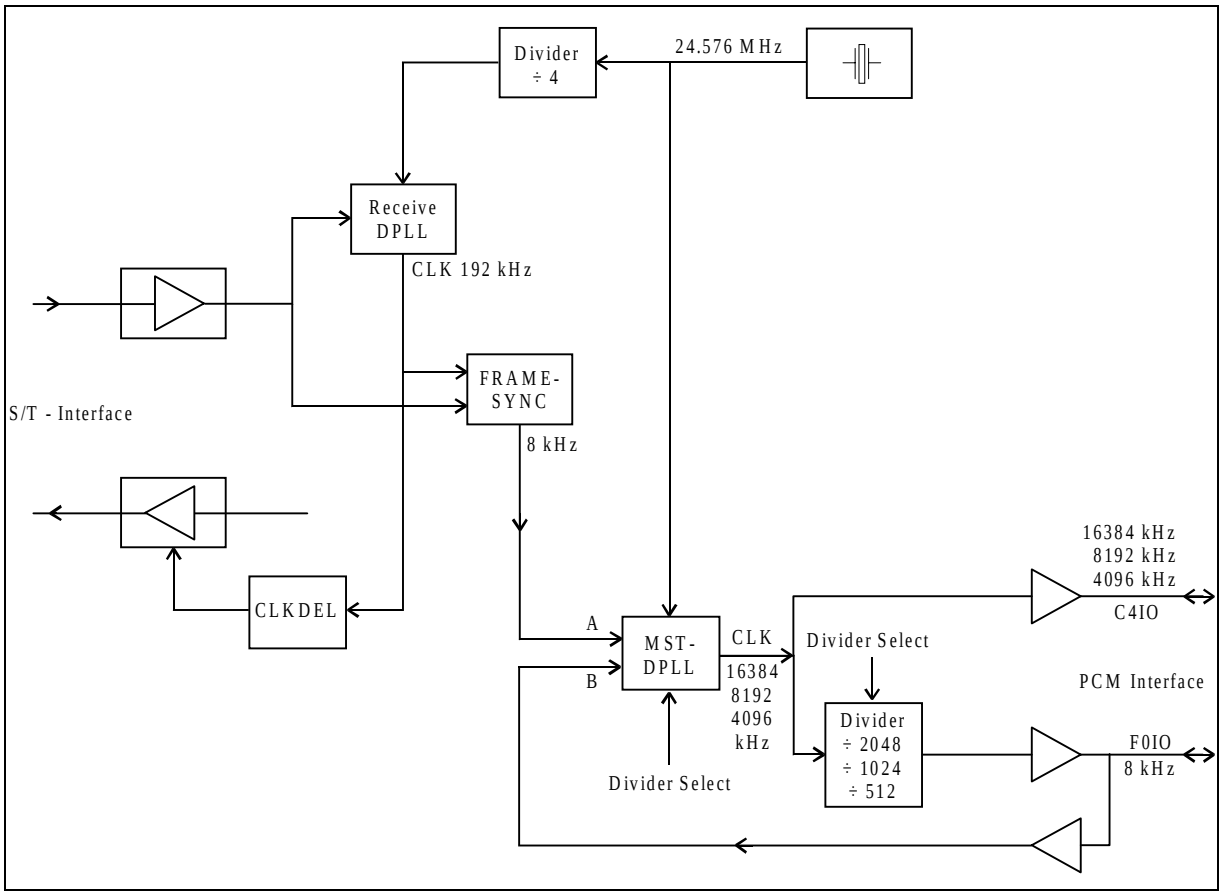


Figure 21: Clock synchronisation in TE-mode

The C4IO clock is adjusted in the 31th time slot at the GCI/IOM bus 1..4 times for one half clock cycle. This can be reduced to one adjustment of a half clock cycle (see MST_MODE1 register). This is useful if another HFC series ISDN controller is connected as slave in NT mode to the PCM bus.

11 HFC-S USB package dimensions

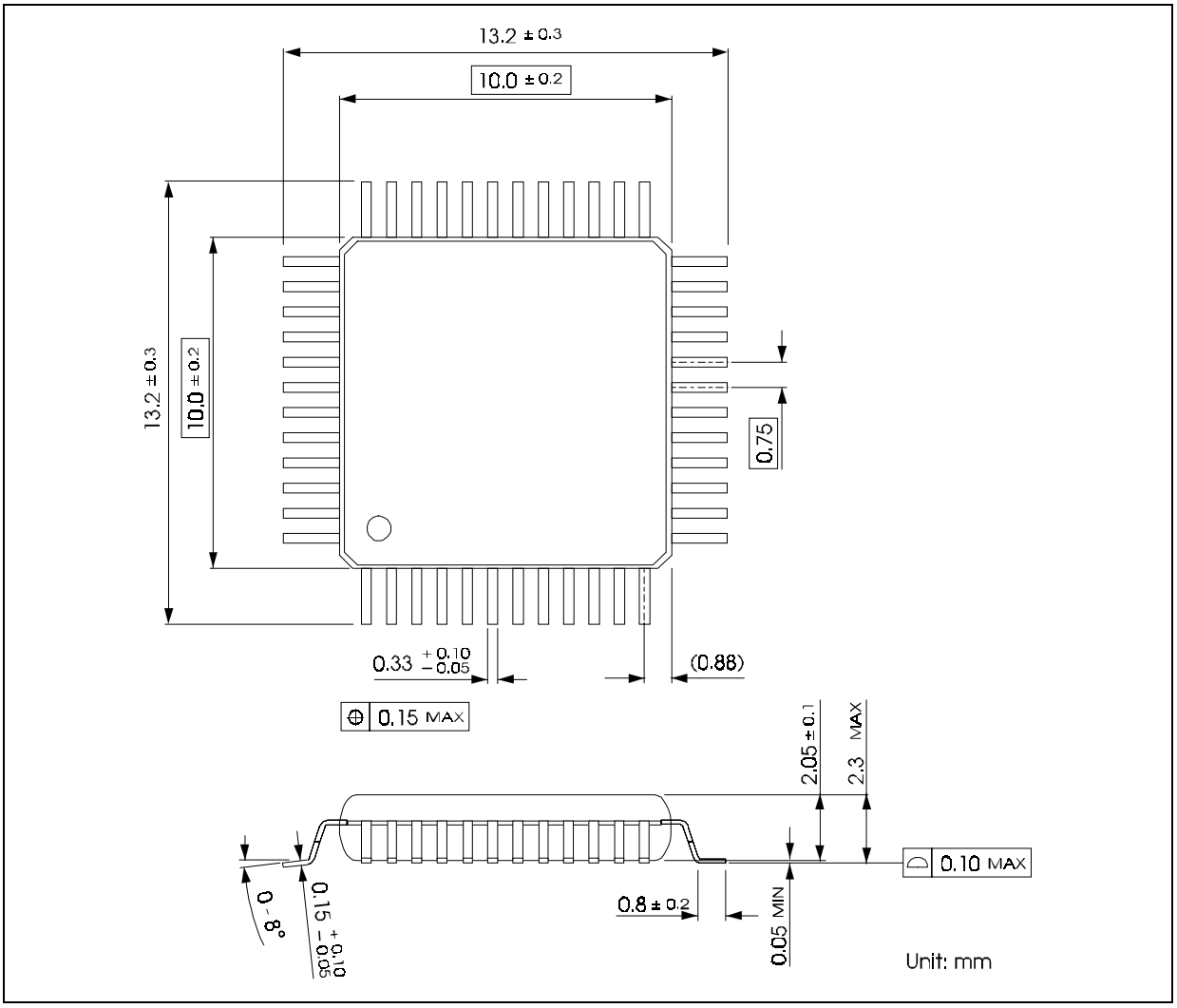
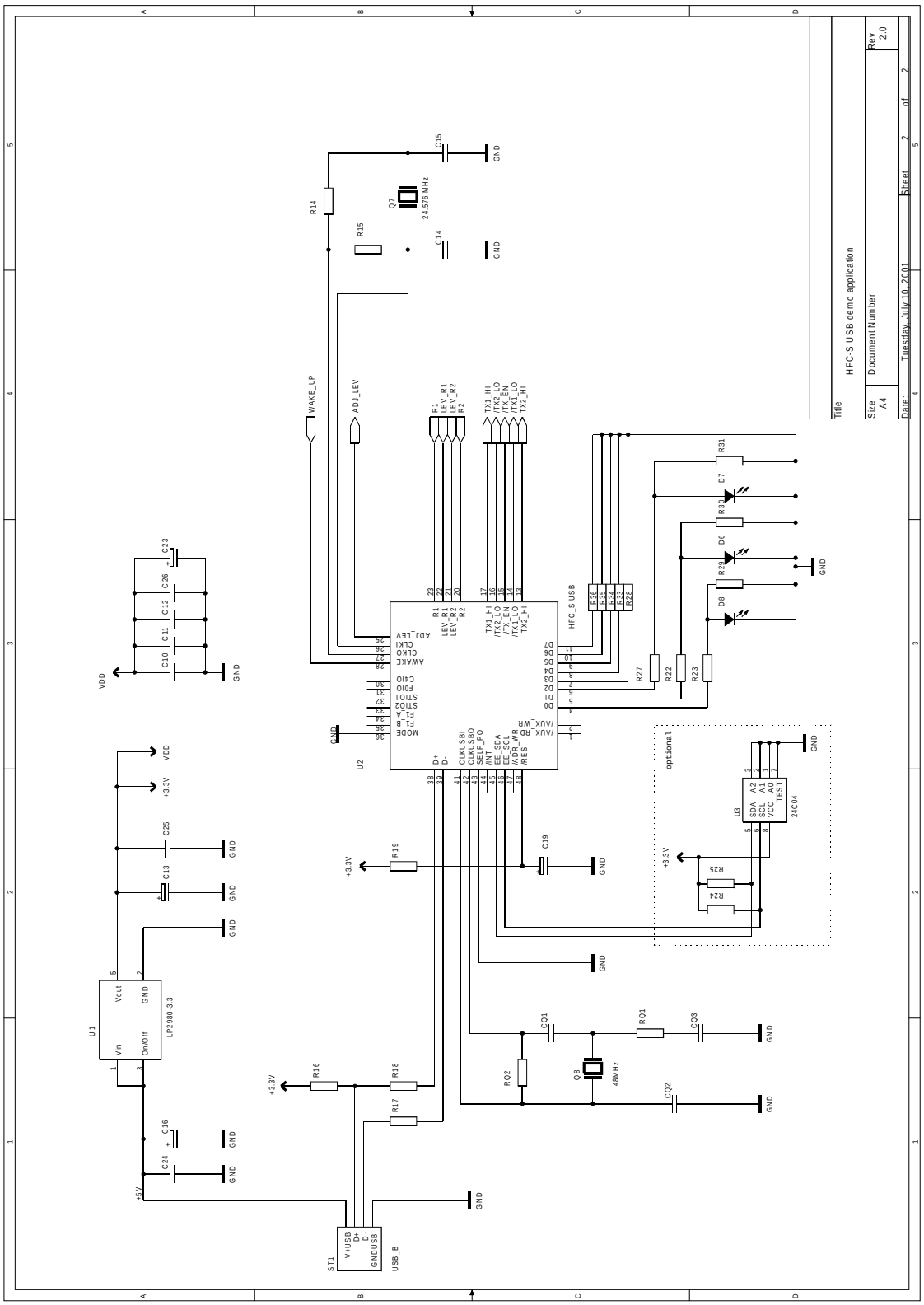
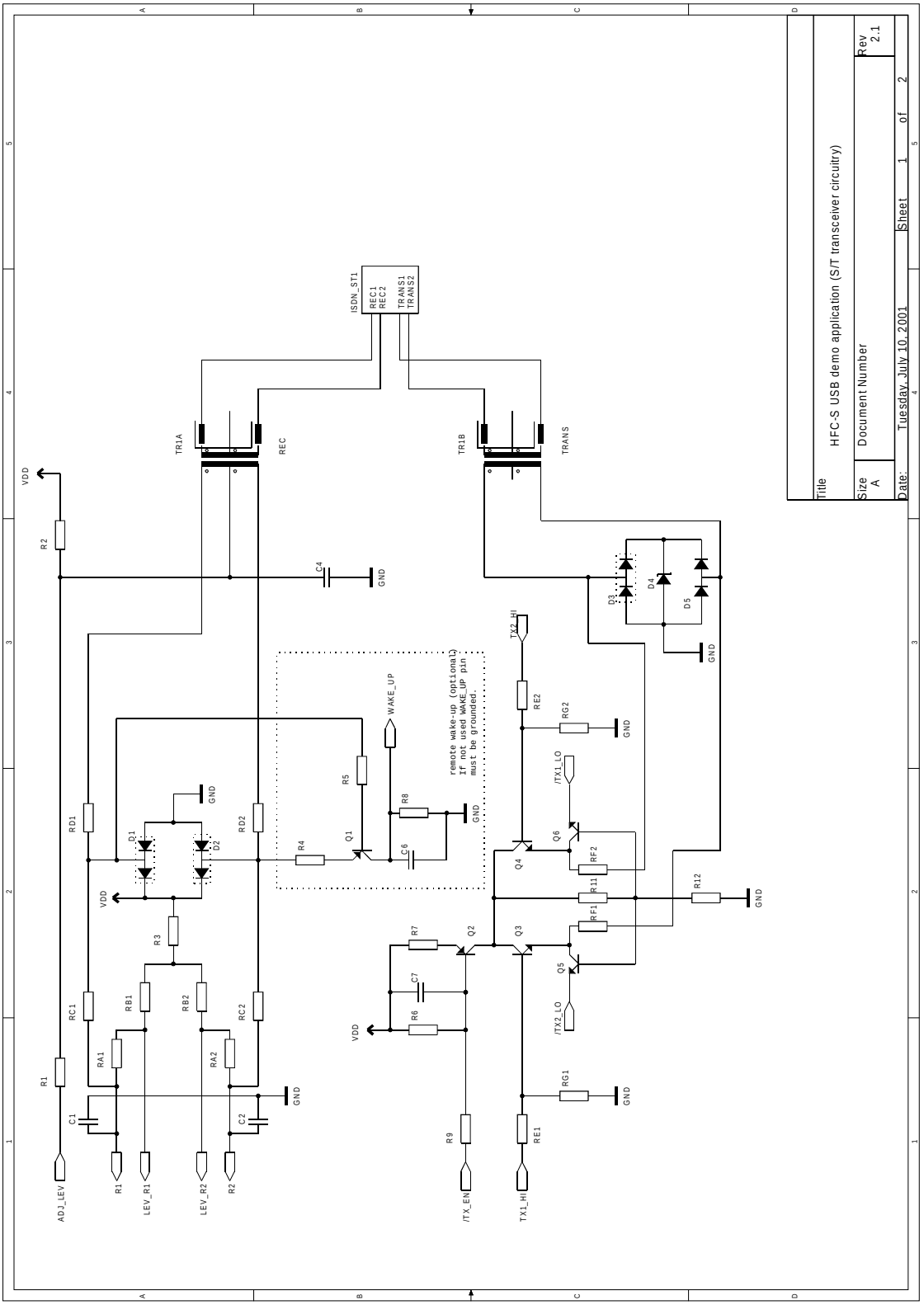


Figure 22: HFC-S USB package dimensions

12 ISDN USB TA sample circuitry with HFC-S USB




HFC-S USB demo application (S/T transceiver circuitry)
Document Number
Rev 2.1
Size A
Date: Tuesday, July 10, 2001
Sheet 1 of 2
ISDN_ST1
REC1
REC2
TRANS1
TRANS2
TR1A
REC
TR1B
TRANS
VDD
GND
WAKE_UP
remote wake-up (optional)
If not used WAKE_UP pin
must be grounded.
TX2_HI
/TX1_LO
/TX2_LO
/TX_EN
TX1_HI
ADJ_LEV
LEV_R1
LEV_R2

Part List

Name	Value
C1	22p
C2	22p
C4	47n
C6	100p (optional; for wake-up)
C7	470p
C10	100n
C11	100n
C12	100n
C19	1µ
C13	1µ
C15	47p
C14	47p
C16	2µ2
C23	10µ
C24	33n
C25	33n
C26	33n
CQ1	22p
CQ2	22p
CQ3	4n7
D1	BAV99
D2	BAV99
D3	BAV99
D5	BAV99
D4	2V7
D6	LED (optional)
D7	LED (optional)
D8	LED (optional)
ISDN_ST1	ISDN Connector
Q1	BC860CL (optional; for wake-up)
Q2	BC860CL
Q3	BC850CL
Q4	BC850CL
Q5	BC850CL
Q6	BC850CL
Q7	24.576 MHz
Q8	48MHz
RA1	100k
RA2	100k
R29	100k (optional; for LED D8)
R30	100k (optional; for LED D6)
R31	100k (optional; for LED D7)
RB1	33k
RB2	33k
RC1	4k7
RC2	4k7
RD1	4k7
RD2	4k7
RE1	560
RE2	560
RF1	12
RF2	12
RG1	4k7
RG2	4k7
RQ1	680
RQ2	4M7
R1	3k9
R2	1M
R3	1M8
R4	10k
R5	100k
R6	3k3
R7	50
R8	4M7
R9	5k6
R11	1k8
R12	2k2
R14	330
R15	1M
R16	1k5
R17	27
R18	27
R19	n.b.
R22	620 (optional; for LED D6)
R23	620 (optional; for LED D8)
R24	10k (optional; for EEPROM)
R25	10k (optional; for EEPROM)
R27	620 (optional; for LED D7)
R28	1M
R33	1M
R34	1M
R35	1M
R36	1M
ST1	USB Connector
TR1	S/T transformer module
U1	LP2980-3.3
U2	HFC-S USB
U3	24C04 (optional)