Data Sheet, V0.3, Sep. 2003

TC1130

32-Bit Single-Chip Microcontroller

Preliminary

32bit

Microcontrollers

Infineon technologies

Never stop thinking.

Edition 2003-09

Published by Infineon Technologies AG,
St.-Martin-Strasse 53,
D-81541 München, Germany

Attention please!

The information herein is given to describe certain components and shall not be considered as warranted characteristics.

Terms of delivery and rights to technical change reserved.

We hereby disclaim any and all warranties, including but not limited to warranties of non-infringement, regarding circuits, descriptions and charts stated herein.

Infineon Technologies is an approved CECC manufacturer.

Information

For further information on technology, delivery terms and conditions and prices please contact your nearest Infineon Technologies Office in Germany or our Infineon Technologies Representatives worldwide (see address list).

Warnings

Due to technical requirements components may contain dangerous substances. For information on the types in question please contact your nearest Infineon Technologies Office.

Infineon Technologies Components may only be used in life-support devices or systems with the express written approval of Infineon Technologies, if a failure of such components can reasonably be expected to cause the failure of that life-support device or system, or to affect the safety or effectiveness of that device or system. Life support devices or systems are intended to be implanted in the human body, or to support and/or maintain and sustain and/or protect human life. If they fail, it is reasonable to assume that the health of the user or other persons may be endangered.

TC1130

32-Bit Single-Chip Microcontroller

Microcontrollers

Never stop thinking.

TC1130
Advance Information
Revision History: 2003-09 V0.3

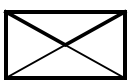
Previous Version:

Page	Subjects (major changes since last revision)

We Listen to Your Comments

Any information within this document that you feel is wrong, unclear or missing at all? Your feedback will help us to continuously improve the quality of this document. Please send your proposal (including a reference to this document) to:

mcdocu.comments@infineon.com



32-Bit Single-Chip Microcontroller **TC1130**
TriCore Family

Advance Information

- High Performance 32-bit TriCore V1.3 CPU with 4-Stage Pipeline
- Floating Point Unit (FPU)
- Dual Issue super-scalar implementation
  - MAC Instruction maximum triple issue
- Circular Buffer and bit-reverse addressing modes for DSP algorithms
- Flexible multi-master interrupt system
- Very fast interrupt response time
- Hardware controlled context switch for task switch and interrupts
- Memory Management Unit (MMU)
- On-chip Memory
  - 32 KByte Data Memory (SPRAM)
  - 32 KByte Code Memory (SPRAM)
  - 16 KByte Instruction Cache (ICACHE).
  - 64KByte SRAM Data Memory Unit (DMU)
  - 16 KByte Boot ROM
- On-chip Bus Systems
  - 64-Bit High Performance Local Memory Bus (LMB) for fast access between caches and on-local memories and FPI Interface
  - On-chip Flexible Peripheral Interconnect Buses (FPI) for interconnections of functional units
- DMA Controller with 8 channels for data transfer operations between peripheral units and memory locations
  - Two high speed Micro Link Interfaces (MLI0/1) for controller communication and emulation
- Flexible External Bus Interface Unit (EBU) to access external data memories
- One Multifunctional General Purpose Timer Units (GPTU) with three 32-bit timer/ counters
- Two Capture and Compare units (CCU60/1) for PWM signal generation, each with
  - 3-channel, 16 bit Capture and Compare unit
  - 1-channel, 16 bit Compare unit
- Three Asynchronous/Synchronous Serial Channels (ASC0/1/2) with baudrate generator, parity, framing and overrun error detection, support FIFO and IrDA data transmission
- Two High Speed Synchronous Serial Channels (SSC0/1) with programmable data length, FIFO support and shift direction
- One MultiCAN Module with four CAN nodes and 64 message buffers for high efficiency data handling
- Fast Ethernet Controller with 10/100 Mbps MII-Based physical devices support

- USB module with compliance to USB Specification Revision 1.1, with support for 1.5MBaud to 12MBaud devices
- Inter-IC (IIC) module with two physical IIC buses
- Digital I/O ports with 3.3V IO capabilities
- Level 2 On-chip Debug Support
- Power Management System
- Clock Generation Unit with PLL
- Maximum CPU and Bus clock frequency at 150MHz without MMU and 120MHz with MMU
- Ambient temperature under bias: -40° to +85°C
- P-LBGA-208 package

Block Diagram

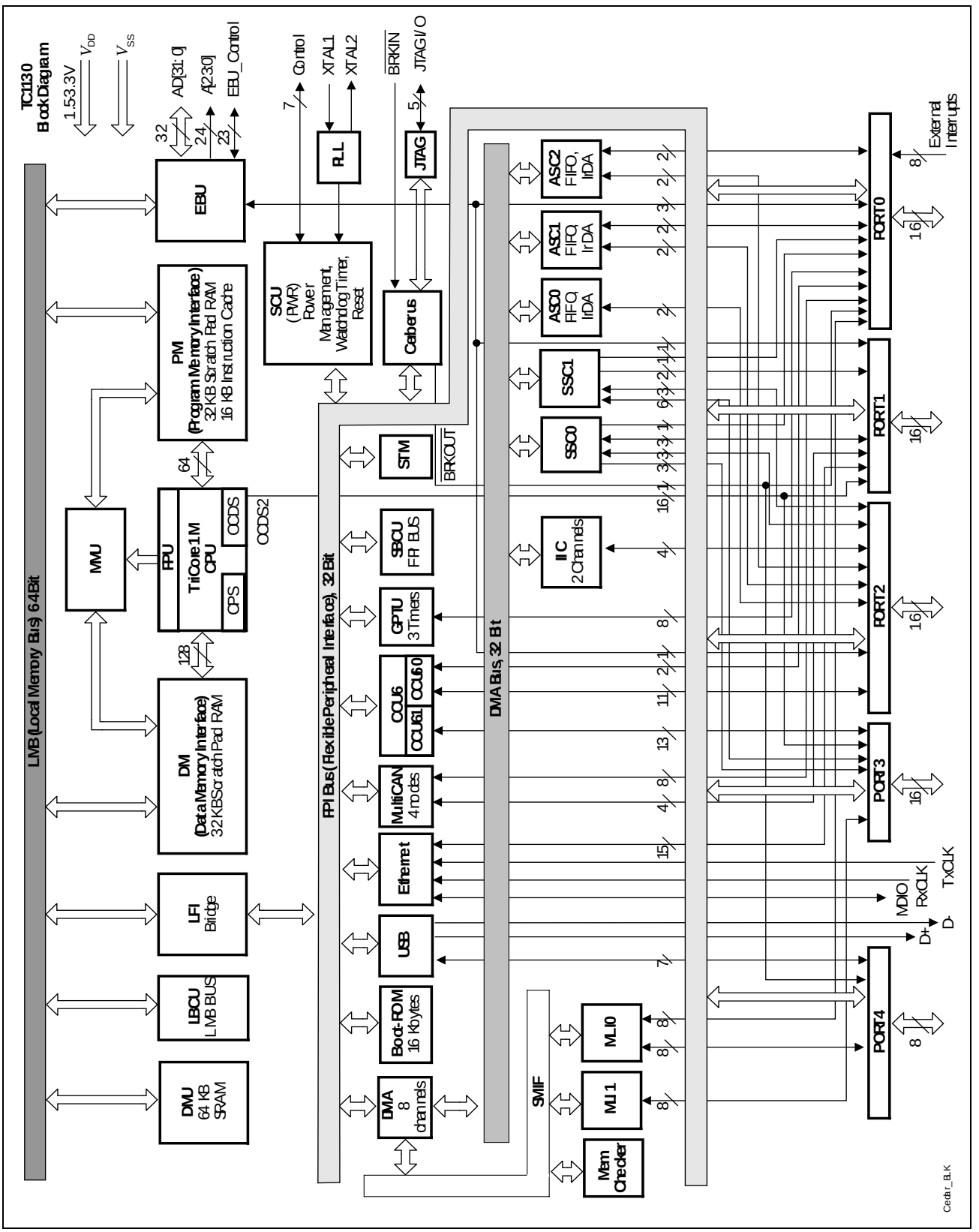


Figure 1 TC1130 Block Diagram

Logic Symbol

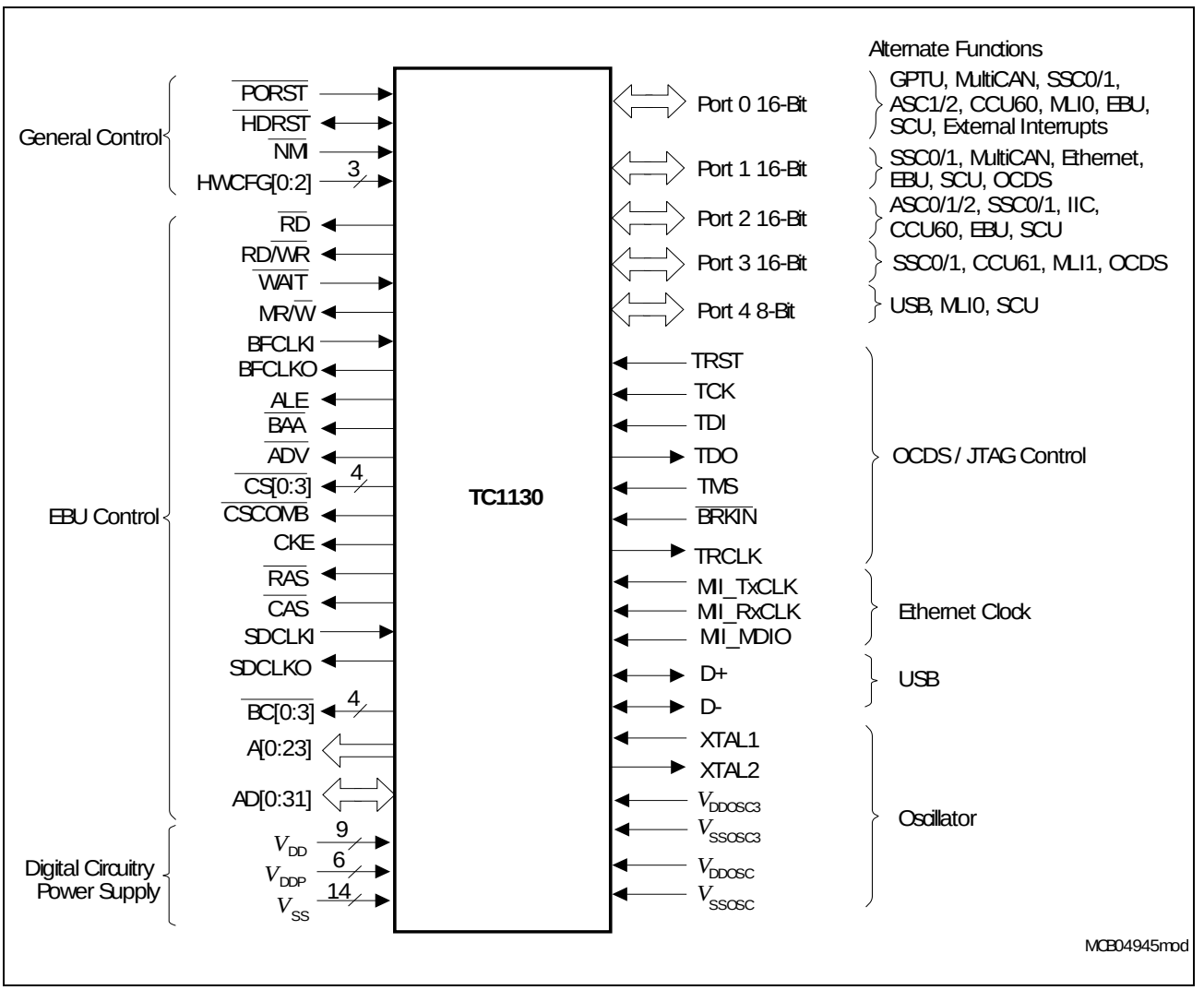


Figure 2 TC1130 Logic Symbol

Pin Configuration

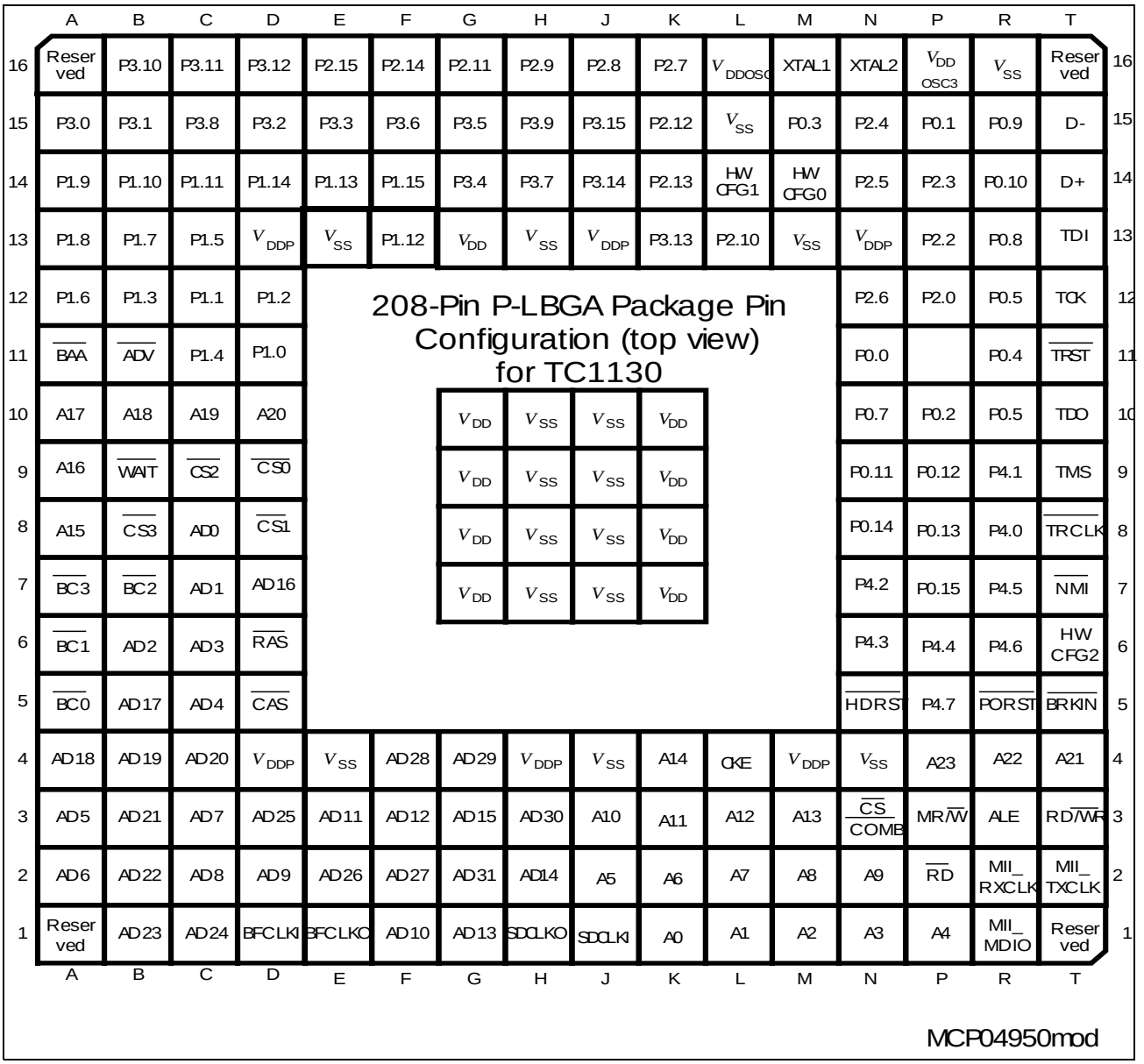


Figure 3 TC1130 Pinning: P-BGA-208 Package (top view)

Table 1 Pin Definitions and Functions

Symbol	Pin	In Out	PU/ PD[1)]	Functions	
P0		I/O		**Port 0** Port 0 is a 16-bit bidirectional general purpose I/O port which can be alternatively used for GPTU, MultiCAN, ASC1/2, SSC0/1, MLI0, EBU and SCU.	
P0.0	N11	I/O	PUC	GPTU_0	GPTU input/output line 0
		I/O		RXD1B	ASC1 receiver input/output B
P0.1	P15	I/O	PUC	GPTU_1	GPTU input/output line 1
		O		TXD1B	ASC1 transmitter output B
P0.2	P19	I/O	PUC	GPTU_2	GPTU input/output line 2
		I/O		RXD2B	ASC2 receiver input/output B
P0.3	M15	I/O	PUC	GPTU_3	GPTU input/output line 3
		O		TXD2B	ASC2 transmitter output B
P0.4	R11	I/O	PUC	GPTU_4	GPTU input/output line 4
		I		SLSI1	SSC1 Slave Select input
		O		BREQ	EBU Bus Request Output
P0.5	R12	I/O	PUC	GPTU_5	GPTU input/output line 5
		I		HOLD	EBU Hold Request Input
		I		CC60_T12HR	CCU0 Timer 12 hardware run
		O		BRKOUT#_B	OCDS Break Out B
P0.6	R10	I/O	PUC	GPTU_6	GPTU input/output line 6
		I/O		HLDA	EBU Hold Acknowledge Input/Output
		I		CC60_T13HR	CCU0 Timer 13 hardware run
		O		SLSO0_0	SSC0 Slave Select output 0
P0.7	N10	I/O	PUC	GPTU_7	GPTU input/output line 7
		O		SLSO1_0	SSC1 Slave Select output 0
P0.8	R13	I	PUC	RXDCAN0_A	CAN node 0 receiver input A
		I		REQ0	External Trigger Input 0
		O		TCLK0A	MLI0 transmit channel clock output A
P0.9	R15	O	PUC	TXDCAN0_A	CAN node 0 transmitter output A
		I		TREADY0A	MLI0 transmit channel ready input A
		I		REQ1	External Trigger Input 1
P0.10	R14	I	PUC	RXDCAN1_A	CAN node 1 receiver input A
		I		REQ2	External Trigger Input 2
		O		TVALID0A	MLI0 transmit channel valid output A
P0.11	N9	O	PUC	TXDCAN1_A	CAN node 1 transmitter output A
		I		REQ3	External Trigger Input 3
		O		TDATA0A	MLI0 transmit channel data output A

Table 1 Pin Definitions and Functions(cont'd)

Symbol	Pin	In Out	PU/ PD[1)]	Functions	
P0.12	P9	I	PUC	RXDCAN2	CAN node 2 receiver input
		I		RCLK0A	MLI0 receive channel clock input A
		I		REQ4	External Trigger Input 4
P0.13	P8	O	PUC	TXDCAN2	CAN node 2 transmitter output
		I		REQ5	External Trigger Input 5
		O		RREADY0A	MLI0 receive channel ready output A
P0.14	N8	I	PUC	RXDCAN3	CAN node 3 receiver input
		I		REQ6	External Trigger Input 6
		I		RVALID0A	MLI0 receive channel valid input A
P0.15	P7	O	PUC	TXDCAN3	CAN node 3 transmitter output
		I		REQ7	External Trigger Input 7
		I		RDATA0A	MLI0 receive channel data input A

Table 1 Pin Definitions and Functions(cont'd)

Symbol	Pin	In Out	PU/ PD[1)]	Functions
P1		I/O		**Port 1** Port 1 serves as 16-bit bidirectional general purpose I/O port which can be used for input/output for Ethernet controller, MultiCAN, CAN, OCDS L2, SSC0/1, EBU and SCU
P1.0	D11	O	PUC	MII_TXD0 Ethernet controller transmit data output line 0
		I		RXDCAN0_B CAN node 0 receiver input B
		I		SWCFG0 Software configuration 0
		O		OCDSA_0 OCDS L2 Debug Line A0
P1.1	C12	O	PUC	MII_TXD1 Ethernet controller transmit data output line 1
		I		SWCFG1 Software configuration 1
		O		TXDCAN0_B CAN node 0 transmitter output B
		O		OCDSA_1 OCDS L2 Debug Line A1
P1.2	D12	O	PUC	MII_TXD2 Ethernet controller transmit data output line 2
		I		RXDCAN1_B CAN node 1 receiver input B
		I		SWCFG2 Software configuration 2
		O		OCDSA_2 OCDS L2 Debug Line A2
P1.3	B12	O	PUC	MII_TXD3 Ethernet controller transmit data output line 3
		O		TXDCAN1_B CAN node 1 transmitter output B
		I		SWCFG3 Software configuration 3
		O		OCDSA_3 OCDS L2 Debug Line A3
P1.4	C11	O	PUC	MII_TXER Ethernet controller transmit error output line
		I		SWCFG4 Software configuration 4
		O		OCDSA_4 OCDS L2 Debug Line A4
P1.5	C13	O	PUC	MII_TXEN Ethernet controller transmit enable output line
		I		SWCFG5 Software configuration 5
		O		OCDSA_5 OCDS L2 Debug Line A5
P1.6	A12	O	PUC	MII_MDC Ethernet controller management data clock output line
		I		SWCFG6 Software configuration 6
		O		OCDSA_6 OCDS L2 Debug Line A6

Table 1 Pin Definitions and Functions(cont'd)

Symbol	Pin	In Out	PU/ PD[1)]	Functions	
P1.7	B13	I	PUC	MII_RXDV	Ethernet Controller receive data valid input line
		I		SWCFG7	Software configuration 7
		O		OCDSA_7	OCDS L2 Debug Line A7
P1.8	A13	I	PUC	MII_CRS	Ethernet Controller carrier input line
		I		SWCFG8	Software configuration 8
		O		OCDSA_8	OCDS L2 Debug Line A8
P1.9	A14	I	PUC	MII_COL	Ethernet Controller collision input line
		I		SWCFG9	Software configuration 9
		O		OCDSA_9	OCDS L2 Debug Line A9
P1.10	B14	I	PUC	MII_RXD0	Ethernet Controller receive data input line 0
		I		SWCFG10	Software configuration 10
		O		OCDSA_10	OCDS L2 Debug Line A10
P1.11	C14	I	PUC	MII_RXD1	Ethernet Controller receive data input line 1
		I		SWCFG11	Software configuration 11
		O		OCDSA_11	OCDS L2 Debug Line A1
		O		SLSO0_1	SSC0 Slave Select output 1
P1.12	F13	I	PUC	MII_RXD2	Ethernet Controller receive data input line 2
		I		SWCFG12	Software configuration 12
		O		OCDSA_12	OCDS L2 Debug Line A12
		O		SLSO1_1	SSC1 Slave Select output 1
P1.13	E14	O	PUC	MII_RXD3	Ethernet Controller receive data input line 3
		I		SWCFG13	Software configuration 13
		O		OCDSA_13	OCDS L2 Debug Line A13
		O		SLSO0_2	SSC0 Slave Select output 2
P1.14	D14	I	PUC	MII_RXER	Ethernet Controller receive error input line
		O		SLSO1_2	SSC1 Slave Select output 2
		I		SWCFG14	Software configuration 14
		O		OCDSA_14	OCDS L2 Debug Line A14
P1.15	F14	I	PUC	SLSI0	SSC0 Slave Select Input
		O		RMW	EBU Read Modify Write
		I		SWCFG15	Software configuration 15
		O		OCDSA_15	OCDS L2 Debug Line A15

Table 1 Pin Definitions and Functions(cont'd)

Symbol	Pin	In Out	PU/ PD[1)]	Functions	
P2		I/O		**Port 2** Port 2 is a 16-bit bidirectional general purpose I/O port which can be alternatively used for ASC0/1/2, SSC0/1, CCU0, IIC, EBU and SCU.	
P2.0	P12	I/O	PUC	RXD0	ASC0 receiver input/output line
		O		CSEMU	EBU Chip Select Output for Emulator Region
P2.1	P11	O	PUC	TXD0	ASC0 transmitter output line
		I		TESTMODE	Test Mode Select Input
P2.2	P13	I/O	PUC	MRST0	SSC0 master receive / slave transmit input/output
P2.3	P14	I/O	PUC	MTSR0	SSC0 master transmit / slave receive input/output
P2.4	N15	I/O	PUC	SCLK0	SSC0 clock input/output line
P2.5	N14	O	PUC	COUT60_3	CCU0 compare channel 3 output
		I/O		MRST1A	SSC1 master receive / slave transmit input/output A
P2.6	N12	I/O	PUC	CC60_0	CCU0 input/output of capture/compare channel 0
		I/O		MTSR1A	SSC1 master transmit / slave receive input/output A
P2.7	K16	O	PUC	COUT60_0	CCU0 output of capture/compare channel 0
		I/O		SCLK1A	SSC1 clock input/output line A
P2.8	J16	I/O	PUC	CC60_1	CCU0 input/output of capture/ compare channel 1
		I/O		RXD1A	ASC1 receiver input/output line A
P2.9	H16	O	PUC	COUT60_1	CCU0 output of capture/compare channel 1
		O		TXD1A	ASC1 transmitter output line A
P2.10	L13	I/O	PUC	CC60_2	CCU0 input/output of capture/ compare channel 2
		I/O		RXD2A	ASC2 receiver input/output line A
P2.11	G16	O	PUC	COUT60_2	CCU0 output of capture/compare channel 2
		O		TXD2A	ASC2 transmitter output line A
P2.12	K15	I/O	—	SDA0	IIC Serial Data line 0
		I		CTRAP0	CCU0 trap input
		O		SLSO0_3	SSC0 Slave Select output 3

Table 1 Pin Definitions and Functions(cont'd)

Symbol	Pin	In Out	PU/ PD[1)]	Functions	
P2.13	K14	I/O	—	SCL0	IIC clock line 0
		I		CCPOS0_0	CCU0 Hall input signal 0
		O		SLSO1_3	SSC1 Slave Select output 3
P2.14	F16	I	—	CCPOS0_1	CCU0 Hall input signal 1
		I/O		SDA1	IIC Serial Data line 1
		O		SLSO0_4	SSC0 Slave Select output 4
P2.15	E16	I	—	CCPOS0_2	CCU0 Hall input signal 2
		I/O		SCL1	IIC clock line 1
		O		SLSO1_4	SSC1 Slave Select output 4

Table 1 Pin Definitions and Functions(cont'd)

Symbol	Pin	In Out	PU/ PD[1)]	Functions
P3		I/O		**Port 3** Port 3 is a 16-bit bidirectional general purpose I/O port which can be alternatively used for MLI1, CCU1, SSC0/1 and OCDS Level 2 debug lines.
P3.0	A15	O	PUC	OCDSB_0 OCDS L2 Debug Line B0
		O		COUT61_3 CCU1 compare channel 3 output
P3.1	B15	O	PUC	OCDSB_1 OCDS L2 Debug Line B1
		I/O		CC61_0 CCU1 input/output of capture/ compare channel 0
P3.2	D15	O	PUC	OCDSB_2 OCDS L2 Debug Line B2
		O		COUT61_0 CCU1 output of capture/compare channel 0
P3.3	E15	O	PUC	OCDSB_3 OCDS L2 Debug Line B3
		I/O		CC61_1 CCU1 input/output of capture/ compare channel 1
P3.4	G14	O	PUC	OCDSB_4 OCDS L2 Debug Line B4
		O		COUT61_1 CCU1 output of capture/compare channel 1
P3.5	G15	O	PUC	OCDSB_5 OCDS L2 Debug Line B5
		I/O		CC61_2 CCU1 input/output of capture/ compare channel 2
P3.6	F15	O	PUC	OCDSB_6 OCDS L2 Debug Line B6
		O		COUT61_2 CCU1 output of capture/compare channel 2
P3.7	H14	O	PUC	OCDSB_7 OCDS L2 Debug Line B7
		I		CTRAP1 CCU1 trap input
		O		SLSO0_5 SSC0 Slave Select output 5
P3.8	C15	O	PUC	OCDSB_8 OCDS L2 Debug Line B8
		I		CCPOS1_0 CCU1 Hall input signal 0
		O		TCLK1 MLI1 transmit channel clock output
		O		SLSO1_5 SSC1 Slave Select output 5
P3.9	H15	O	PUC	OCDSB_9 OCDS L2 Debug Line B9
		I		CCPOS1_1 CCU1 Hall input signal 1
		I		TREADY1 MLI1 transmit channel ready input
		O		SLSO0_6 SSC0 Slave Select output 6
P3.10	B16	O	PUC	OCDSB_10 OCDS L2 Debug Line B10
		I		CCPOS1_2 CCU1 Hall input signal 2
		O		TVALID1 MLI1 transmit channel valid output
		O		SLSO1_6 SSC1 Slave Select output 6

Table 1 Pin Definitions and Functions(cont'd)

Symbol	Pin	In Out	PU/ PD[1)]	Functions	
P3.11	C16	O	PUC	OCDSB_11	OCDS L2 Debug Line B11
		O		TDATA1	MLI1 transmit channel data output
		O		SLSO0_7	SSC0 Slave Select output 7
		I		CC61_T12HR	CCU1 Timer 12 hardware run
P3.12	D16	O	PUC	OCDSB_12	OCDS L2 Debug Line B12
		I		RCLK1	MLI1 receive channel clock input
		O		SLSO1_7	SSC1 Slave Select output 7
		I		CC61_T13HR	CCU1 Timer 13 hardware run
P3.13	K13	O	PUC	OCDSB_13	OCDS L2 Debug Line B13
		O		RREADY1	MLI1 receive channel ready output
		I/O		MRST1B	SSC1 master receive / slave transmit input/output B
P3.14	J14	O	PUC	OCDSB_14	OCDS L2 Debug Line B14
		I		RVALID1	MLI1 receive channel valid input
		I/O		MTSR1B	SSC1 master transmit / slave receive input/output B
P3.15	J15	O	PUC	OCDSB_15	OCDS L2 Debug Line B15
		I		RDATA1	MLI1 receive channel data input
		I/O		SCLK1B	SSC1 clock input/output line B

Table 1 Pin Definitions and Functions(cont'd)

Symbol	Pin	In Out	PU/ PD[1)]	Functions
P4		I/O		**Port 4** Port 4 is a 8-bit bidirectional general purpose I/O port which can be alternatively used for USB, MLI0 and SCU.
P4.0	R8	I	PUC	USBCLK 48MHz input clock
		O		TCLK0B MLI0 transmit channel clock output B
P4.1	R9	I	PUC	RCVI USB data input
		I		TREADY0B MLI0 transmit channel ready input B
P4.2	N7	I	PUC	VPI USB D+ CMOS level mirror of differential signal
		O		TVALID0B MLI0 transmit channel valid output B
P4.3	N6	I	PUC	VMI USB D- CMOS level mirror of differential signal
		O		TDATA0B MLI0 transmit channel data output B
P4.4	P6	O	PUC	VPO USB D+ CMOS level output
		I		RCLK0B MLI0 receive channel clock input B
P4.5	R7	O	PUC	VMO USB D- CMOS level output
		O		RREADY0B MLI0 receive channel ready output B
P4.6	R6	O	PUC	USBOE Direction select for transmit or receive
		I		RVALID0B MLI0 receive channel valid input B
P4.7	P5	I	PUC	RDATA0B MLI0 receive channel data input B
		O		BRKOUT#_A OCDS Break Out A
$\overline{\text{HDRST}}$	N5	I/O	PUA	**Hardware Reset Input/Reset Indication Output** Assertion of this bidirectional open-drain pin causes a synchronous reset of the chip through external circuitry. This pin must be driven for a minimum duration. The internal reset circuitry drives this pin in response to a power-on, hardware, watchdog and power-down wake-up reset for a specific period of time. For a software reset, activation of this pin is programmable.
$\overline{\text{PORST}}$	R5	I	PUC	**Power-on Reset Input** A low level on $\overline{\text{PORST}}$ causes an asynchronous reset of the entire chip. $\overline{\text{PORST}}$ is a fully asynchronous level sensitive signal.
$\overline{\text{NMI}}$	T7	I	PUC	**Non-Maskable Interrupt Input** A high-to-low transition on this pin causes a NMI-Trap request to the CPU.

Table 1 Pin Definitions and Functions(cont'd)

Symbol	Pin	In Out	PU/ PD[1)]	Functions
$\overline{\text{TRST}}$	T11	I	PDC	**JTAG Module Reset/Enable Input** A low level at this pin resets and disables the JTAG module. A high level enables the JTAG module.
TCK	T12	I	PUC	**JTAG Module Clock Input**
TDI	T13	I	PUC	**JTAG Module Serial Data Input**
TDO	T10	O	—	**JTAG Module Serial Data Output**
TMS	T9	I	PUC	**JTAG Module State Machine Control Input**
$\overline{\text{TRCLK}}$	T8	O	—	**Trace Clock for OCDS_L2 Lines**
HWCFG0 **HWCFG1** **HWCFG2**	M14 L14 T6	I I I	PUC PUC PDC	**Hardware Configuration Inputs** The Configuration Inputs define the boot options of the TC1130 after a hardware invoked reset operation.
$\overline{\text{BRKIN}}$	T5	I	PUC	**OCDS Break Input** A low level on this pin causes a break in the chip's execution when the OCDS is enabled. In addition, the level of this pin during power-on reset determines the boot configuration.
MII_ TXCLK	T2	I	PDC	**Ethernet Controller Transmit Clock** MII_TXD[3:0] and MII_TXEN are driven off the rising edge of the MII_TXCLK by the core and sampled by the PHY on the rising edge of the MII_TXCLK.
MII_ RXCLK	R2	I	PDC	**Ethernet Controller Receive Clock** MII_RXCLK is a continuous clock. Its frequency is 25 MHz for 100 Mbps operation, and 2.5 MHz for 10Mbps. MII_RXD[3:0], MII_RXDV and MII_EXER are driven by the PHY off the falling edge of MII_RXCLK and sampled on the rising edge of MII_RXCLK.
MII_ MDIO	R1	I/O	PDA	**Ethernet Controller Management Data Input / Output** When a read command is being executed, data which is clocked out of the PHY will be presented on the input line. When the Core is clocking control or data onto the MII_MDIO line, the signal will carry the information.
D+	T14	I/O	—	**USB D+ data line**
D-	T15	I/O	—	**USB D- data line**

Table 1 Pin Definitions and Functions(cont'd)

Symbol	Pin	In Out	PU/ PD[1)]	Functions
$\overline{CS0}$ **$\overline{CS1}$** **$\overline{CS2}$** **$\overline{CS3}$**	D9 D8 C9 B8	O O O O	PUC PUC PUC PUC	**EBU Chip Select Output Line 0** **EBU Chip Select Output Line 1** **EBU Chip Select Output Line 2** **EBU Chip Select Output Line 3** Each corresponds to a programmable region. Only one can be active at one time.
$\overline{CSCOMB}$	N3	O	PUC	**EBU Chip Select Output for combination function (Overlay Memory and Global)**
SDCLKI	J1	I	—	SDRAM clock input (clock feedback).
SDCLKO	H1	O	—	SDRAM clock output.Accesses to SDRAM devices are synchronized to this clock
$\overline{RAS}$	D6	O	PUC	**EBU SDRAM Row Address Strobe Output**
$\overline{CAS}$	D5	O	PUC	**EBU SDRAM Column Address Strobe Output**
CKE	L4	O	PUC	**EBU SDRAM Clock Enable Output**
BFCLKI	D1	I	—	Burst FLASH clock input (clock feedback).
BFCLKO	E1	O	—	Burst FLASH clock output. Accesses to Burst FLASH devices are synchronized to this clock.
$\overline{RD}$	P2	O	PUC	**EBU Read Control Line** Output in the master mode Input in the slave mode.
RD/$\overline{WR}$	T3	O	PUC	**EBU Write Control Line** Output in the master mode Input in the slave mode.
$\overline{WAIT}$	B9	I	PUC	**EBU Wait Control Line**
ALE	R3	O	PDC	**EBU Address Latch Enable Output**
MR/$\overline{W}$	P3	O	PUC	**EBU Motorola-style Read / Write Output**
$\overline{BAA}$	A11	O	PUC	**EBU Burst Address Advance Output** For advancing address in a burst flash access
$\overline{ADV}$	B11	O	PUC	**EBU Burst Flash Address Valid Output**

Table 1 Pin Definitions and Functions(cont'd)

Symbol	Pin	In Out	PU/ PD[1)]	Functions
				EBU Address / Data Bus Input / Output Lines
AD0	C8	I/O	PUC	EBU Address / Data Bus Line 0
AD1	C7	I/O	PUC	EBU Address / Data Bus Line 1
AD2	B6	I/O	PUC	EBU Address / Data Bus Line 2
AD3	C6	I/O	PUC	EBU Address / Data Bus Line 3
AD4	C5	I/O	PUC	EBU Address / Data Bus Line 4
AD5	A3	I/O	PUC	EBU Address / Data Bus Line 5
AD6	A2	I/O	PUC	EBU Address / Data Bus Line 6
AD7	C3	I/O	PUC	EBU Address / Data Bus Line 7
AD8	C2	I/O	PUC	EBU Address / Data Bus Line 8
AD9	D2	I/O	PUC	EBU Address / Data Bus Line 9
AD10	F1	I/O	PUC	EBU Address / Data Bus Line 10
AD11	E3	I/O	PUC	EBU Address / Data Bus Line 11
AD12	F3	I/O	PUC	EBU Address / Data Bus Line 12
AD13	G1	I/O	PUC	EBU Address / Data Bus Line 13
AD14	H2	I/O	PUC	EBU Address / Data Bus Line 14
AD15	G3	I/O	PUC	EBU Address / Data Bus Line 15
AD16	D7	I/O	PUC	EBU Address / Data Bus Line 16
AD17	B5	I/O	PUC	EBU Address / Data Bus Line 17
AD18	A4	I/O	PUC	EBU Address / Data Bus Line 18
AD19	B4	I/O	PUC	EBU Address / Data Bus Line 19
AD20	C4	I/O	PUC	EBU Address / Data Bus Line 20
AD21	B3	I/O	PUC	EBU Address / Data Bus Line 21
AD22	B2	I/O	PUC	EBU Address / Data Bus Line 22
AD23	B1	I/O	PUC	EBU Address / Data Bus Line 23
AD24	C1	I/O	PUC	EBU Address / Data Bus Line 24
AD25	D3	I/O	PUC	EBU Address / Data Bus Line 25
AD26	E2	I/O	PUC	EBU Address / Data Bus Line 26
AD27	F2	I/O	PUC	EBU Address / Data Bus Line 27
AD28	F4	I/O	PUC	EBU Address / Data Bus Line 28
AD29	G4	I/O	PUC	EBU Address / Data Bus Line 29
AD30	H3	I/O	PUC	EBU Address / Data Bus Line 30
AD31	G2	I/O	PUC	EBU Address / Data Bus Line 31
$\overline{BC0}$	A5	O	PUC	**EBU Byte Control Line 0**
$\overline{BC1}$	A6	O	PUC	**EBU Byte Control Line 1**
$\overline{BC2}$	B7	O	PUC	**EBU Byte Control Line 2**
$\overline{BC3}$	A7	O	PUC	**EBU Byte Control Line 3**

Table 1 Pin Definitions and Functions(cont'd)

Symbol	Pin	In Out	PU/ PD[1)]	Functions
				EBU Address Bus Input / Output Lines
A0	K1	O	PUC	EBU Address Bus Line 0
A1	L1	O	PUC	EBU Address Bus Line 1
A2	M1	O	PUC	EBU Address Bus Line 2
A3	N1	O	PUC	EBU Address Bus Line 3
A4	P1	O	PUC	EBU Address Bus Line 4
A5	J2	O	PUC	EBU Address Bus Line 5
A6	K2	O	PUC	EBU Address Bus Line 6
A7	L2	O	PUC	EBU Address Bus Line 7
A8	M2	O	PUC	EBU Address Bus Line 8
A9	N2	O	PUC	EBU Address Bus Line 9
A10	J3	O	PUC	EBU Address Bus Line 10
A11	K3	O	PUC	EBU Address Bus Line 11
A12	L3	O	PUC	EBU Address Bus Line 12
A13	M3	O	PUC	EBU Address Bus Line 13
A14	K4	O	PUC	EBU Address Bus Line 14
A15	A8	O	PUC	EBU Address Bus Line 15
A16	A9	O	PUC	EBU Address Bus Line 16
A17	A10	O	PUC	EBU Address Bus Line 17
A18	B10	O	PUC	EBU Address Bus Line 18
A19	C10	O	PUC	EBU Address Bus Line 19
A20	D10	O	PUC	EBU Address Bus Line 20
A21	T4	O	PUC	EBU Address Bus Line 21
A22	R4	O	PUC	EBU Address Bus Line 22
A23	P4	O	PUC	EBU Address Bus Line 23
XTAL1 **XTAL2**	M16 N16	I O	— —	**Oscillator/PLL/Clock Generator Input/Output Pins** XTAL1 is the input to the main oscillator amplifier and input to the internal clock generator. XTAL2 is the output of the main oscillator amplifier circuit. For clocking the device from an external source, XTAL1 is driven with the clock signal while XTAL2 is left unconnected. For crystal oscillator operation XTAL1 and XTAL2 are connected to the crystal with the appropriate recommended oscillator circuitry.
V_{DDOSC3}	P16	—	—	**Main Oscillator Power Supply (3.3V)**
V_{SSOSC3}	R16	—	—	**Main Oscillator Ground**
V_{DDOSC}	L16	—	—	**Main Oscillator Power Supply (1.5V)**

Table 1 Pin Definitions and Functions(cont'd)

Symbol	Pin	In Out	PU/ PD[1)]	Functions
V_{SSOSC}	L15	—	—	**Main Oscillator Ground**
V_{DD}	G7, G8 G9 G10 G13 K7,K8 K9 K10	—	—	**Core and Logic Power Supply (1.5V)**
V_{DDP}	D4, D13, H4, J13, M4, N13,	—	—	**Ports Power Supply (3.3V)**
V_{SS}	E4 E13 H7, H8 H9 H10 H13 J4,J7 J8,J9 J10 M13 N4	—	—	**Ground**
N.C.	A1, A16, T1, T16	—	—	**Not Connected** These pins must not be connected.

1) Refers to internal pull-up or pull-down device connected and corresponding type. The notation '—' indicates that the internal pull-up or pull-down device is not enabled.

Parallel Ports

The TC1130 has 72 digital input/output port lines, which are organized into four parallel 16-bit ports and one parallel 8-bit port, Port P0 to Port P4 with 3.3V nominal voltage.

The digital parallel ports can be all used as general purpose I/O lines or they can perform input/output functions for the on-chip peripheral units. An overview on the port-to-peripheral unit assignment is shown in **Figure 4**.

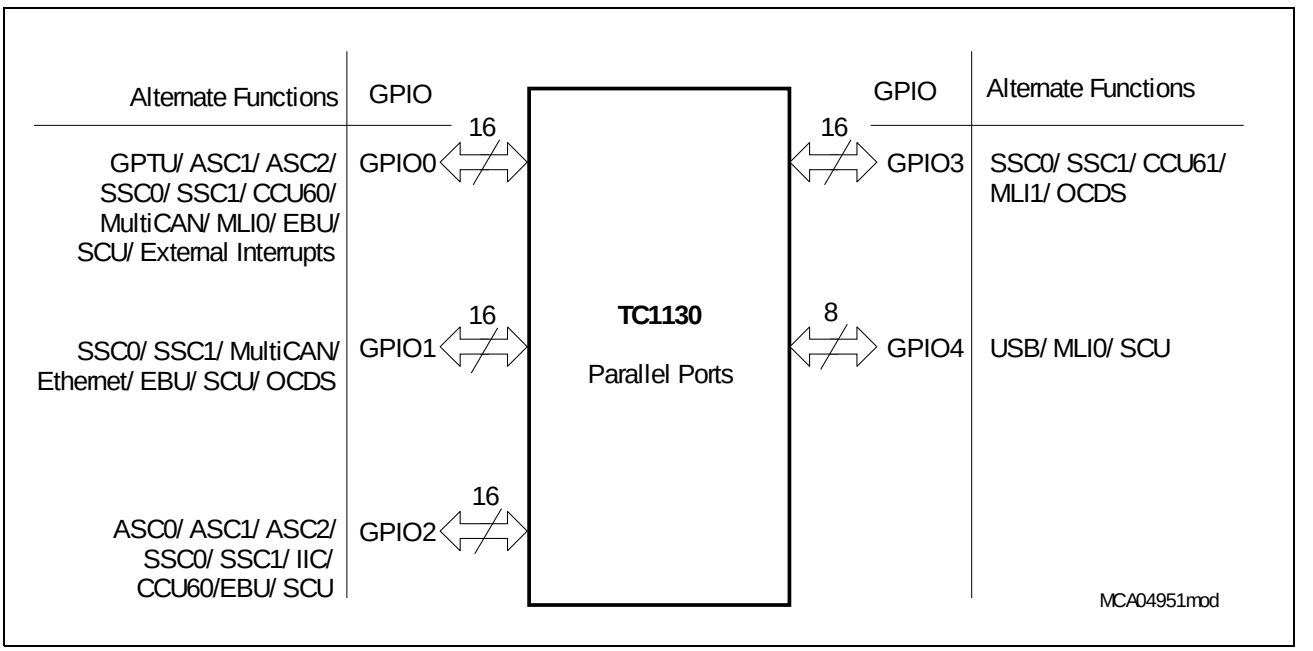


Figure 4 Parallel Ports of the TC1130

Serial Interfaces

The TC1130 includes five serial peripheral interface units:

- Asynchronous/Synchronous Serial Interface (ASC)
- High-Speed Synchronous Serial Interface (SSC)
- Inter IC Serial Interface (IIC)
- Universal Serial Bus Interface (USB)
- Micro Link Serial Bus Interface (MLI)

Asynchronous/Synchronous Serial Interface (ASC)

Figure 5 shows a global view of the functional block of three Asynchronous/ Synchronous Serial interfaces (ASC0, ASC1 and ASC2).

Each ASC Module, (ASC0/ASC1/ASC2) communicates with the external world via one pair of I/O lines. The RXD line is the receive data input signal (in Synchronous Mode also output). TXD is the transmit output signal. Clock control, address decoding, and interrupt service request control are managed outside the ASC Module kernel.

The Asynchronous/Synchronous Serial Interfaces provide serial communication between the TC1130 and other microcontrollers, microprocessors or external peripherals.

Each ASC supports full-duplex asynchronous communication and half-duplex synchronous communication. In Synchronous Mode, data is transmitted or received synchronous to a shift clock which is generated by the ASC internally. In Asynchronous Mode, 8-bit or 9-bit data transfer, parity generation, and the number of stop bits can be selected. Parity, framing, and overrun error detection are provided to increase the reliability of data transfers. Transmission and reception of data are double-buffered. For multiprocessor communication, a mechanism is included to distinguish address bytes from data bytes. Testing is supported by a loop-back option. A 13-bit baud rate generator provides the ASC with a separate serial clock signal that can be very accurately adjusted by a prescaler implemented as a fractional divider.

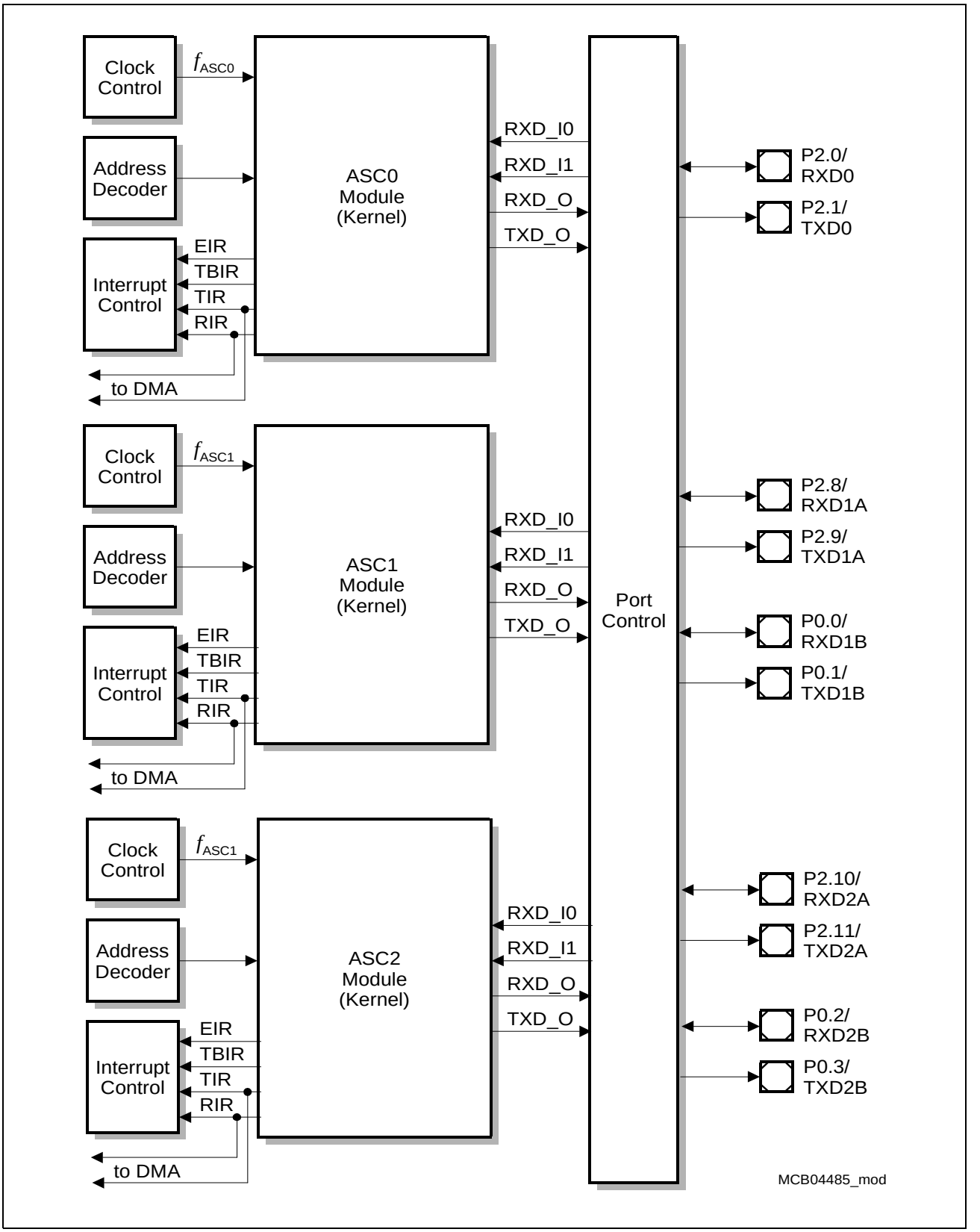


Figure 5 General Block Diagram of the ASC Interfaces

Features:

- Full-duplex asynchronous operating modes
  - 8-bit or 9-bit data frames, LSB first
  - Parity bit generation/checking
  - One or two stop bits
  - Baud rate from 4.6875 MBaud to 1.1 Baud (@ 75 MHz clock)
- Multiprocessor mode for automatic address/data byte detection
- Loop-back capability
- Half-duplex 8-bit synchronous operating mode
  - Baud rate from 9.375 MBaud to 762.9 Baud (@ 75 MHz clock)
- Support for IrDA data transmission up to 115.2 KBaud maximum.
- Double buffered transmitter/receiver
- Interrupt generation
  - On a transmitter buffer empty condition
  - On a transmit last bit of a frame condition
  - On a receiver buffer full condition
  - On an error condition (frame, parity, overrun error)
- FIFO
  - 8 byte receive FIFO (RXFIFO)
  - 8 byte transmit FIFO (TXFIFO)
  - Independent control of RXFIFO and TXFIFO
  - 9-bit FIFO data width
  - Programmable Receive/Transmit Interrupt Trigger Level
  - Receive and Transmit FIFO filling level indication
  - Overrun error generation
  - Underflow error generation

High-Speed Synchronous Serial Interface (SSC)

Figure 6 shows a global view of the functional blocks of two High-Speed Synchronous Serial interfaces (SSC0 and SSC1).

Each SSC supports full-duplex and half-duplex serial synchronous communication up to 37.5 MBaud (@ 75 MHz module clock) with receive and transmit FIFO support. The serial clock signal can be generated by the SSC itself (master mode) or can be received from an external master (slave mode). Data width, shift direction, clock polarity and phase are programmable. This allows communication with SPI-compatible devices. Transmission and reception of data is double-buffered. A shift clock generator provides the SSC with a separate serial clock signal. Eight slave select inputs are available for slave mode operation. Eight programmable slave select outputs (chip selects) are supported in master mode.

Features:

- Master and slave mode operation
  - Full-duplex or half-duplex operation
  - Automatic pad control possible
- Flexible data format
  - Programmable number of data bits: 2 to 16 bit
  - Programmable shift direction: LSB or MSB shift first
  - Programmable clock polarity: idle low or high state for the shift clock
  - Programmable clock/data phase: data shift with leading or trailing edge of the shift clock
- Baud rate generation minimum at 572.2 Baud (@ 75 MHz module clock)
- Interrupt generation
  - On a transmitter empty condition
  - On a receiver full condition
  - On an error condition (receive, phase, baud rate, transmit error)
- Four-pin interface
- Flexible SSC pin configuration
- Up to eight slave select inputs in slave mode
- Up to eight programmable slave select outputs SLSO in master mode
  - Automatic SLSO generation with programmable timing
  - Programmable active level and enable control
- 4-stage receive FIFO (RXFIFO) and 4-stage transmit FIFO (TXFIFO)
  - Independent control of RXFIFO and TXFIFO
  - 2 to 16 bit FIFO data width
  - Programmable receive/transmit interrupt trigger level
  - Receive and transmit FIFO filling level indication
  - Overrun error generation
  - Underflow error generation

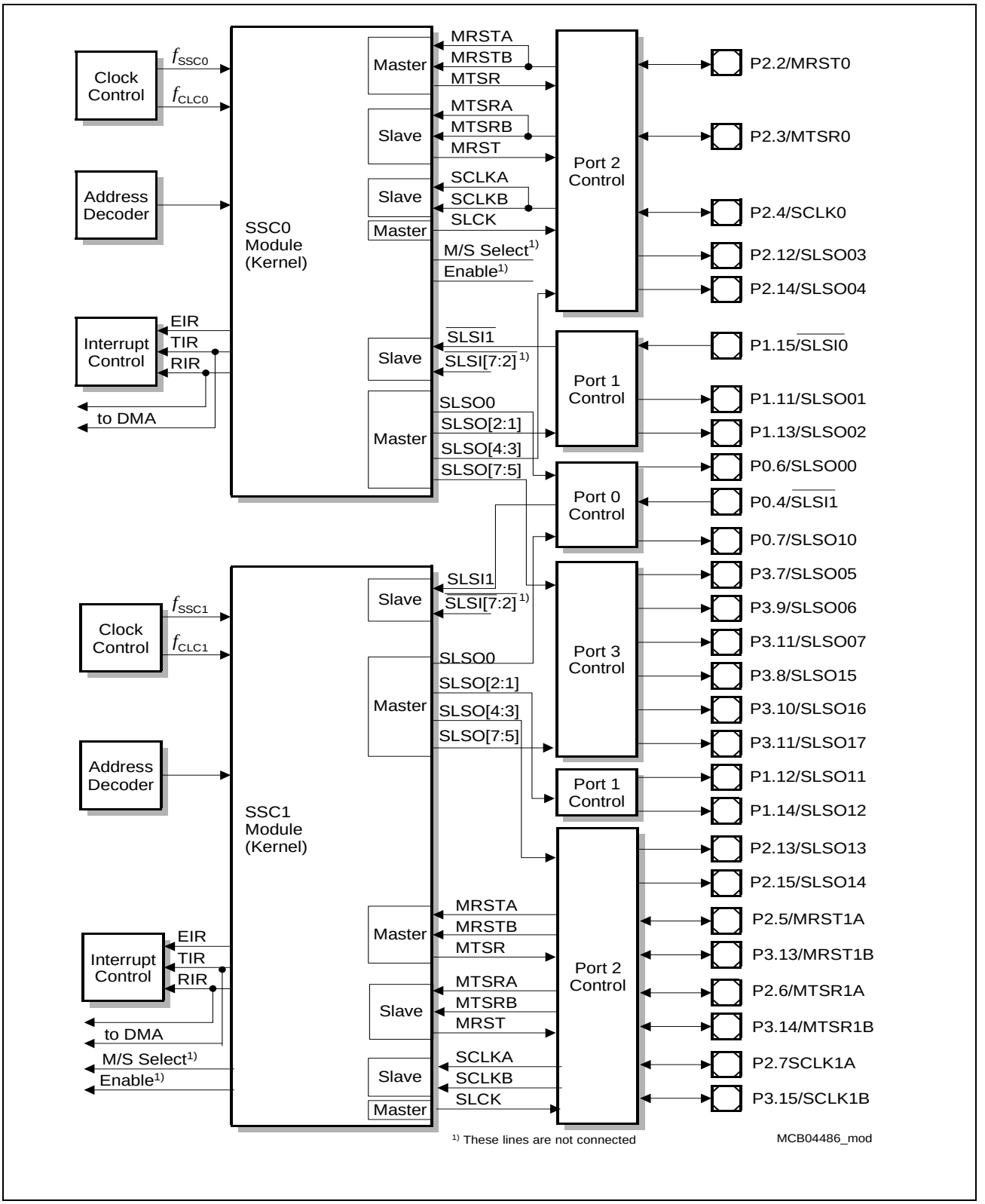


Figure 6 General Block Diagram of the SSC Interfaces

Inter IC Serial Interface (IIC)

Figure 7 shows a global view of the functional blocks of the Inter IC Serial Interface (IIC).

The IIC module has four I/O lines, located at Port 2. The IIC module is further supplied by a clock control, interrupt control, and address decoding logic. One DMA request can be generated by IIC module.

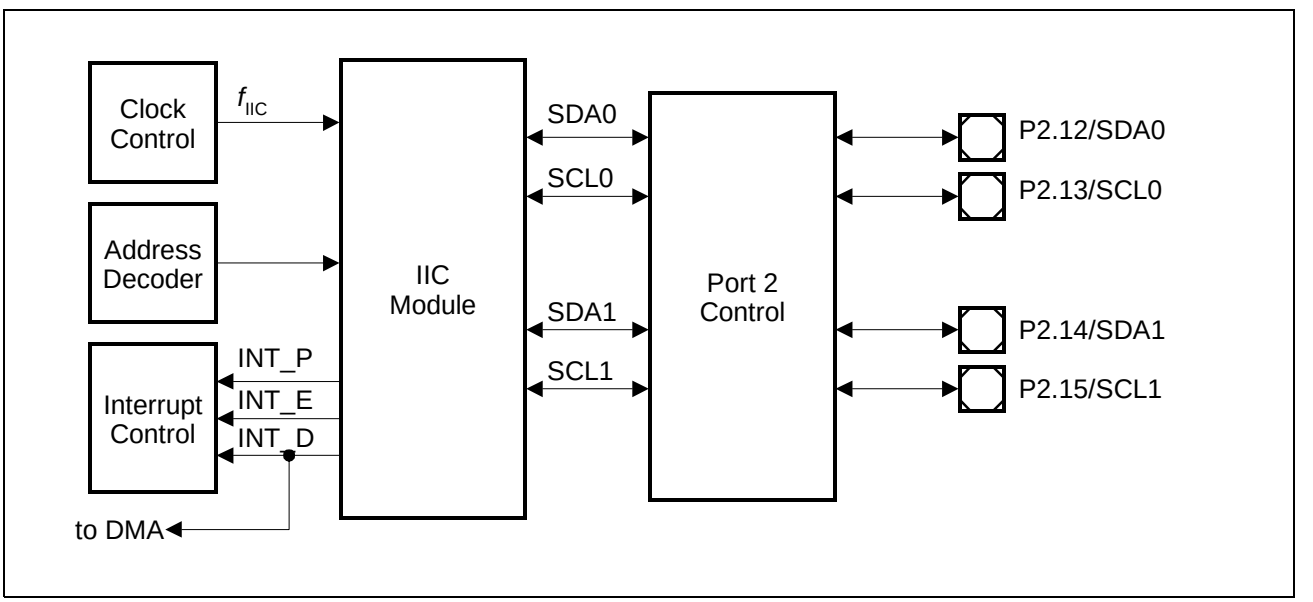


Figure 7 General Block Diagram of the IIC Interface

The on-chip IIC Bus module connects the platform buses to other external controllers and/or peripherals via the two-line serial IIC interface. One line is responsible for clock transfer and synchronization (SCL), the other is responsible for the data transfer (SDA). The IIC Bus module provides communication at data rates of up to 400 Kbit/s and features 7-bit addressing as well as 10-bit addressing. This module is fully compatible to the IIC bus protocol.

The module can operate in three different modes:

Master mode, where the IIC controls the bus transactions and provides the clock signal.

Slave mode, where an external master controls the bus transactions and provides the clock signal.

Multimaster mode, where several masters can be connected to the bus, i.e. the IIC can be master or slave.

The on-chip IIC bus module allows efficient communication via the common IIC bus. The module unloads the CPU of low level tasks like

- (De)Serialization of bus data.
- Generation of start and stop conditions.
- Monitoring the bus lines in slave mode.
- Evaluation of the device address in slave mode.
- Bus access arbitration in multimaster mode.

Features

- Software compatible to V1.0 of C161RI.
- Extended buffer allows up to 4 send/receive data bytes to be stored.
- Selectable baud rate generation.
- Support of standard 100 KBaud and extended 400 KBaud data rates.
- Operation in 7-bit addressing mode or 10-bit addressing mode.
- Flexible control via interrupt service routines or by polling.
- Dynamic access to up to 2 physical IIC busses.

Universal Serial Bus Interface (USB)

Figure 8 shows a global view of the functional blocks of the Universal Serial Bus Interface (USB).

The USB module is further supplied by clock control, interrupt control, address decoding, and port control logic. One DMA request can be generated by USB module.

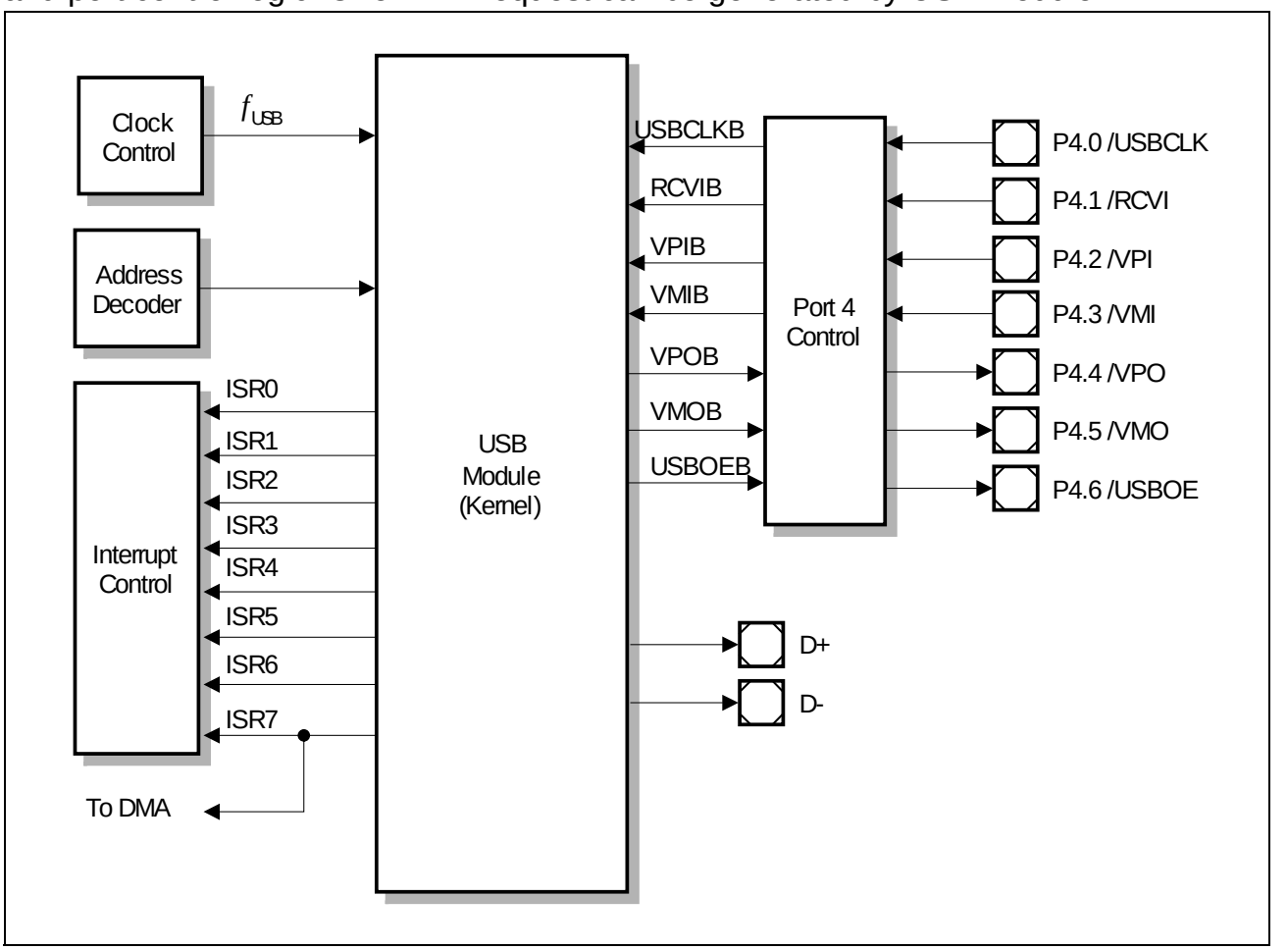


Figure 8 General Block Diagram of the USB

The USB handles all transactions between the serial USB bus and the internal (parallel) bus of the microcontroller. The USB module includes several units which are required to support data handling with the USB bus: the on-chip USB transceiver (optionally), the flexible USB buffer block with a 32 bit wide RAM, the buffer control unit with sub modules for USB and CPU memory access control, the UDC_IF device interface for USB protocol handling, the microcontroller interface unit (MCU) with the USB specific special function registers and the interrupt generation unit. A clock generation unit provides the clock signal for the USB module for full speed and low speed USB operation.

Features

- USB1.1 Device Standard Interface
- Differential I/O allow cable length up to 5m without additional hardware at target's end.
- Hot attach
- USB1.1 full speed device
- USB protocol handling in hardware
- Clock and data recovery from USB
- Bit stripping and bit stuffing functions
- CRC5 checking, CRC16 generation and checking
- Serial to parallel data conversion
- Maintenance of data synchronization bits (DATA0/DATA1 Toggle Bits)
- Supports multiple configurations, interfaces and alternate settings
- Sixteen endpoints with user configurable endpoint information
- Flexible intermediate buffering of transmission data
- Powerful data handling capability, FIFO-support
- Back-to-back transfers fully supported by module automatism
- Multi packet transfer without CPU load
- Handles data transfer with minimum CPU load
- Auto increment and single address modes selectable for easy data access
- Powerful interrupt generation
- Meets suspend power consumption restrictions in Power Down Mode
- Remote wakeup from USB bus activity
- Explicit support of setup information
- Enhanced status monitoring

Micro Link Serial Bus Interface (MLI)

Figure 9 shows a global view of the functional blocks of two Micro Link Serial Bus Interfaces (MLI0 & MLI1).

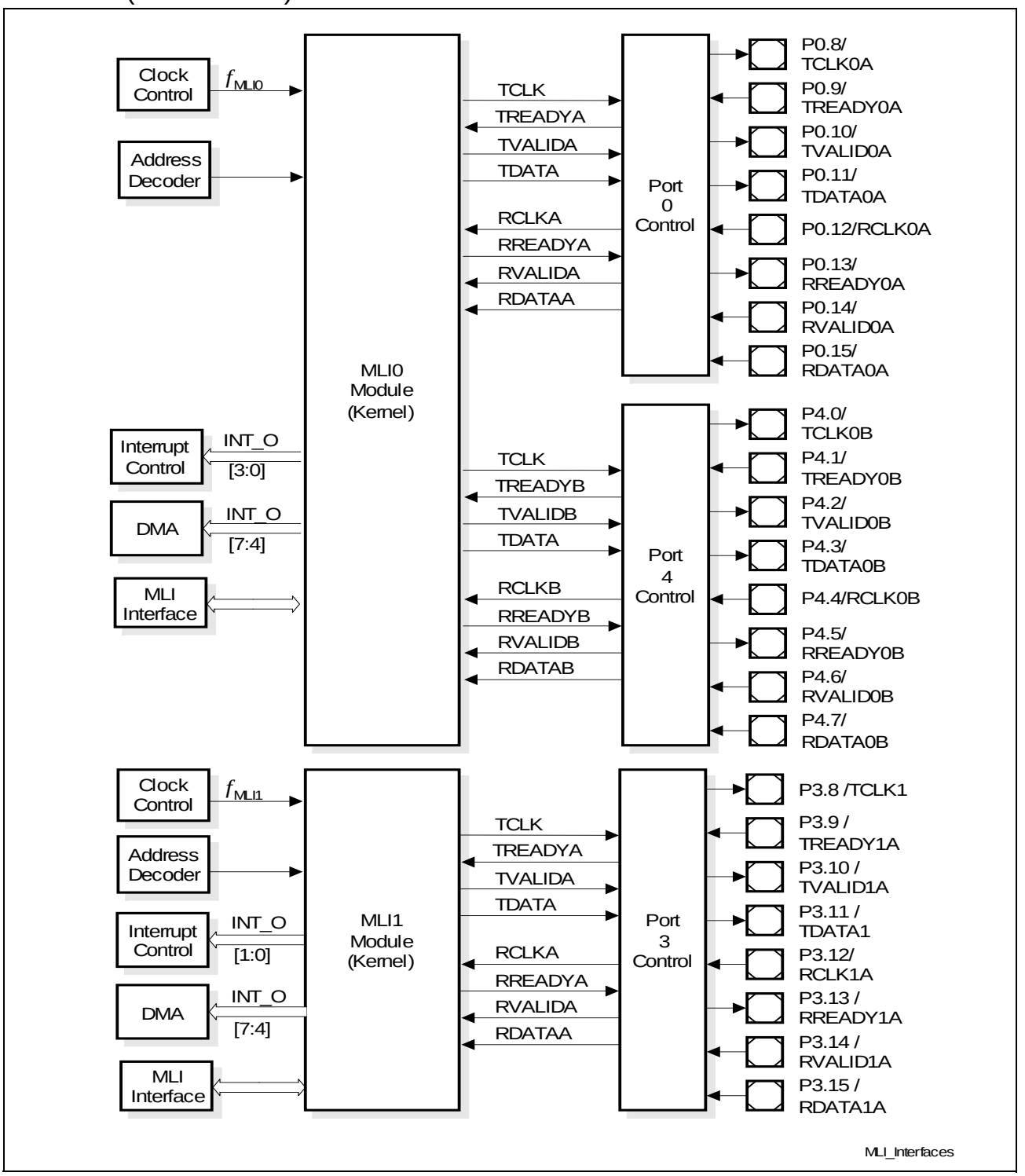


Figure 9 General Block Diagram of the MLI0 and MLI1

The Micro Link Serial Bus Interface is dedicated for the serial communication between controllers of the AUDO - NG family. The communication is intended to be fast and intelligent due to an address translation system, and it is not necessary to have any special program in the second controller.

Features:

- Serial communication from the MLI transmitter to MLI receiver of another controller
- Module supports connection of each MLI with up to four MLI from other controllers (see implementation sub-chapter for details for this product)
- Fully transparent read/write access supported (= remote programming)
- Complete address range of target controller available
- Special protocol to transfer data, address offset, or address offset and data
- Error control using a parity bit
- 32 - bits, 16 - bits, and 8 - bits data transfers
- Address offset width: from 1 to 16 - bits
- Baud rate: f_{MLI} / 2 (symmetric shift clock approach),
  baud rate definition by the corresponding fractional divider

General Purpose Timer Unit

Figure 10 shows a global view of all functional blocks of the General Purpose Timer Unit (GPTU).

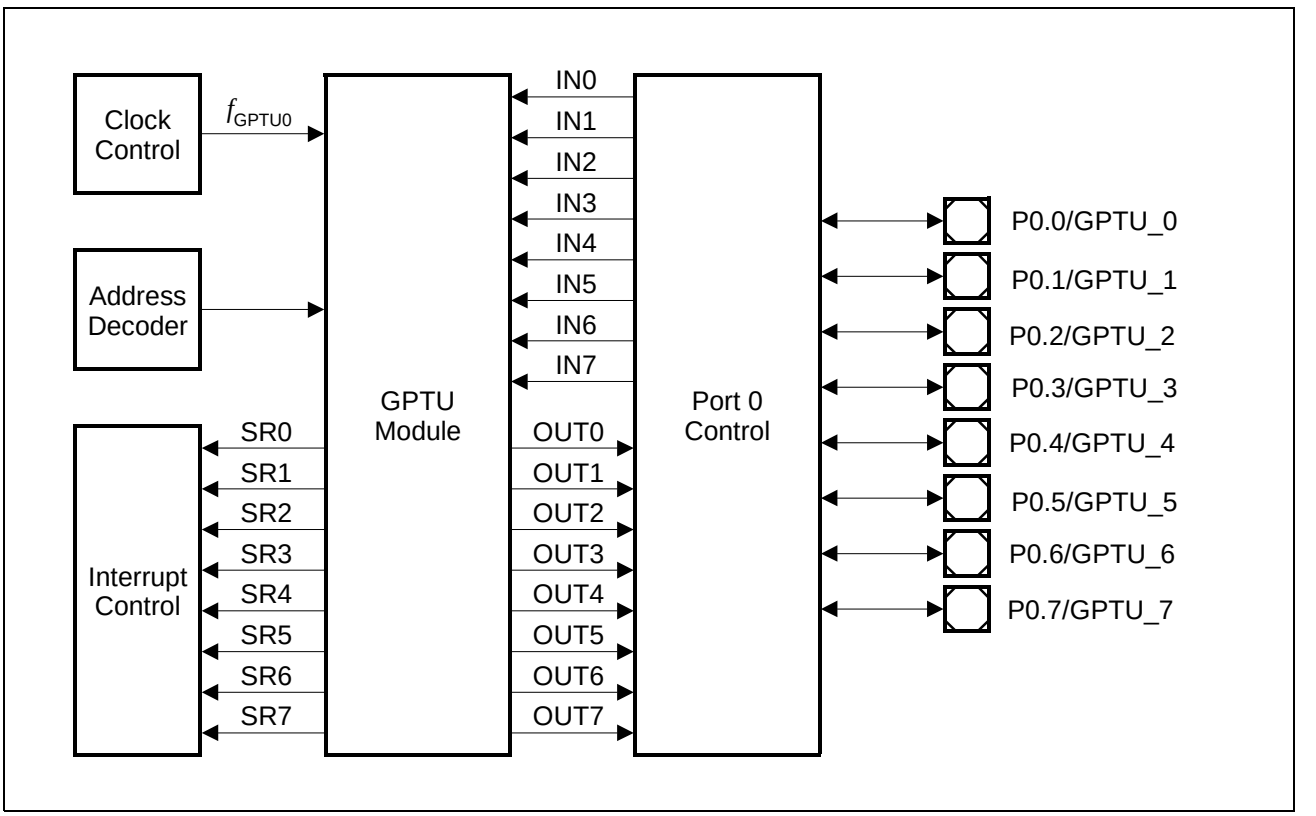


Figure 10 General Block Diagram of the GPTU Interface

The GPTU consists of three 32-bit timers designed to solve such application tasks as event timing, event counting, and event recording. The GPTU communicates with the external world via eight I/O lines located at Port 0.

The three timers of GPTU Module T0, T1, and T2, can operate independently from each other or can be combined:

General Features:

- All timers are 32-bit precision timers with a maximum input frequency of f_{GPTU}.
- Events generated in T0 or T1 can be used to trigger actions in T2
- Timer overflow or underflow in T2 can be used to clock either T0 or T1
- T0 and T1 can be concatenated to form one 64-bit timer

Features of T0 and T1:

- Each timer has a dedicated 32-bit reload register with automatic reload on overflow
- Timers can be split into individual 8-, 16-, or 24-bit timers with individual reload registers

- Overflow signals can be selected to generate service requests, pin output signals, and T2 trigger events
- Two input pins can define a count option

Features of T2:

- Count up or down is selectable
- Operating modes:
  - Timer
  - Counter
  - Quadrature counter (incremental/phase encoded counter interface)
- Options:
  - External start/stop, one-shot operation, timer clear on external event
  - Count direction control through software or an external event
  - Two 32-bit reload/capture registers
- Reload modes:
  - Reload on overflow or underflow
  - Reload on external event: positive transition, negative transition, or both transitions
- Capture modes:
  - Capture on external event: positive transition, negative transition, or both transitions
  - Capture and clear timer on external event: positive transition, negative transition, or both transitions
- Can be split into two 16-bit counter/timers
- Timer count, reload, capture, and trigger functions can be assigned to input pins. T0 and T1 overflow events can also be assigned to these functions.
- Overflow and underflow signals can be used to trigger T0 and/or T1 and to toggle output pins
- T2 events are freely assignable to the service request nodes.

Capture/Compare Unit 6 (CCU6)

Figure 11 shows a global view of all functional blocks of two Capture/Compare Units (CCU60 & CCU61).

Both of the CCU6 modules is further supplied by clock control, interrupt control, address decoding, and port control logic. One DMA request can be generated by each CCU6 module.

Each CCU6 provides two independent timers (T12, T13), which can be used for PWM generation, especially for AC-motor control. Additionally, special control modes for block commutation and multi-phase machines are supported.

Timer 12 Features

- Three capture/compare channels, each channel can be used either as capture or as compare channel.
- Generation of a three-phase PWM supported (six outputs, individual signals for highside and lowside switches)
- 16 bit resolution, maximum count frequency = peripheral clock
- Dead-time control for each channel to avoid short-circuits in the power stage
- Concurrent update of the required T12/13 registers
- Center-aligned and edge-aligned PWM can be generated
- Single-shot mode supported
- Many interrupt request sources
- Hysteresis-like control mode

Timer 13 Features

- One independent compare channel with one output
- 16 bit resolution, maximum count frequency = peripheral clock
- Can be synchronized to T12
- Interrupt generation at period-match and compare-match
- Single-shot mode supported

Additional Features

- Block commutation for Brushless DC-drives implemented
- Position detection via Hall-sensor pattern
- Automatic rotational speed measurement for block commutation
- Integrated error handling
- Fast emergency stop without CPU load via external signal ($\overline{\text{CTRAP}}$)
- Control modes for multi-channel AC-drives
- Output levels can be selected and adapted to the power stage

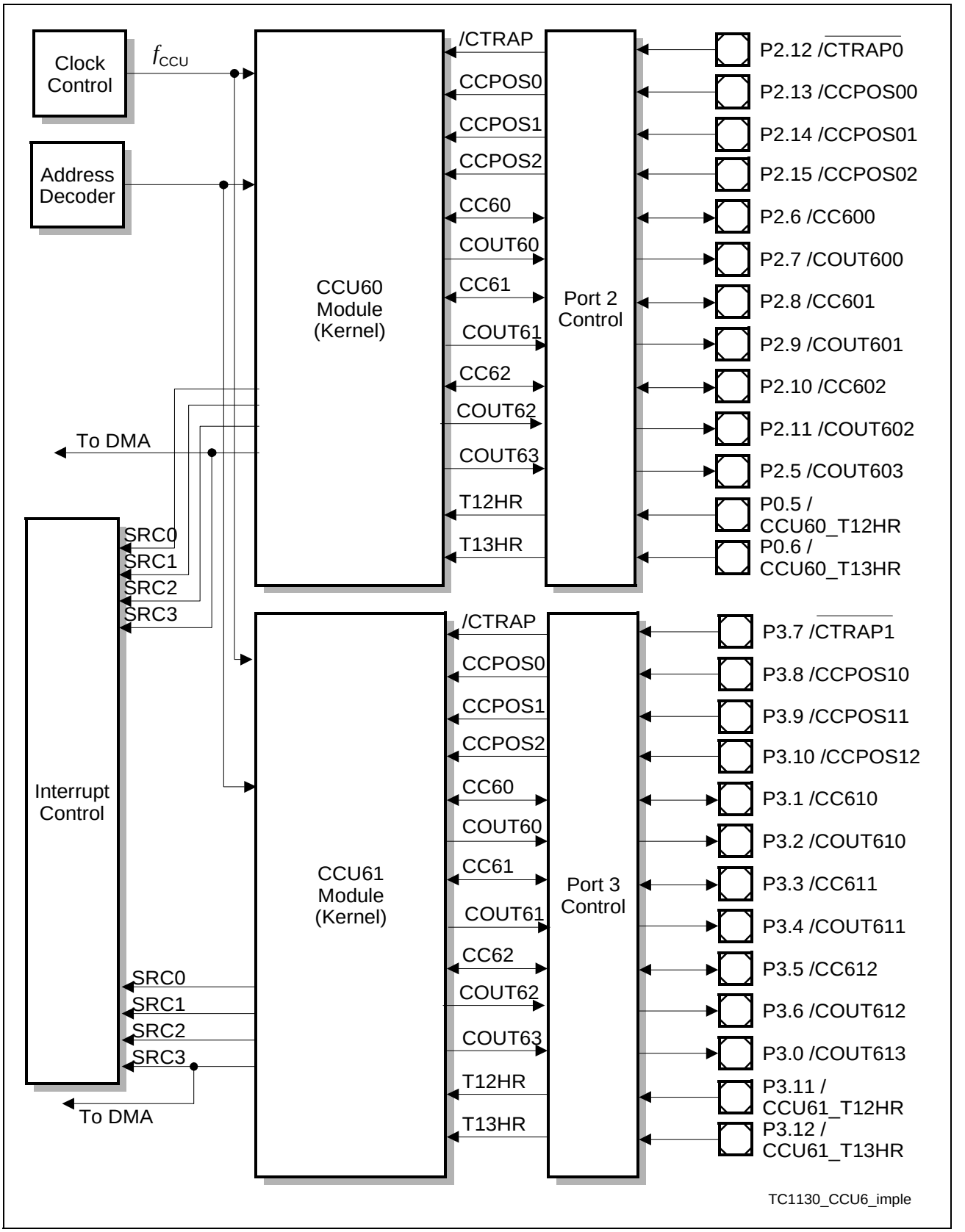


Figure 11 General Block Diagram of the CCU6

MultiCAN

Figure 12 shows a global view of all functional blocks of the MultiCAN module.

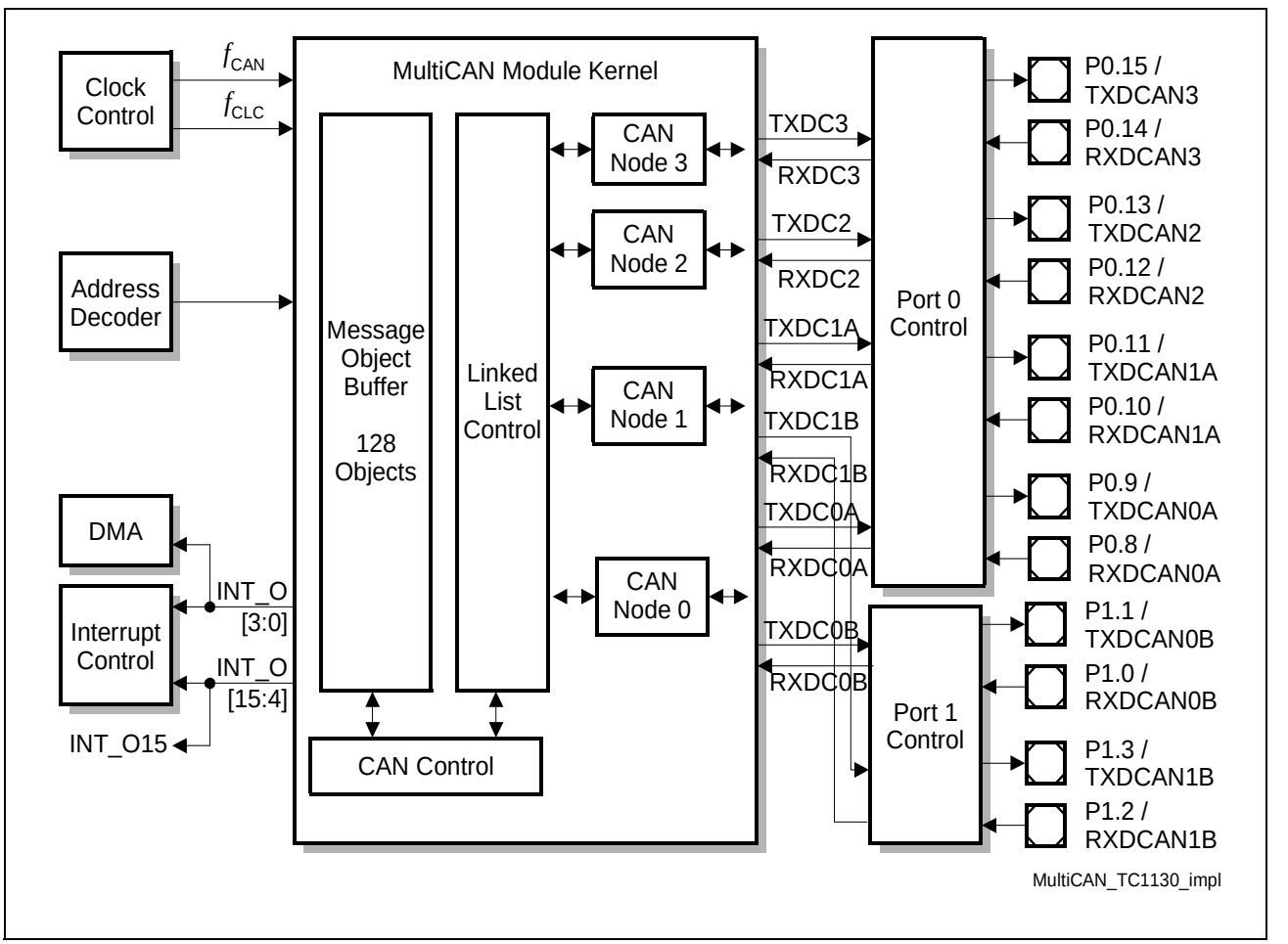


Figure 12 General Block Diagram of the MultiCAN Interfaces

The MultiCAN module contains 4 Full-CAN nodes operating independently or exchanging data and remote frames via a gateway function. Transmission and reception of CAN frames is handled in accordance to CAN specification V2.0 part B (active). Each CAN node can receive and transmit standard frames with 11-bit identifiers as well as extended frames with 29-bit identifiers.

All CAN nodes share a common set of message objects, where each message object may be individually allocated to one of the CAN nodes. Besides serving as a storage container for incoming and outgoing frames, message objects may be combined to build gateways between the CAN nodes or to setup a FIFO buffer.

The message objects are organized in double chained lists, where each CAN node has it's own list of message objects. A CAN node stores frames only into message objects that are allocated to the list of the CAN node. It only transmits messages from objects of this list.

A powerful, command driven list controller performs all list operations.

The bit timings for the CAN nodes are derived from the peripheral clock (f_{CAN}) and are programmable up to a data rate of 1 MBaud. A pair of receive and transmit pins connects each CAN node to a bus transceiver.

Features

- Compliant to ISO 11898.
- CAN functionality according to CAN specification V2.0 B active.
- Dedicated control registers are provided for each CAN node.
- A data transfer rate up to 1 MBaud is supported.
- Flexible and powerful message transfer control and error handling capabilities are implemented.
- Advanced CAN bus bit timing analysis and baud rate detection can be performed for each CAN node via the frame counter.
- Full-CAN functionality: A set of 128 message objects can be individually
  - allocated (assigned) to any CAN node
  - configured as transmit or receive object
  - setup to handle frames with 11-bit or 29-bit identifier
  - counted or assigned a timestamp via a frame counter
  - configured to remote monitoring mode
- Advanced Acceptance Filtering:
  - Each message object provides an individual acceptance mask to filter incoming frames.
  - A message object can be configured to accept only standard or only extended frames or to accept both standard and extended frames.
  - Message objects can be grouped into 4 priority classes.
  - The selection of the message to be transmitted first can be performed on the basis of frame identifier, IDE bit and RTR bit according to CAN arbitration rules.
- Advanced Message Object Functionality:
  - Message Objects can be combined to build FIFO message buffers of arbitrary size, which is only limited by the total number of message objects.
  - Message objects can be linked to form a gateway to automatically transfer frames between 2 different CAN buses. A single gateway can link any two CAN nodes. An arbitrary number of gateways may be defined.
- Advanced Data Management:
  - The Message objects are organized in double chained lists.
  - List reorganizations may be performed any time, even during full operation of the CAN nodes.
  - A powerful, command driven list controller manages the organization of the list structure and ensures consistency of the list.
  - Message FIFOs are based on the list structure and can easily be scaled in size during CAN operation.

  - Static Allocation Commands offer compatibility with TwinCAN applications, which are not list based.
- Advanced Interrupt Handling:
  - Up to 16 interrupt output lines are available. Most interrupt requests can be individually routed to one of the 16 interrupt output lines.
  - Message postprocessing notifications can be flexibly aggregated into a dedicated register field of 256 notification bits.

Ethernet Controller

The MAC controller implements the IEEE 802.3 and operates either at 100 Mbps or 10 Mbps. **Figure 13** shows a global view of the Ethernet Controller module with the module specific interface connections.

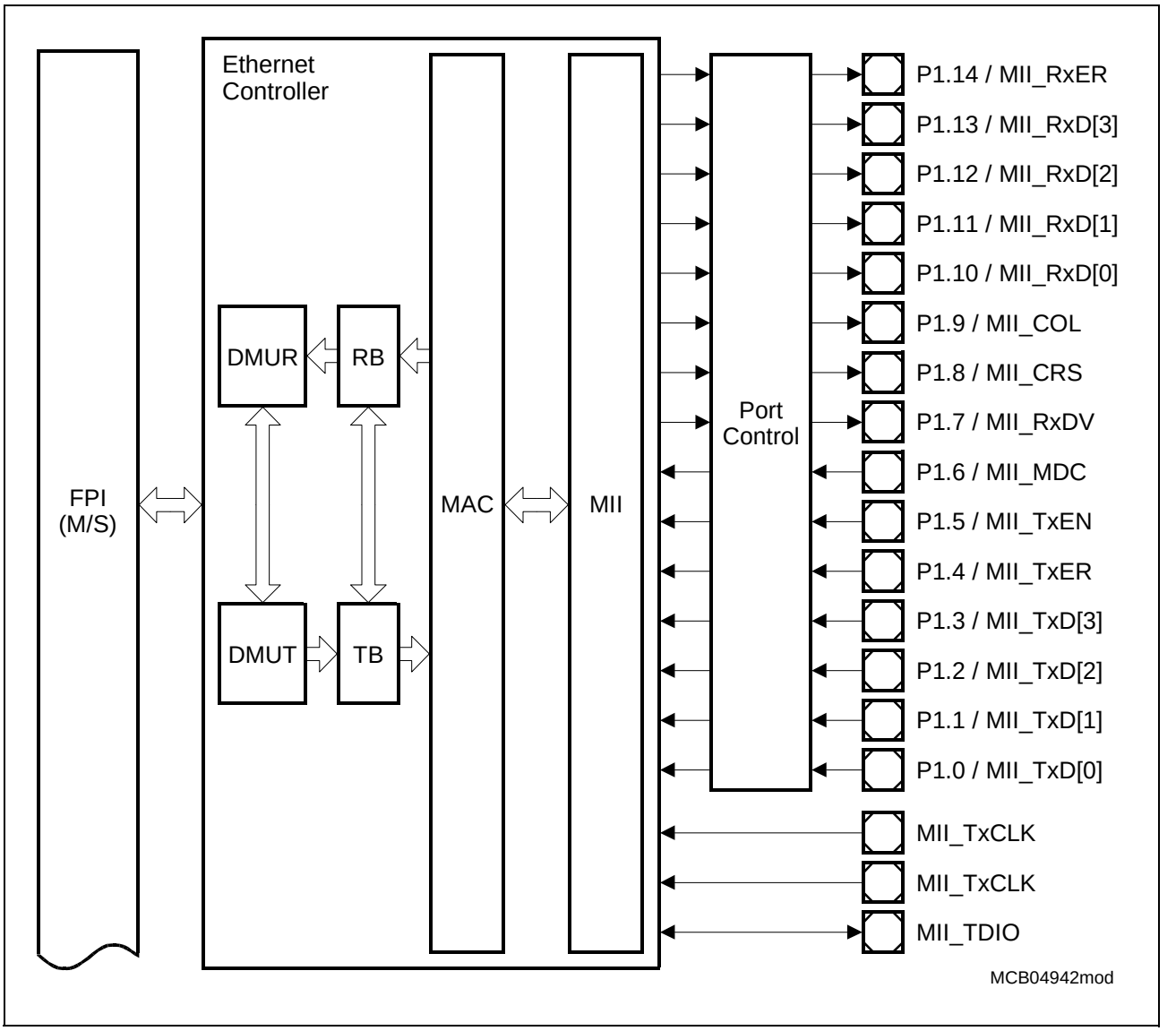


Figure 13 General Block Diagram of the Ethernet Controller

The Ethernet controller comprises the following functional blocks:

1. Media Access Controller (MAC)
2. Receive Buffer (RB)
3. Transmit Buffer (TB)
4. Data Management Unit in Receive Direction (DMUR)
5. Data Management Unit in Transmit Direction (DMUT)

RB as well as TB provides on-chip data buffering whereas DMUR and DMUT perform data transfer from/to the shared memory.

Two interfaces are provided by the Ethernet Controller Module:

1. MII interface for connection of Ethernet PHYs via eighteen Input / Output lines
2. Master/slave FPI bus interface for connection to the on-chip system bus for data transfer as well as configuration.

Features

- Media Independent Interface (MII) according to IEEE 802.3
- Support 10 or 100 Mbps MII-based Physical devices.
- Support Full Duplex Ethernet.
- Support data transfer between Ethernet Controller and COM-DRAM.
- Support data transfer between Ethernet Controller and SDRAM via EBU.
- 256 x 32 bit Receive buffer and Transmit buffer each.
- Support burst transfers up to 8 x 32 Byte.

Media Access Controller (MAC)

- 100/10-Mbps operations
- Full IEEE 802.3 compliance
- Station management signaling
- Large on-chip CAM (Content Addressable Memory)
- Full duplex mode
- 80-byte transmit FIFO
- 16-byte receive FIFO
- PAUSE Operation
- Flexible MAC Control Support
- Support Long Packet Mode and Short Packet Mode
- PAD generation

Media Independent Interface (MII)

- Media independence.
- Multi-vendor point of interoperability.
- Support connection of MAC layer and Physical (PHY) layer devices.
- Capable of supporting both 100 Mb/s and 10 Mb/s data rates.
- Data and delimiters are synchronous to clock references.
- Provides independent four bit wide transmit and receive data paths.
- Support connection of PHY layer and Station Management (STA) devices.
- Provides a simple management interface.
- Capable of driving a limited length of shielded cable.

On-Chip Memories

The TC1130 provides the following on-chip memories:

- Program Memory Interface (PMI) with
  - 32 KBytes Scratch-pad Code RAM (SRAM)
  - 16 KBytes Instruction Cache Memory (I-CACHE)
- Data Memory Interface (DMI) with
  - 28 KBytes Scratch-pad Data RAM (SRAM)
  - 4 KBytes Data Cache Memory (D-CACHE)
- Data Memory Unit (DMU) with
  - 64 KBytes SRAM
- 16 KBytes Boot ROM (BROM)

Address Map

Table 2 defines the specific segment oriented address blocks of the TC1130 with its address range, size, and PMI/DMI access view. Table 3 shows the block address map of the Segment 15 which includes on-chip peripheral units and ports.

Table 2 TC1130 Block Address Map

Segment	Address Range	Size	Description	DMI Acc.	PMI Acc.	
0 – 7	$0000\ 0000_H$ – $7FFF\ FFFF_H$	2 GB	MMU Space	via FPI	via FPI	cached
8	$8000\ 0000_H$ – $8FFF\ FFFF_H$	256 MB	External Memory Space mapped from Segment 10	via LMB	via LMB	
9	$9000\ 0000_H$ – $9FDF\ FFFF_H$	256 MB	Reserved	via FPI	via FPI	
10	$A000\ 0000_H$ – $AFBF\ FFFF_H$	252 MB	External Memory Space	via LMB	via LMB	non-cached
	$AFC0\ 0000_H$ – $AFC0\ FFFF_H$	64 KB	DMU Space			
	$AFC1\ 0000_H$ – $AFFF\ FFFF_H$	~4 MB	Reserved			
11	$B000\ 0000_H$ – $BFFF\ FFFF_H$	256 MB	Reserved	via FPI	via FPI	
12	$C000\ 0000_H$ – $C000\ FFFF_H$	64 KB	DMU	via LMB	via LMB	cached
	$C001\ 0000_H$ – $CFFF\ FFFF_H$	~ 256 MB	Reserved			

Table 2 TC1130 Block Address Map(cont'd)

Seg-ment	Address Range	Size	Description	DMI Acc.	PMI Acc.	
13	$D000\ 0000_H$ – $D000\ 7FFF_H$	32 KB	DMI Local Data RAM (LDRAM)	DMI local	via LMB	non-cached
	$D000\ 8000_H$ – $D3FF\ FFFF_H$	~ 64 MB	Reserved			
	$D400\ 0000_H$ – $D400\ 7FFF_H$	32 KB	PMI Local Code Scratchpad RAM (SPRAM)	via LMB	PMI local	
	$D400\ 8000_H$ – $D7FF\ FFFF_H$	~64 MB	Reserved			
	$D800\ 0000_H$ – $DDFF\ FFFF_H$	96 MB	External Memory Space	via LMB	via LMB	
	$DE00\ 0000_H$ – $DEFF\ FFFF_H$	16 MB	Emulator Memory Space			
	$DF00\ 0000_H$ – $DFFF\ BFFF_H$	~16 MB	Reserved	–	–	
	$DFFF\ C000_H$ – $DFFF\ FFFF_H$	16 KB	Boot ROM Space	via FPI	via FPI	
14	$E000\ 0000_H$ – $E7FF\ FFFF_H$	128 MB	External Memory Space	via LMB	via LMB	
	$E800\ 0000_H$ – $E83F\ FFFF_H$	4 MB	Reserved for mapped space for lower 4 MByte of Local Memory in segment 12 (Transformed by LFI bridge to $C000\ 0000_H$ – $C03F\ FFFF_H$)	–	–	
	$E840\ 0000_H$ – $E84F\ FFFF_H$	1 MB	Reserved for mapped space for lower 1 MByte of Local Memory in segment 13 (Transformed by LFI bridge to $D000\ 0000_H$ – $D00F\ FFFF_H$)	access only from FPI bus side of LFI	access only from FPI bus side of LFI	
	$E850\ 0000_H$ – $E85F\ FFFF_H$	1 MB	Reserved for mapped space for 1 MByte of Local Memory in segment 13 (Transformed by LFI bridge to $D400\ 0000_H$ – $D40F\ FFFF_H$)			
	$E860\ 0000_H$ – $EFFF\ FFFF_H$	122 MB	Reserved	–	–	

Table 2 TC1130 Block Address Map(cont'd)

Seg-ment	Address Range	Size	Description	DMI Acc.	PMI Acc.	
15	$F000\ 0000_H$ – $F00F\ FFFF_H$	1 MB	On-Chip System Peripherals & Ports	via FPI	via FPI	non-cached
	$F010\ 0000_H$ – $F027\ FFFF_H$	1.5 MB	Peripherals on SMIF Interface of DMA Controller			
	$F028\ 0000_H$ – $F200\ 00FF_H$	~29.5 MB	Reserved	–	–	
	$F200\ 0100_H$ – $F200\ 05FF_H$	1280 Bytes	Ethernet Controller Registers	via FPI	via FPI	
	$F200\ 0600_H$ – $F7E0\ FEFF_H$	~94 MB	Reserved	–	–	
	$F7E0\ FF00_H$ – $F7E0\ FFFF_H$	256 Bytes	CPU Slave Interface Registers (CPS)	via LMB	via LMB	
	$F7E1\ 0000_H$ – $F7E1\ FFFF_H$	64 KB	Core SFRs			
	$F7E2\ 0000_H$ – $F7FF\ FFFF_H$	~1.8 MB	Reserved	–	–	
	$F800\ 0000_H$ – $F87F\ FFFF_H$	8 MB	LMB Peripheral Space (EBU and local memory DMU control registers)	via LMB	via LMB	
	$F880\ 0000_H$ – $FFFF\ FFFF_H$	120 MB	Reserved	–	–	

Table 3 Block Address Map of Segment 15

Symbol	Description	Address Range	Size
System Peripheral Bus (SPB)			
SCU	System Control Unit (incl. WDT)	$F000\ 0000_H$ - $F000\ 00FF_H$	256 Bytes
SBCU	FPI Bus Control Unit	$F000\ 0100_H$ - $F000\ 01FF_H$	256 Bytes
STM	System Timer	$F000\ 0200_H$ - $F000\ 02FF_H$	256 Bytes
OCDS	On-Chip Debug Support (Cerberus)	$F000\ 0300_H$ - $F000\ 03FF_H$	256 Bytes
–	Reserved	$F000\ 0400_H$ - $F000\ 04FF_H$	256 Bytes
–	Reserved	$F000\ 0500_H$ - $F000\ 05FF_H$	256 Bytes

Table 3 Block Address Map of Segment 15(cont'd)

Symbol	Description	Address Range	Size
GPTU	General Purpose Timer Unit	F000 0600$_H$ - F000 06FF$_H$	256 Bytes
–	Reserved	F000 0700$_H$ - F000 07FF$_H$	256 Bytes
–	Reserved	F000 0800$_H$ - F000 08FF$_H$	256 Bytes
–	Reserved	F000 0900$_H$ - F000 09FF$_H$	256 Bytes
–	Reserved	F000 0A00$_H$ - F0000AFF$_H$	256 Bytes
–	Reserved	F000 0B00$_H$ - F0000BFF$_H$	256 Bytes
P0	Port 0	F000 0C00$_H$ -F0000CFF$_H$	256 Bytes
P1	Port 1	F000 0D00$_H$ -F0000DFF$_H$	256 Bytes
P2	Port 2	F000 0E00$_H$ -F000 0EFF$_H$	256 Bytes
P3	Port 3	F000 0F00$_H$ - F000 0FFF$_H$	256 Bytes
P4	Port 4	F000 1000$_H$ - F000 10FF$_H$	256 Bytes
–	Reserved	F000 1100$_H$ - F000 11FF$_H$	256 Bytes
–	Reserved	F000 1200$_H$ - F000 12FF$_H$	256 Bytes
–	Reserved	F000 1300$_H$ - F000 13FF$_H$	256 Bytes
–	Reserved	F000 1400$_H$ - F000 14FF$_H$	256 Bytes
–	Reserved	F000 1500$_H$ - F000 15FF$_H$	256 Bytes
–	Reserved	F000 1600$_H$ - F000 16FF$_H$	256 Bytes
–	Reserved	F000 1700$_H$ - F000 17FF$_H$	256 Bytes
–	Reserved	F000 1800$_H$ - F000 18FF$_H$	256 Bytes
–	Reserved	F000 1900$_H$ - F000 19FF$_H$	256 Bytes
CCU60	Capture/Compare Unit 0	F000 2000$_H$ - F000 20FF$_H$	256 Bytes
CCU61	Capture/Compare Unit 1	F000 2100$_H$ - F000 21FF$_H$	256 Bytes
–	Reserved	F000 2200$_H$ - F000 3BFF$_H$	–
DMA	Direct Memory Access Controller	F000 3C00$_H$ - F0003EFF$_H$	3 × 256 Bytes
–	Reserved	F000 3F00$_H$ - F000 3FFF$_H$	–
CAN	MultiCAN Controller	F000 4000$_H$ - F000 5FFF$_H$	8 KBytes
–	Reserved	F000 6000$_H$ - F00E1FFF$_H$	–
USB	USB RAM based Registers	F00E 2000$_H$ - F00E 219F$_H$	416 Bytes
USB	USB RAM	F00E 21A0$_H$ - F00E 27FF$_H$	1.6 KBytes
USB	USB Registers	F00E 2800$_H$ - F00E 28FF$_H$	256 Bytes

Table 3 Block Address Map of Segment 15(cont'd)

Symbol	Description	Address Range	Size
–	Reserved	F00E 2900$_H$ - F00F FFFF$_H$	–
Units on SMIF Interface of DMA Controller			
–	Reserved	F010 0000$_H$ - F010 00FF$_H$	256 Byte
SSC0	Synchronous Serial Interface 0	F010 0100$_H$ - F010 01FF$_H$	256 Byte
SSC1	Synchronous Serial Interface 1	F010 0200$_H$ - F010 02FF$_H$	256 Byte
ASC0	Async./Sync. Serial Interface 0	F010 0300$_H$ - F010 03FF$_H$	256 Byte
ASC1	Async./Sync. Serial Interface 1	F010 0400$_H$ - F010 04FF$_H$	256 Byte
ASC2	Async./Sync. Serial Interface 2	F010 0500$_H$ - F010 05FF$_H$	256 Byte
I2C	Inter IC	F010 0600$_H$ - F010 06FF$_H$	256 Byte
–	Reserved	F010 0700$_H$ - F010BFFF$_H$	–
MLI0	Multi Link Interface 0	F010 C000$_H$ -F010C0FF$_H$	256 Bytes
MLI1	Multi Link Interface 1	F010 C100$_H$ -F010C1FF$_H$	256 Bytes
MCHK	Memory Checker	F010 C200$_H$ -F010C2FF$_H$	256 Bytes
–	Reserved	F010 C300$_H$-F01D FFFF$_H$	–
MLI0_ SP0	MLI0 Small Transfer Window 0	F01E 0000$_H$- F01E 1FFF$_H$	8 KBytes
MLI0_ SP1	MLI0 Small Transfer Window 1	F01E 2000$_H$- F01E 3FFF$_H$	8 KBytes
MLI0_ SP2	MLI0 Small Transfer Window 2	F01E 4000$_H$- F01E 5FFF$_H$	8 KBytes
MLI0_ SP3	MLI0 Small Transfer Window 3	F01E 6000$_H$- F01E 7FFF$_H$	8 KBytes
MLI1_ SP0	MLI1 Small Transfer Window 0	F01E 8000$_H$- F01E 9FFF$_H$	8 KBytes
MLI1_ SP1	MLI1 Small Transfer Window 1	F01E A000$_H$- F01E BFFF$_H$	8 KBytes
MLI1_ SP2	MLI1 Small Transfer Window 2	F01E C000$_H$- F01E DFFF$_H$	8 KBytes
MLI1_ SP3	MLI1 Small Transfer Window 3	F01E E000$_H$- F01E FFFF$_H$	8 KBytes
–	Reserved	F01F 0000$_H$- F01F FFFF$_H$	–
MLI0_ LP0	MLI0 Large Transfer Window 0	F020 0000$_H$ - F020 FFFF$_H$	64 K Bytes

Table 3 Block Address Map of Segment 15(cont'd)

Symbol	Description	Address Range	Size
MLI0_ LP1	MLI0 Large Transfer Window 1	F021 0000$_H$ - F021 FFFF$_H$	64 K Bytes
MLI0_ LP2	MLI0 Large Transfer Window 2	F022 0000$_H$ - F022 FFFF$_H$	64 K Bytes
MLI0_ LP3	MLI0 Large Transfer Window 3	F023 0000$_H$ - F023 FFFF$_H$	64 K Bytes
MLI1_ LP0	MLI1 Large Transfer Window 0	F024 0000$_H$ - F024 FFFF$_H$	64 K Bytes
MLI1_ LP1	MLI1 Large Transfer Window 1	F025 0000$_H$ - F025 FFFF$_H$	64 K Bytes
MLI1_ LP2	MLI1 Large Transfer Window 2	F026 0000$_H$ - F026 FFFF$_H$	64 K Bytes
MLI1_ LP3	MLI1 Large Transfer Window 3	F027 0000$_H$ - F027 FFFF$_H$	64 K Bytes
–	Reserved	F028 0000$_H$ - F200 00FF$_H$	–
ECU	Ethernet Controller Unit	F200 0100$_H$ - F200 05FF$_H$	1280Bytes
–	Reserved	F200 0600$_H$ - F7E0FEFF$_H$	–
CPU (Part of System Peripheral Bus)			
CPU SFRs	CPU Slave Interface	F7E0 FF00$_H$ -F7E0FFFF$_H$	256 Bytes
	Reserved	F7E1 0000$_H$ –F7E17FFF$_H$	–
	MMU	F7E1 8000$_H$ –F7E180FF$_H$	256 Bytes
	Reserved	F7E1 8100$_H$ -F7E1BFFF$_H$	–
	Memory Protection Registers	F7E1 C000$_H$-F7E1EFFF$_H$	12K Bytes
	Reserved	F7E1 F000$_H$- F7E1FCFF$_H$	–
	Core Debug Register (OCDS)	F7E1 FD00$_H$-F7E1FDFF$_H$	256 Bytes
	Core Special Function Registers (CSFRs)	F7E1 FE00$_H$-F7E1FEFF$_H$	256 Bytes
	General Purpose Register (GPRs)	F7E1 FF00$_H$-F7E1 FFFF$_H$	256 Bytes
–	Reserved	F7E2 0000$_H$ -F7FFFFFF$_H$	–
Local Memory Buses (LMB)			
EBU	External Bus Interface Unit	F800 0000$_H$ - F800 03FF$_H$	1KBytes

Table 3 Block Address Map of Segment 15(cont'd)

Symbol	Description	Address Range	Size
DMU	Data Memory Unit	F800 0400_H - F800 $04FF_H$	256 Bytes
-	Reserved	F800 0500_H -F87F $FBFF_H$	–
DMI	Data Memory Interface Unit	F87F $FC00_H$-F87F $FCFF_H$	256 Bytes
PMI	Program Memory Interface Unit	F87F $FD00_H$-F87F $FDFF_H$	256 Bytes
LBCU	Local Memory Bus Control Unit	F87F $FE00_H$ - F87F $FEFF_H$	256 Bytes
LFI	LMB to FPI Bus Bridge	F87F $FF00_H$ - F87F $FFFF_H$	256 Bytes
–	Reserved	F880 0000_H - FFFF $FFFF_H$	–

Memory Protection System

The TC1130 memory protection system specifies the addressable range and read/write permissions of memory segments available to the currently executing task. The memory protection system controls the position and range of addressable segments in memory. It also controls the kinds of read and write operations allowed within addressable memory segments. Any illegal memory access is detected by the memory protection hardware, which then invokes the appropriate Trap Service Routine (TSR) to handle the error. Thus, the memory protection system protects critical system functions against both software and hardware errors. The memory protection hardware can also generate signals to the Debug Unit to facilitate tracing illegal memory accesses.

In TC1130, TriCore supports two address spaces: The virtual address space and The physical address space. Both address space are 4GB in size and divided into 16 segments with each segment being 256MB. The upper 4 bits of the 32-bit address are used to identify the segment. Virtual segments are numbered 0 - 15. But a virtual address is always translated into a physical address before accessing memory. The virtual address is translated into a physical address using one of two translation mechanisms: (a) direct translation, and (b) Page Table Entry (PTE) based translation. If the virtual address belongs to the upper half of the virtual address space then the virtual address is directly used as the physical address (direct translation). If the virtual address belongs to the lower half of the address space, then the virtual address is used directly as the physical address if the processor is operating in Physical mode (direct translation) or translated using a Page Table Entry if the processor is operating in Virtual mode (PTE translation). These are managed by Memory Management Unit (MMU)

Memory protection is enforced using separate mechanisms for the two translation paths.

Protection for direct translation

Memory protection for addresses that undergo direct translation is enforced using the range based protection that has been used in the previous generation of the TriCore architecture. The range based protection mechanism provides support for protecting

memory ranges from unauthorized read, write, or instruction fetch accesses. The TriCore architecture provides up to four protection register sets with the PSW.PRS field controlling the selection of the protection register set. Because the TC1130 uses a Harvard-style memory architecture, each Memory Protection Register Set is broken down into a Data Protection Register Set and a Code Protection Register Set. Each Data Protection Register Set can specify up to four address ranges to receive particular protection modes. Each Code Protection Register Set can specify up to two address ranges to receive particular protection modes.

Each of the Data Protection Register Sets and Code Protection Register Sets determines the range and protection modes for a separate memory area. Each contains register pairs which determine the address range (the Data Segment Protection Registers and Code Segment Protection Registers) and one register (Data Protection Mode Register) which determines the memory access modes which apply to the specified range.

Protection for PTE based translation

Memory protection for addresses that undergo PTE based translation is enforced using the PTE used for the address translation. The PTE provides support for protecting a process from unauthorized read, write, or instruction fetches by other processes. The PTE has the following bits that are provided for the purpose of protection:

- XE (Execute Enable) enables instruction fetch to the page.
- WE (Write Enable) enables data writes to the page.
- RE (Read Enable) enables data reads from the page.

Furthermore, User-0 accesses to virtual addresses in the upper half of the virtual address space are disallowed when operating in Virtual mode. In Physical mode, User-0 accesses are disallowed only to segments 14 and 15. Any User-0 access to a virtual address that is restricted to User-1 or Super-visor mode will cause a Virtual Address Protection (VAP) Trap in both the Physical and Virtual modes.

Memory Checker

The Memory Checker Module (MCHK) allows to check the data consistency of memories. It uses DMA moves to read from the selected address area and to write the value read in a memory checker input register (the moves should be 32 bit moves). A polynomial checksum calculation is done with each write operation to the memory checker input register

On-Chip Bus System

The TC1130 includes two bus systems:

- Local Memory Bus (LMB)
- On-Chip FPI Bus (FPI)

The LMB-to-FPI (LFI) bridge interconnects the FPI bus and LMB Bus.

Local Memory Bus (LMB)

The Local Memory Bus interconnects the memory units and functional units, such as CPU and DMU. The main target of the LMB bus is to support devices with fast response times, optimized for speed. This allows the DMI and PMI fast access to local memory and reduces load on the FPI bus. The Tricore system itself is located on LMB bus. Via External Bus Unit, it interconnects TC1130 and external components.

The Local Memory Bus is a synchronous, pipelined, split bus with variable block size transfer support. It supports 8, 16, 32 & 64 bits single beat transactions and variable length 64 bits block transfers.

Key Features

The LMB provides the following features:

- Synchronous, Pipelined, Multi-master, 64-bit high performance bus
- Optimized for high speed and high performance
- 32 bit address, 64 bit data busses
- Support Split transactions
- Support Variable block size transfer
- Burst Mode Read/Write to Memories
- Connect Caches and on-chip memory and FPI Bus
- Slave controlled wait state insertion
- Support Locked transaction (read-modify-write)

On-Chip FPI Bus

The FPI Bus interconnects the functional units of the TC1130, such as the DMA and on-chip peripheral components. The FPI Bus is designed to be quick to acquire by on-chip functional units, and quick to transfer data. The low setup overhead of the FPI Bus access protocol guarantees fast FPI Bus acquisition, which is required for time-critical applications.The FPI Bus is designed to sustain high transfer rates. For example, a peak transfer rate of up to 800 MBytes/s can be achieved with a 100 MHz bus clock and 32-bit data bus. Multiple data transfers per bus arbitration cycle allow the FPI Bus to operate at close to its peak bandwidth.

Features

- Supports multiple bus masters
- Supports demultiplexed address/data operation
- Address bus up to 32 bits and data buses are 64 bits wide
- Data transfer types include 8-, 16-, 32- and 64 bit sizes
- Supports Burst transfer
- Single- and multiple-data transfers per bus acquisition cycle
- Designed to minimize EMI and power consumption
- Controlled by an Bus Control Unit (BCU)
  - Arbitration of FPI Bus master requests
  - Handling of bus error.

LFI

The LMB-to-FPI Interface (LFI) block provides the circuitry to interface (bridge) the FPI bus to the Local Memory Bus (LMB).

LFI Features

- Compatible with the FPI 3.2 and LMB bus Specification V2.4
- Supports Burst/Single transactions, from FPI to LMB.
- Supports Burst/Single transactions, from LMB to FPI
- High efficiency and performance:
  - fastest access across the bridge takes three cycles, using a bypass.
  - There are no dead cycles on arbitration.
- Acts as the default master on FPI side.
- Supports abort, error and retry conditions on both sides of the bridge.
- Supports FPI’s clock the same, or half, as the LMB’s clock frequency.
- LMB clock is shut when no transactions are issue to LFI from both buses and none are in process in the LFI to minimize the power consumption.

LMB External Bus Unit

The LMB External Bus Control Unit (EBU) of the TC1130 is the interface between external resources, like memories and peripheral units, and the internal resources connected to on-chip buses if enabled. The basic structure and external interconnections of the EBU are shown in **Figure 14**.

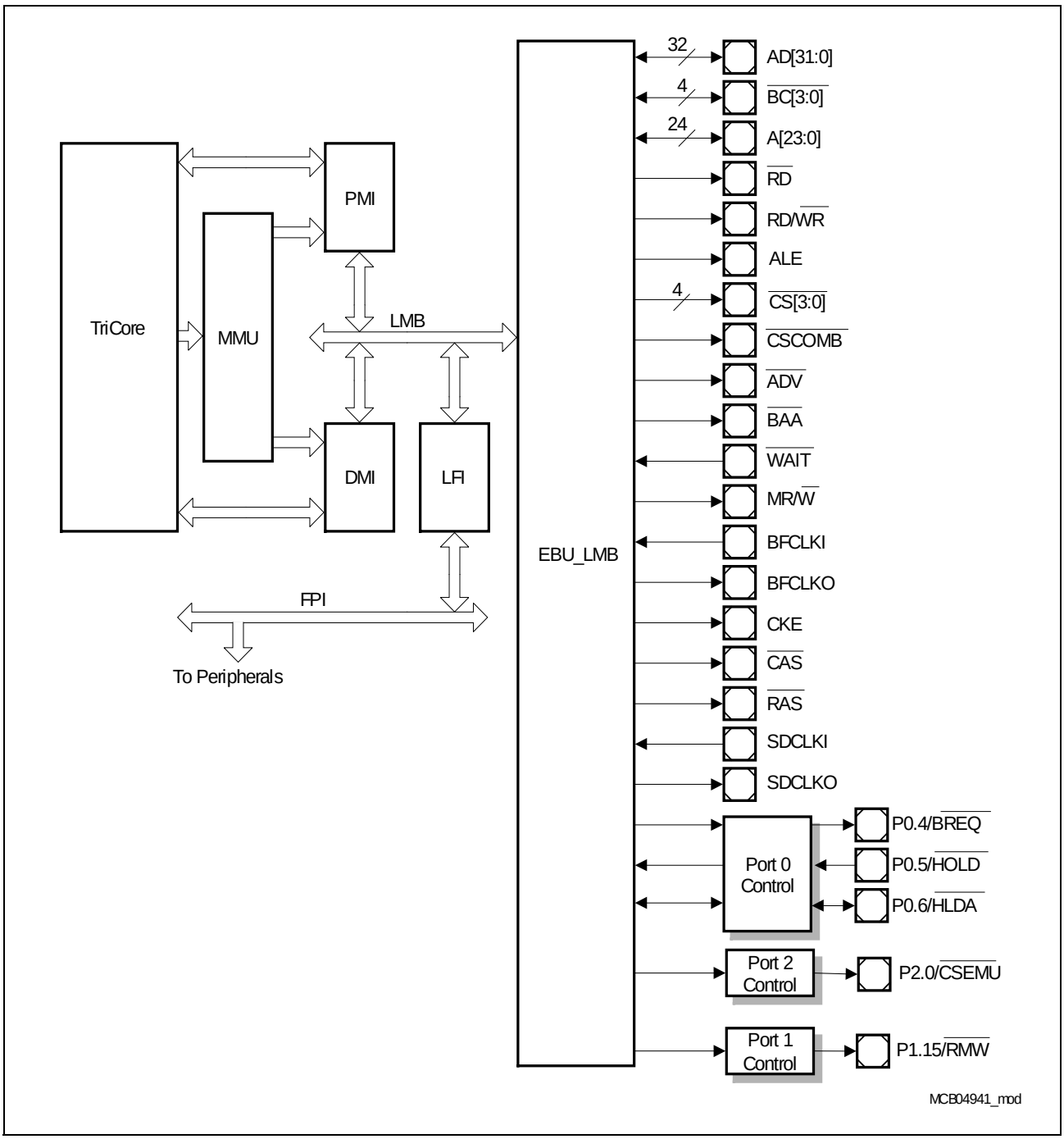


Figure 14 EBU Structure and Interfaces

The EBU is mainly used for the operation that masters on LMB bus access external memories through EBU.

The EBU controls all transactions required for this operations and in particular handles the arbitration of the external bus between multi-masters.

The types of external resources accessed by the EBU are:

- INTEL style peripherals (separate $\overline{RD}$ and $\overline{WR}$ signals)
- ROMs, EPROMs
- Static RAMs
- PC 100 SDRAMs (Burst Read/Write Capacity / Multi-Bank/Page support)
- Specific types of Burst Mode Flashes (Intel 28F800F3/28F160F3, AMD 29BL162)
- Special support for external emulator/debug hardware

Features

- Support Local Memory Bus (LMB 64-bit)
- Support External bus frequency: LMB frequency =1:1 or 1:2
- Highly programmable access parameters
- Support Intel-style peripherals/devices
- Support PC 100 SDRAM (burst access, multibanking, precharge, refresh)
- Support 16-and 32-bit SDRAM data bus and 64,128 and 256MBit devices
- Support Burst flash (Intel 28F800F3/160F3,AMD 29BL162)
- Support Multiplexed access (address &data on the same bus) when PC 100 SDRAM is not implemented
- Support Data Buffering: Code Prefetch Buffer, Read/Write Buffer.
- External master arbitration compatible to C166 and other Tricore devices
- 4 programmable address regions (1 dedicated for emulator)
- Support Little-endian
- Signal for controlling data flow of slow-memory buffer

Direct Memory Access (DMA)

The Direct Memory Access Controller executes DMA transactions from a source address location to a destination address location, without intervention of the CPU. One DMA transaction is controlled by one DMA channel. Each DMA channel has assigned its own channel register set. The total of 8 channels are provided by one DMA sub-block.

The DMA module is connected to 3 bus interfaces in TC1130, the Flexible Peripheral Interconnect Bus (FPI), the DMA Bus and the Micro Link Bus. It can do transfers on each of the buses as well as between the buses.

In addition it bridges accesses from the Flexible Peripheral Interconnect Bus to the peripherals on the DMA Bus, allowing easy access to these peripherals by CPU. Clock control, address decoding, DMA request wiring, and DMA interrupt service request control are implementation specific and managed outside the DMA controller kernel.

Features

- 8 independent DMA channels
  - Up to 8 selectable request inputs per DMA channel
  - Programmable priority of DMA channels within a DMA sub-block (2 levels)
  - Software and hardware DMA request generation
  - Hardware requests by selected peripherals and external inputs
- Programmable priority of the DMA sub-block on the bus interfaces
- Buffer capability for move actions on the buses (min. 1 move per bus is buffered).
- Individually programmable operation modes for each DMA channel
  - Single mode: stops and disables DMA channel after a predefined number of DMA transfers
  - Continuous mode: DMA channel remains enabled after a predefined number of DMA transfers; DMA transaction can be repeated.
  - Programmable address modification
- Full 32-bit addressing capability of each DMA channel
  - 4 GByte address range
  - Support of circular buffer addressing mode
- Programmable data width of a DMA transaction: 8-bit, 16-bit, or 32-bit
- Micro Link supported
- Register set for each DMA channel
  - Source and destination address register
  - Channel control and status register
  - Transfer count register
- Flexible interrupt generation (the service request node logic for the MLI channels is also implemented in the DMA module)
- All buses/interfaces connected to the DMA module must work at the same frequency.
- Read/write requests of the System Bus Side to the Remote Peripherals are bridged to the DMA Bus (only the DMA is master on the DMA bus)

The basic structure and external interconnections of the DMA are shown in **Figure 15**

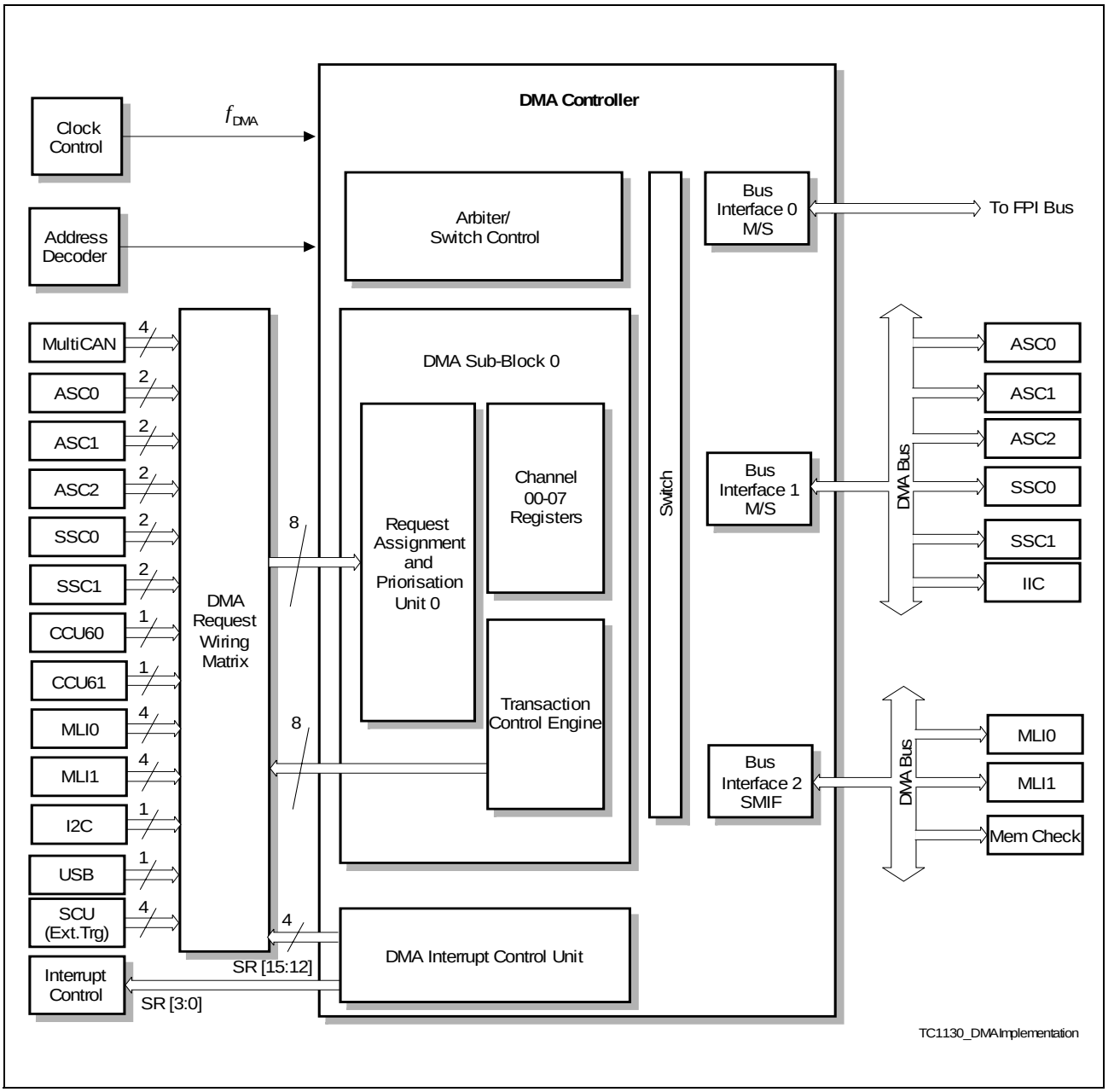


Figure 15 DMA Controller Structure and Interconnections

System Timer

The STM within the TC1130 is designed for global system timing applications requiring both high precision and long range. The STM provides the following features:

- Free-running 56-bit counter
- All 56 bits can be read synchronously
- Different 32-bit portions of the 56-bit counter can be read synchronously
- Flexible interrupt generation on partial STM content compare match
- Driven by clock f_{STM} after reset (default after reset is $f_{STM} = f_{SYS} = 150$ MHz)
- Counting starts automatically after a reset operation
- STM is reset under following reset causes:
  - Wake-up reset (PMG_CON.DSRW must be set)
  - Software reset (RST_REQ.RRSTM must be set)
  - Power-on reset
- STM (and clock divider) is not reset at watchdog reset and hardware reset ($\overline{\text{HDRST}}$ = 0)

The STM is an upward counter, running with the system clock frequency f_{SYS} (after reset $f_{STM} = f_{SYS}$). It is enabled per default after reset, and immediately starts counting up. Other than via reset, it is no possible to affect the contents of the timer during normal operation of the application, it can only be read, but not written to. Depending on the implementation of the clock control of the STM, the timer can optionally be disabled or suspended for power-saving and debugging purposes via a clock control register

The maximum clock period is $2^{56}/f_{STM}$. At f_{STM} = 150 MHz (maximum), for example, the STM counts 15.2 years before overflowing. Thus, it is capable of continuously timing the entire expected product life-time of a system without overflowing.

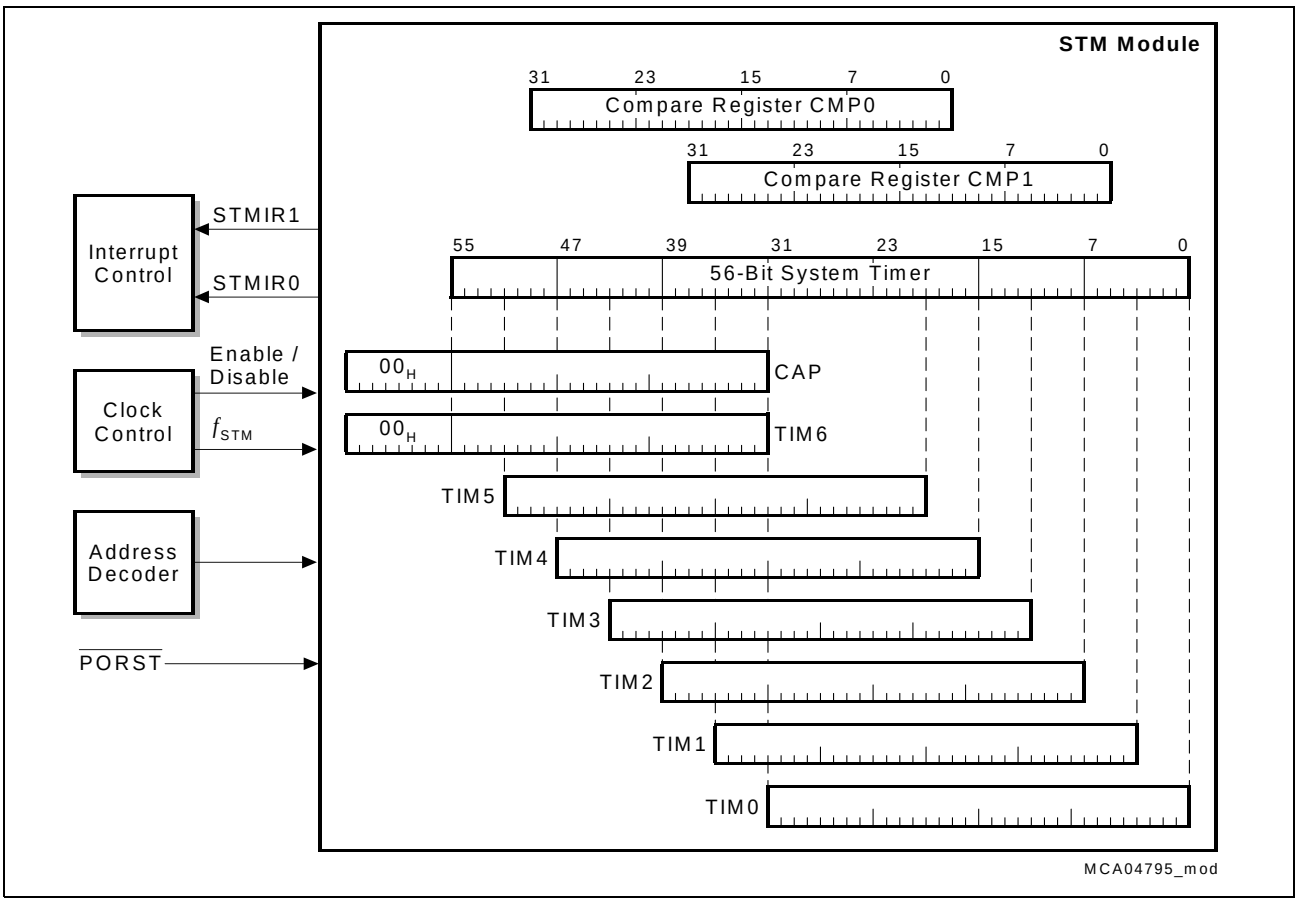


Figure 16 Block Diagram of the STM Module

Watchdog Timer

The Watchdog Timer (WDT) provides a highly reliable and secure way to detect and recover from software or hardware failure. The WDT helps to abort an accidental malfunction of the TC1130 in a user-specified time period. When enabled, the WDT will cause the TC1130 system to be reset if the WDT is not serviced within a user-programmable time period. The CPU must service the WDT within this time interval to prevent the WDT from causing a TC1130 system reset. Hence, routine service of the WDT confirms that the system is functioning properly.

In addition to this standard “Watchdog” function, the WDT incorporates the EndInit feature and monitors its modifications. A system-wide line is connected to the ENDINIT bit implemented in a WDT control register, serving as an additional write-protection for critical registers (besides Supervisor Mode protection). Registers protected via this line can only be modified when Supervisor Mode is active and bit ENDINIT = 0.

A further enhancement in the TC1130’s Watchdog Timer is its reset prewarning operation. Instead of immediately resetting the device on the detection of an error, as known from standard Watchdogs, the WDT first issues an Non-maskable Interrupt (NMI) to the CPU before finally resetting the device at a specified time period later. This gives the CPU a chance to save system state to memory for later examination of the cause of the malfunction, an important aid in debugging.

Features

- 16-bit Watchdog counter
- Selectable input frequency: $f_{SYS}/256$ or $f_{SYS}/16384$
- 16-bit user-definable reload value for normal Watchdog operation, fixed reload value for Time-Out and Prewarning Modes
- Incorporation of the ENDINIT bit and monitoring of its modifications
- Sophisticated password access mechanism with fixed and user-definable password fields
- Proper access always requires two write accesses. The time between the two accesses is monitored by the WDT and limited.
- Access Error Detection: Invalid password (during first access) or invalid guard bits (during second access) trigger the Watchdog reset generation.
- Overflow Error Detection: An overflow of the counter triggers the Watchdog reset generation.
- Watchdog function can be disabled; access protection and ENDINIT monitor function remain enabled.
- Double Reset Detection: If a Watchdog induced reset occurs twice without a proper access to its control register in between, a severe system malfunction is assumed and the TC1130 is held in reset until a power-on reset. This prevents the device from being periodically reset if, for instance, connection to the external memory has been lost such that even system initialization could not be performed.

- Important debugging support is provided through the reset prewarning operation by first issuing an NMI to the CPU before finally resetting the device after a certain period of time.

System Control Unit

The System Control Unit (SCU) of the TC1130 handles the system control tasks. All these system functions are tightly coupled, thus, they are conveniently handled by one unit, the SCU. The system tasks of the SCU are:

- PLL Control
  - PLL_CLC Clock Control Register
- Reset Control
  - Generation of all internal reset signals
  - Generation of external $\overline{\text{HDRST}}$ reset signal
- Boot Scheme
  - Hardware Booting Scheme
  - Software Booting Scheme
- Power Management Control
  - Enabling of several power-down modes
  - Control of the PLL in power-down modes
- Watchdog Timer
- OCDS2 Trace Port Control
- Device Identification Registers

Interrupt System

An interrupt request can be serviced by the CPU which is called “Service Provider”. Interrupt requests are referred as “Service Requests” in this document.

Each peripheral in the TC1130 can generate service requests. Additionally, the Bus Control Unit, the Debug Unit, the DMA Controller and even the CPU itself can generate service requests to the Service Provider. As shown in **Figure 17**, each unit that can generate service requests is connected to one or multiple Service Request Nodes (SRN). Each SRN contains a Service Request Control Register mod_SRCx, where “mod” is the identifier of the service requesting unit and “x” an optional index. The SRNs are connected to the Interrupt Control Unit (ICU) via the CPU Interrupt Arbitration Bus. The ICU arbitrates service requests for the CPU and administers the Interrupt Arbitration Bus.

Units which can generate service requests are:

- Asynchronous/Synchronous Serial Interfaces (ASC0 & ASC1 & ASC2) with 4 SRNs each
- High-Speed Synchronous Serial Interfaces (SSC0 & SSC1) with 3 SRNs each
- Inter IC Interface (IIC) with 3 SRNs
- Universal Serial Bus (USB) with 8 SRNs
- Micro Link Interface MLI0 with 4 SRNs and MLI1 with 2 SRNs
- General Purpose Timer Unit (GPTU) with 8 SRNs
- Capture/Compare Unit (CCU60 & CCU61) with 4 SRNs each
- MultiCAN (CAN) with 16 SRNs
- Ethernet Controller with 9 SRNs
- External Interrupts with 4 SRNs
- Direct Memory Access Controller (DMA) with 4 SRNs
- DMA Bus with 1 SRN
- System Timer (STM) with 2 SRNs
- Bus Control Units (SBCU and LBCU) with 1 SRN each
- Peripheral Control Processor (PCP) with 12 SRNs
- Central Processing Unit (CPU) with 4 SRNs
- Floating Point Unit (FPU) with 1 SRN
- Debug Unit (OCDS) with 1 SRN

The CPU can make service requests directly to itself (via the ICU). The CPU Service Request Nodes are activated through software.

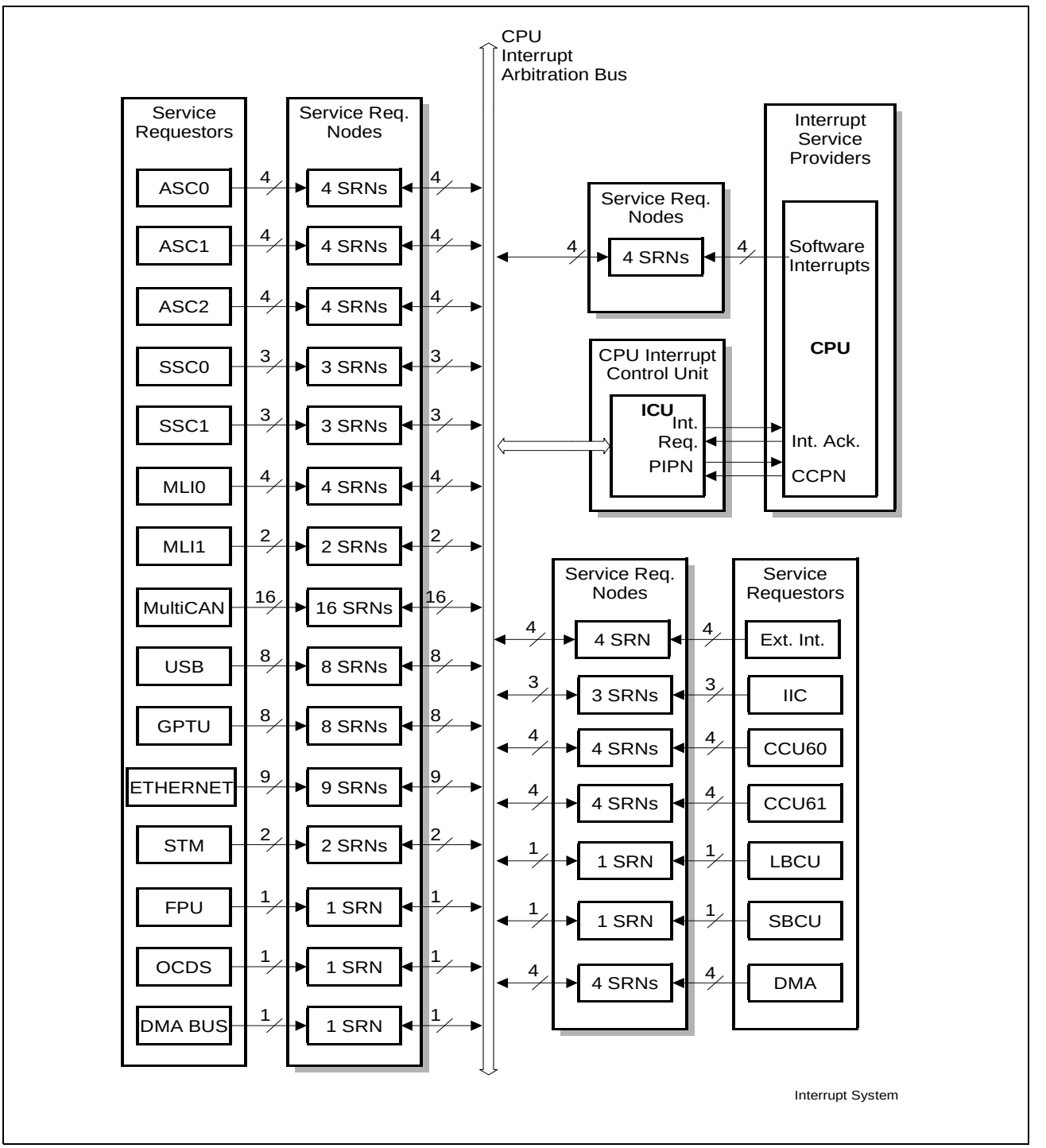


Figure 17 Block Diagram of the TC1130 Interrupt System

Boot Options

The TC1130 booting schemes provides a number of different boot options for the start of code execution. **Table 4** shows the boot options available in the TC1130.

Table 4 Boot Selections

$\overline{\text{BRKIN}}$[1)]	$\overline{\text{TM}}$[1)]	HWCFG [2:0]	Type of Boot	PC Start Value (User Entry)
1	1	000	**Bootstrap Loader**. Serial boot from ASC to PMI scratchpad, run loaded program	DFFF FFFC$_H$[2)] (D400 0000$_H$)
		001	**Bootstrap Loader**. Serial boot from CAN to PMI scratchpad, run loaded program	DFFF FFFC$_H$[2)] (D400 0000$_H$)
		010	**Bootstrap Loader**. Serial boot from SSC to PMI scratchpad, run loaded program	DFFF FFFC$_H$[2)] (D400 0000$_H$)
		011	External memory, EBU as master	DFFF FFFC$_H$[2)] (A000 0000$_H$)
		100	External memory, EBU as slave	DFFF FFFC$_H$[2)] (A000 0000$_H$)
		101	Reserved	----
		110	PMI scratchpad	D400 0000$_H$
		111	Reserved (STOP)	----
0	1	000	Tristate chip	----
		001	Go to external emulator space	DFFFFFFC$_H$[2)] (DE00 0000$_H$)
		010-111	Reserved (STOP)	----
0	0	000-111	Reserved (STOP)	----

1) This input signal is active low.

2) This is the BootROM entry address; The start address of user program in parentheses

Power Management System

The TC1130 power management system allows software to configure the various processing units so that they automatically adjust to draw the minimum necessary power for the application.

There are three power management modes:

- Run Mode
- Idle Mode
- Deep Sleep Mode

Table 5 describes these features of the power management modes.

Table 5 Power Management Mode Summary

Mode	Description
Run	The system is fully operational. All clocks and peripherals are enabled, as determined by software.
Idle	The CPU clock is disabled, waiting for a condition to return it to Run Mode. Idle Mode can be entered by software when the processor has no active tasks to perform. All peripherals remain powered and clocked. Processor memory is accessible to peripherals. A reset, Watchdog Timer event, a falling edge on the $\overline{\text{NMI}}$ pin, or any enabled interrupt event will return the system to Run Mode.
Deep Sleep	The system clock is shut off; only an external signal will restart the system. Entering this state requires an orderly shut-down controlled by the Power Management State Machine (PMSM).

Besides these explicit software-controlled power-saving modes, TC1130 supports automatic power-saving in that operating units, which are currently not required or idle, are shut off automatically until their operation is required again.

On-Chip Debug Support

The On-Chip Debug Support of the TC1130 consists of the following building blocks:

- OCDS L1 module of TriCore
- OCDS L2 interface of TriCore
- OCDS L1 module in the BCU of the FPI Bus
- OCDS L1 facilities within the DMA
- OCDS L2 interface of DMA
- OCDS System Control Unit (OSCU)
- Multi Core Break Switch (MCBS)
- JTAG based Debug Interface (Cerberus JDI)
- Suspend functionality of peripherals

Features

- TriCore L1 OCDS:
  - Hardware event generation unit
  - Break by DEBUG instruction or break signal
  - Full Single-Step support in hardware, possible also with software break
  - Access to memory, SFRs, etc. on the fly
- DMA L1 OCDS:
  - Output break request on errors
  - Suspending of pre-selected channels
- Level 2 trace port with 16 pins that outputs either TriCore, or DMA trace
- OCDS System Control Unit (Cerberus OSCU)
  - Minimum number of pins required (no OCDS enable pin)
  - Hardware allows hot attach of a debugger to a running system
  - System is secure (can be locked from internal)
- Multi Core Break Switch (Cerberus MCBS):
  - TriCore, DMA, break pins, and BCUs as break sources
  - TriCore as break targets; other parts can in addition be suspended
  - Synchronous stop and restart of the system
  - Break to Suspend converter

Figure 18 shows a basic block diagram of the building blocks.

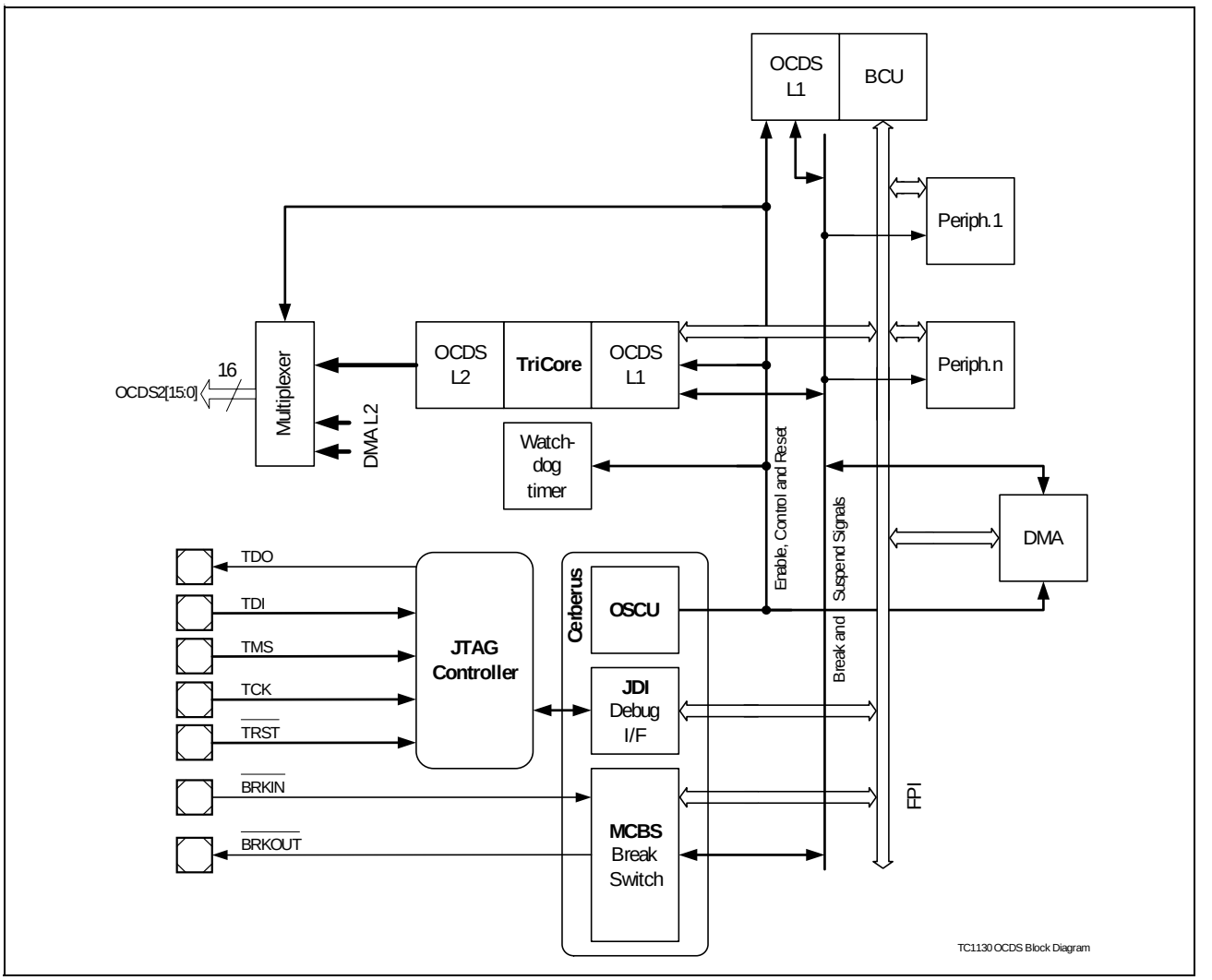


Figure 18 OCDS Support Basic Block Diagram

Clock Generation Unit

The Clock Generation Unit (CGU) allows a very flexible clock generation for TC1130. The power consumption is indirect proportional to the frequency, whereas the performance of the microcontroller is direct proportional to the frequency. During user program execution the frequency can be programmed for an optimal ratio between performance and power consumption. Therefore the power consumption can be adapted to the actual application state.

Features

The Clock Generation Unit serves different purposes:

- PLL Feature for multiplying clock source by different factors
- Direct Drive for direct clock put through
- Comfortable state machine for secure switching between basic PLL, direct or prescaler operation
- Power Down Mode support
- USB Clock source and control

The Clock Generation Unit in the TC1130, shown in **Figure 19**, consists of an oscillator circuit and one Phase-Locked Loop (PLL). The PLL can convert a low-frequency external clock signal to a high-speed internal clock for maximum performance. The PLL also has fail-safe logic that detects degenerate external clock behavior such as abnormal frequency deviations or a total loss of the external clock. It can execute emergency actions if it losses the lock on the external clock.

In general, the Clock Generation Unit (CGU) is controlled through the System Control Unit (SCU) module of the TC1130.

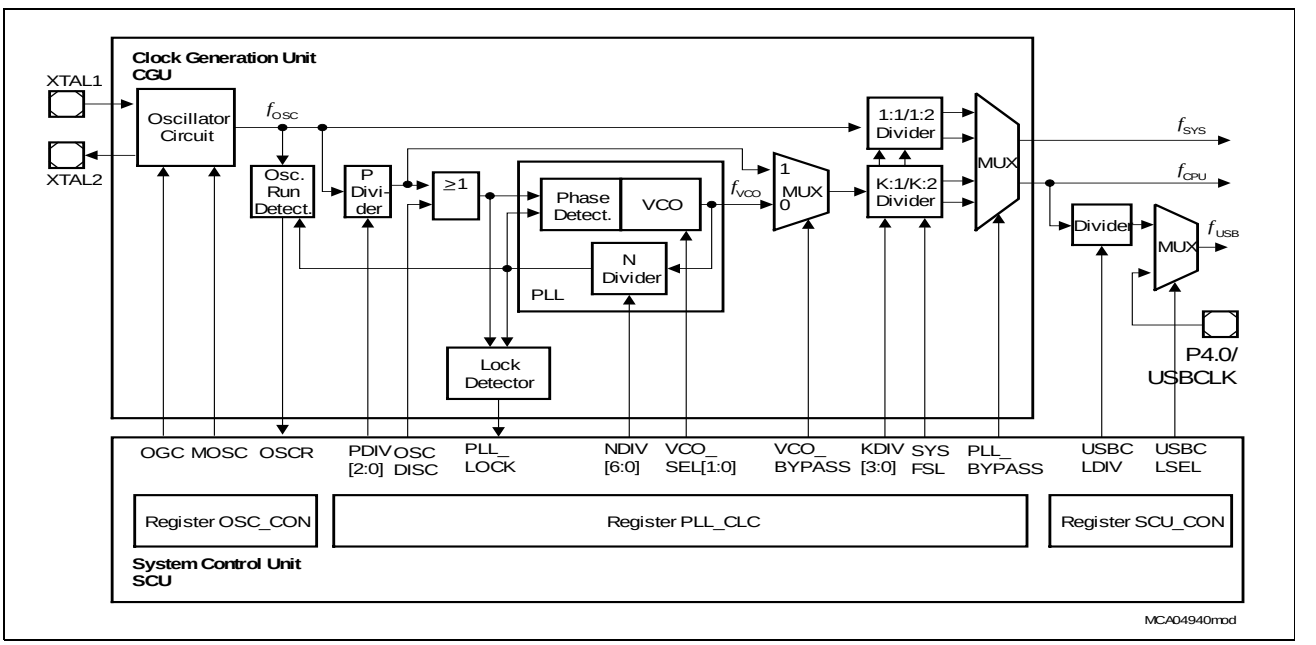


Figure 19 Clock Generation Unit Block Diagram

Recommended Oscillator Circuits

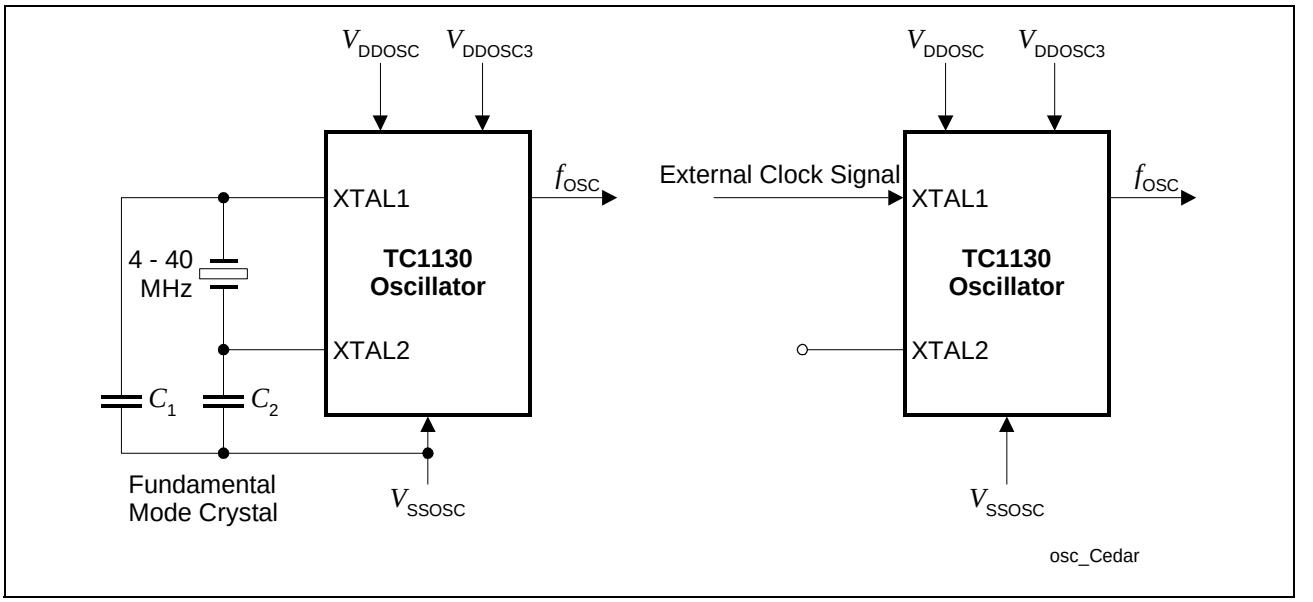


Figure 20 Oscillator Circuitries

For the main oscillator of the TC1130, the following external passive components are recommended:

- Crystal: 0~40 MHz
- C1, C2: 10 pF

A block capacitor between V_{DDOSC3} and V_{SSOSC}, V_{DDOSC} and V_{SSOSC} is recommended, too.

Power Supply

The TC1130 provides an ingenious power supply concept in order to improve the EMI behavior as well as to minimize the crosstalk within on-chip modules.

Figure 21 shows the TC1130's power supply concept, where certain logic modules are individually supplied with power. This concept improves the EMI behavior by reduction of the noise cross coupling.

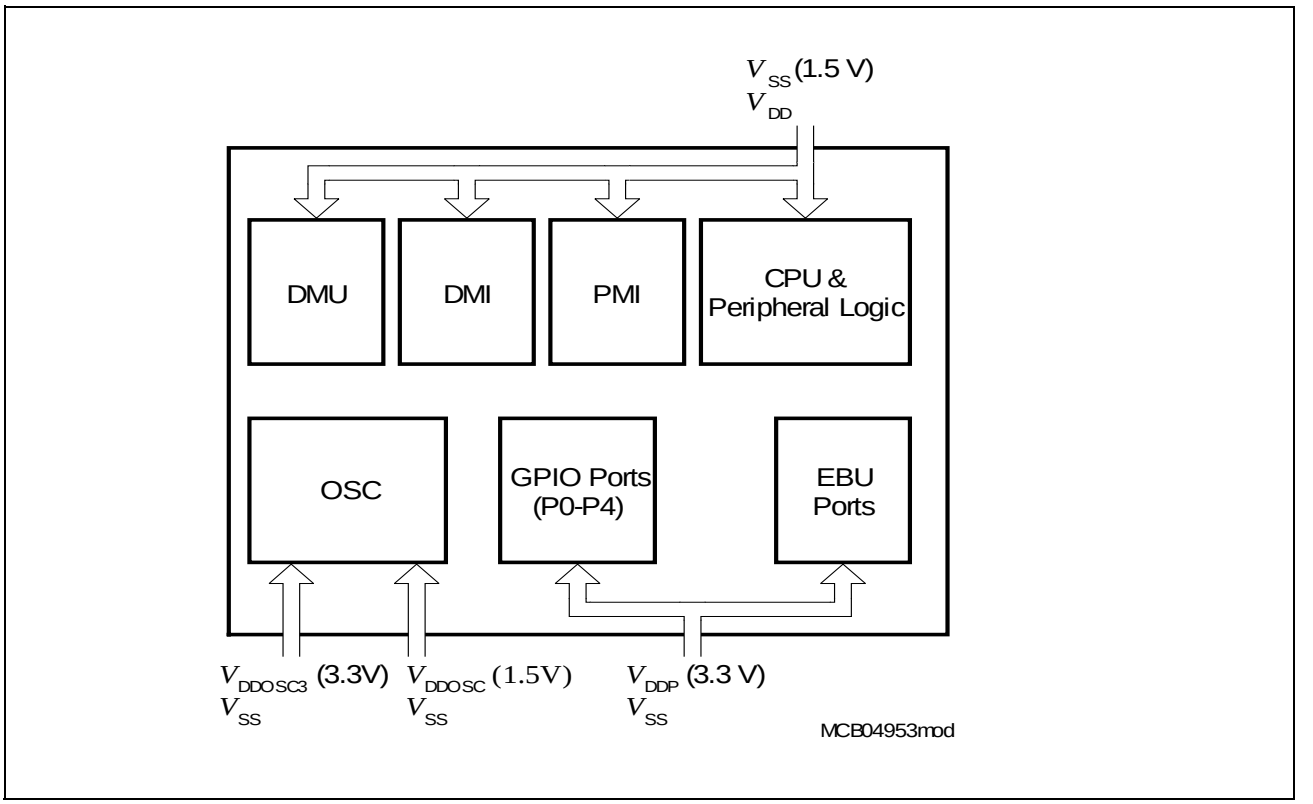


Figure 21 TC1130 Power Supply Concept

Power Sequencing

During Power-Up reset pin $\overline{\text{PORST}}$ has to be held active until both power supply voltages have reached at least their minimum values.

During the Power-Up time (rising of the supply voltages from 0 to their regular operating values) it has to be ensured, that the core V_{DD} power supply reaches its operating value first, and then the GPIO V_{DDP} power supply. During the rising time of the core voltage it must be ensured that $0 < V_{DD}-V_{DDP} < 0.5$ V.

During power-down, the core and GPIO power supplies V_{DD} and V_{DDP} respectively, have to be switched off completely until all capacitances are discharged to zero, before the next power-up.

Note: The state of the pins are undefined when only the port voltage V_{DDP} is switched on.

Identification Register Values

Table 6 TC1130 Identification Registers

Short Name	Address	Value
SCU_ID	F000 0008$_H$	002C C001$_H$
MANID	F000 0070$_H$	0000 1820$_H$
CHIPID	F000 0074$_H$	0000 8C01$_H$
RTID	F000 0078$_H$	0000 0000$_H$
SBCU_ID	F000 0108$_H$	0000 6A0A$_H$
STM_ID	F000 0208$_H$	0000 C005$_H$
JDP_ID	F000 0308$_H$	0000 6307$_H$
GPTU_ID	F000 0608$_H$	0001 C002$_H$
CCU60_ID	F000 2008$_H$	0042 C004$_H$
CCU61_ID	F000 2108$_H$	0042 C004$_H$
DMA_ID	F000 3C08$_H$	001A C011$_H$
CAN_ID	F000 4008$_H$	002B C021$_H$
USB_ID	F00E 2808$_H$	0000 4A00$_H$
SSC0_ID	F010 0108$_H$	0000 4530$_H$
SSC1_ID	F010 0208$_H$	0000 4530$_H$
ASC0_ID	F010 0308$_H$	0000 44E2$_H$
ASC1_ID	F010 0408$_H$	0000 44E2$_H$
ASC2_ID	F010 0508$_H$	0000 44E2$_H$
IIC_ID	F010 0608$_H$	0000 4604$_H$
MLI0_ID	F010 C008$_H$	0025 C004$_H$
MLI1_ID	F010 C108$_H$	0025 C004$_H$
MCHK_ID	F010 C208$_H$	001B C001$_H$
CPS_ID	F7E0 FF08$_H$	0015 C006$_H$
MMU_ID	F7E1 8008$_H$	0009 C002$_H$
CPU_ID	F7E1 FE18$_H$	000A C005$_H$
EBU_ID	F800 0008$_H$	0014 C004$_H$
DMU_ID	F800 0408$_H$	002D C001$_H$
DMI_ID	F87F FC08$_H$	0008 C004$_H$
PMI_ID	F87F FD08$_H$	000B C004$_H$

Table 6 TC1130 Identification Registers

Short Name	Address	Value
LBCU_ID	F87F FE08$_H$	000F C005$_H$
LFI_ID	F87F FF08$_H$	000C C005$_H$

Absolute Maximum Rating Targets

Parameter	Symbol	Limit Values		Unit	Notes
		min.	max.		
Ambient temperature	T_A	-40	85	°C	under bias
Storage temperature	T_{ST}	-65	150	°C	–
Junction temperature	T_J	-40	125	°C	under bias
Voltage at 1.5V power supply pins with respect to V_{SS} [1]	V_{DDC}	-0.5	1.7	V	–
Voltage at 3.3V power supply pins with respect to V_{SS} [2]	V_{DDP}	-0.5	4.0	V	–
Voltage on any pin with respect to V_{SS} [2]	V_{IN}	-0.5	4.0	V	–
Input current on any pin during overload condition	I_{IN}	-10	10	mA	[3]
Absolute sum of all input currents during overload condition	ΣI_{IN}	–	|100|	mA	[3]
CPU & LMB Bus Frequency	f_{sys}	–	150	MHz	–
FPI Bus Frequency	f_{FPI}	–	100	MHz	–
Power dissipation	P_D	–	tbd	W	–

1) Applicable for V_{DD} and V_{DDOSC}.

2) Applicable for V_{DDP} and V_{DDOSC3}. The maximum voltage difference must not exceed 4.0V in any case (i.e. Supply Voltage = 4.0V and Input Voltage = -0.5V is not allowed).

3) Restricted life time: TBD

Note: Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability. During absolute maximum rating overload conditions ($V_{IN} > V_{DD}$ or $V_{IN} < V_{SS}$) the voltage on V_{DD} pins with respect to ground (V_{SS}) must not exceed the values defined by the absolute maximum ratings.

Operating Condition

The following operating conditions must not be exceeded in order to ensure correct operation of the TC1130. All parameters specified in the following table refer to these operating conditions, unless otherwise noticed.

Parameter	Symbol	Limit Values		Unit	Notes Conditions
		min.	max.		
Digital supply voltage	V_{DDC}	1.43	1.58	V	
	V_{DDP}	3.14	3.47	V	
Digital ground voltage	V_{SS}	0		V	–
Digital core supply current	I_{DD}		525	mA	
Ambient temperature under bias	T_A	-40	+85	°C	–
CPU clock	f_{SYS}	–[1]	150	MHz	–
Overload current	I_{OV}	-1	1	mA	2)3)
		-3	3		duty cycle ≤ 25%
Short circuit current	I_{SC}	-1	1	mA	4)
		-3	3		duty cycle ≤ 25%
Absolute sum of overload + short circuit currents	$\Sigma\|I_{OV}\| + \|I_{SC}\|$	–	|50|	mA	3)
			|100|		duty cycle ≤ 25%
Inactive device pin current (V_{DD} = V_{DDP} = 0)	I_{ID}	-1	1	mA	–
External load capacitance	C_L	–	50	pF	
ESD strength		2000	–	V	Human Body Model (HBM)

1) The TC1130 uses a static design, so the minimum operation frequency is 0 MHz. Due to test time restriction no lower frequency boundary is tested, however.

2) Overload conditions occur if the standard operating conditions are exceeded, i.e. the voltage on any pin exceeds the specified range (i.e. $V_{OV} > V_{DDP}$ + 0.5 V or $V_{OV} < V_{SS}$ - 0.5 V). The absolute sum of input overload currents on all digital IO pins may not exceed **50 mA**. The supply voltage must remain within the specified limits.

3) Not 100% tested, guaranteed by design and characterization.

4) Applicable for digital inputs.

Parameter Interpretation

The parameters listed on the following pages partly represent the characteristics of the TC1130 and partly its demands on the system. To aid in interpreting the parameters right, when evaluating them for a design, they are marked in column “Symbol”:

CC (**C**ontroller **C**haracteristics):

The logic of the TC1130 will provide signals with the respective timing characteristics.

SR (**S**ystem **R**equirement):

The external system must provide signals with the respective timing characteristics to the TC1130.

DC Characteristics

DC-Characteristics

V_{SS} = 0 V; T_A = -40°C to +125°C

Parameter	Symbol		Limit Values		Unit	Test Condition
			min.	max.		
GPIO pins, Dedicated pins and EBU pins						
Input low voltage	V_{IL}	SR	-0.3	0.8	V	LvTTL
Input high voltage	V_{IH}	SR	2.0	V_{DDP} + 0.3	V	LvTTL
Output low voltage	V_{OL}	CC	–	0.4	V	I_{OL} = 2mA
Output high voltage	V_{OH}	CC	2.4	–	V	I_{OH} = -2mA
Pull-up current [1]	$\|I_{PUA}\|$	CC	–	149	µA	V_{IN} = 0V
	$\|I_{PUC}\|$	CC	–	7.2	µA	V_{IN} = 0V
Pull-down current [2]	$\|I_{PDA}\|$	CC	–	156	µA	V_{IN} = V_{DDP}
	$\|I_{PDC}\|$	CC	–	15.7	µA	V_{IN} = V_{DDP}
Input leakage current [3]	I_{OZ1}	CC	–	±350	nA	0 < V_{IN} < V_{DDP}
Pin Capacitance [4]	C_{IO}	CC	–	10	pF	f = 1 MHz T_A = 25 °C
Oscillator Pins						
Input low voltage at XTAL1	V_{ILX}	SR	0	0.1	V	
Input high voltage at XTAL1	V_{IHX}	SR	1.4	1.5	V	

Notes:

1) The current is applicable to the pins, for which a pull up has been specified. Refer to **Table 1**. I_{PUx} refers to the pull up current for type *x in absolute values*.

2) The current is applicable to the pins, for which a pull down has been specified. Refer to **Table 1**. I_{PDx} refers to the pull down current for type *x in absolute values*.

3) Excluded following pins : $\overline{\text{NMI}}$, $\overline{\text{TRST}}$, TCK, TDI, TMS, MII_TXCLK, MII_RXCLK, MII_MDIO, ALE, P2.1,HWCFG0, HWCFG1, HWCFG2, BRKIN, PORST, HDRST.

4) Not 100% tested, guaranteed by design characterization.

USB Interface

Table 7 DC Electrical Characteristics

Parameter	Symbol	Limit Values			Unit	Test Conditions
		min.	typ.	max.		
Supply Voltage						
Supply Voltage Extern	V_{DDP}	3.14	3.3	3.47	V	
Supply Voltage Intern	V_{DDE}	1.4	1.5	1.6	V	
Input Level						
Differential Input Level		0.2			V	|V(D+) - V(D-)|
Differential Common Mode Range		0.8		2.5	V	Range of Sensitivity
Single Ended Receiver Threshold		*low* < 0.8		*high* > 2.0	V	
Output Levels						
Static Output Low				< 0.3	V	with 1.5 kΩ to 3.6 V
Static Output High		2.8	3.3	3.6	V	with 15 kΩ to ground
Leakage Current						
Hi_Z State Data Line Leakage		-10		10	µA	$0 < V_{in} < 3.3V$

Table 8 Full Speed Electrical Characteristic

Parameter	Symbol	Limit Values			Unit	Test Conditions
		min.	typ.	max.		
Driver Characteristics						
Rise / Fall Time		4		20	ns	Capacitive load 50 pF
Rise / Fall Time Matching		90	100	110	%	Capacitive load 50 pF
Crossover Voltage of differential Signals		1.3		2.0	V	Capacitive load 50 pF
Driver Output Impedance		28		44	Ω	Steady State Driver
Termination Impedance		1.425	1.5	1.575	kΩ	

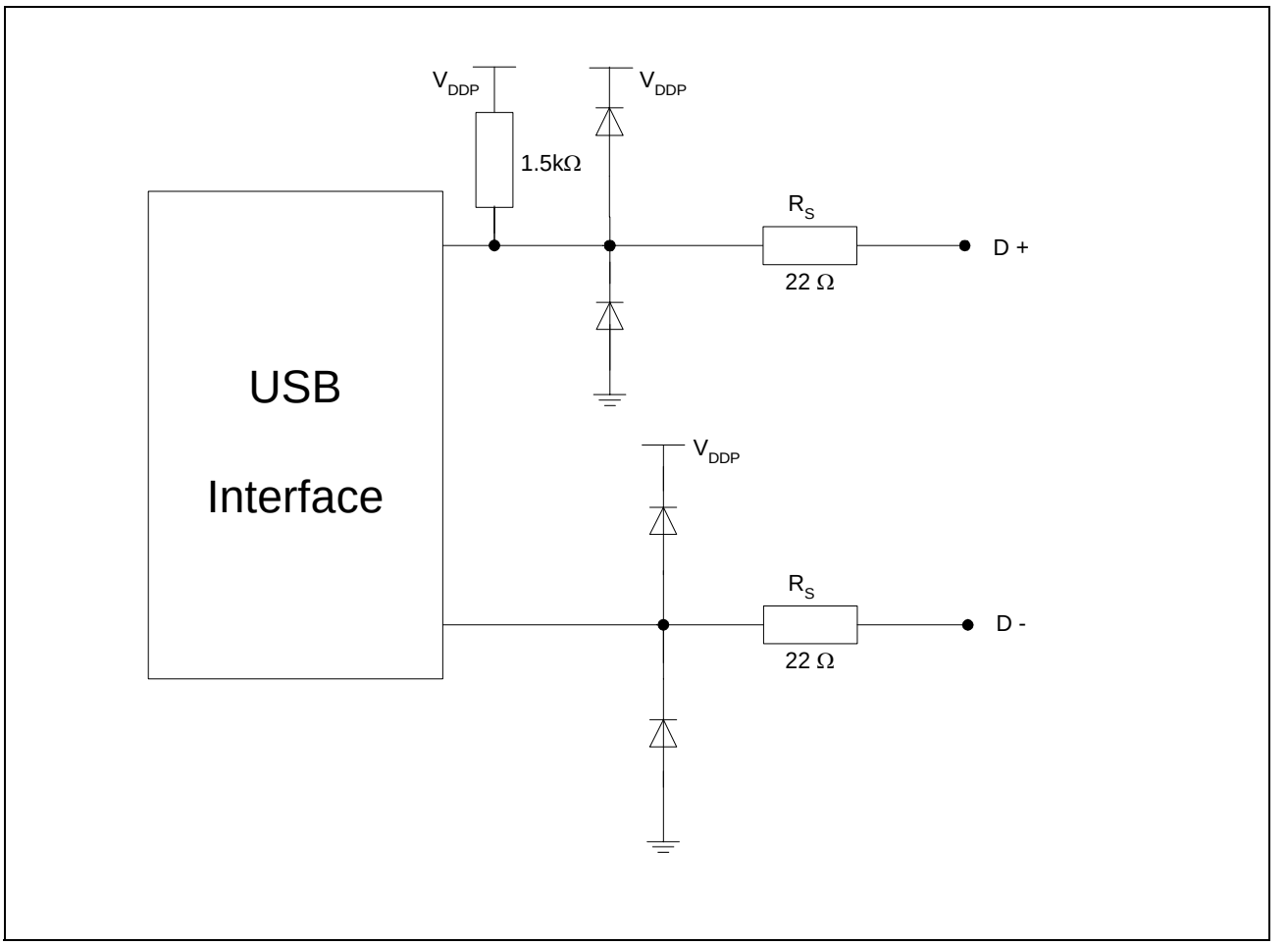


Figure 22 USB Interface

IIC Pins

Each IIC Pin is an open drain output pin with different characteristics than other pins. The related characteristics are given in the following table

Parameter	Symbol	Limit values		Unit	Test Conditions
		min.	max.		
Output low voltage	V_{OL} CC	-	0.4 0.6	V	3 mA sink current 6 mA sink current
Input high voltage[1)]	V_{IH} SR	$0.7V_{DDP}$	$V_{DDP}+0.5$	V	-
Input low voltage[1)]	V_{IL} SR	-0.5	$0.3V_{DDP}$	V	-

1) Guaranteed by design characterization

Note: No 5 V IIC interface is supported with these pads. Only voltages lower than 3.63 V must be applied to these pads.

Note: IIC pins have no Pull-Up and Pull-Down devices.

Power Supply Current

Parameter	Symbol	Limit values		Unit	Test Conditions
		typ. [1]	max.		
Active mode supply current	I_{DD}	314	679	mA	Sum of I_{DDS} [2]
		153	345	mA	I_{DD} at V_{DD} [3]
		156	322	mA	I_{DD} at V_{DDP}
Idle mode supply current	I_{ID}	74	154	mA	Sum of I_{DDS} [2][4]
		66	130	mA	I_{DD} at V_{DD} [3][4]
		6	15	mA	I_{DD} at V_{DDP} [4]
Deep sleep mode supply current	I_{DS}	2	19	mA	Sum of I_{DDS} [2][5]
		2	19	mA	I_{DD} at V_{DD} [3][5]
		3.6	58	µA	I_{DD} at V_{DDP} [5]

1) Typical values are measured at 25°C, CPU clock at xxx MHz and nominal supply voltage, i.e. 3.3V for V_{DDP}, V_{DDOSC3} and 1.5V for V_{DD}, V_{DDOSC}. These currents are measured using a typical application pattern. The power consumption of modules can increase or decrease using other application programs.

2) These power supply currents are defined as the sum of all currents at the V_{DD} power supply lines: V_{DD} + V_{DDP} + V_{DDOSC3} + V_{DDOSC}.

3) This measurement includes the TriCore and Logic power supply lines.

4) CPU is in idle state, input clock to all peripherals are enabled,

5) Clock generation is disabled at the source.

AC Characteristics

*Note: The values in **Blue** color are gotten from STA.*

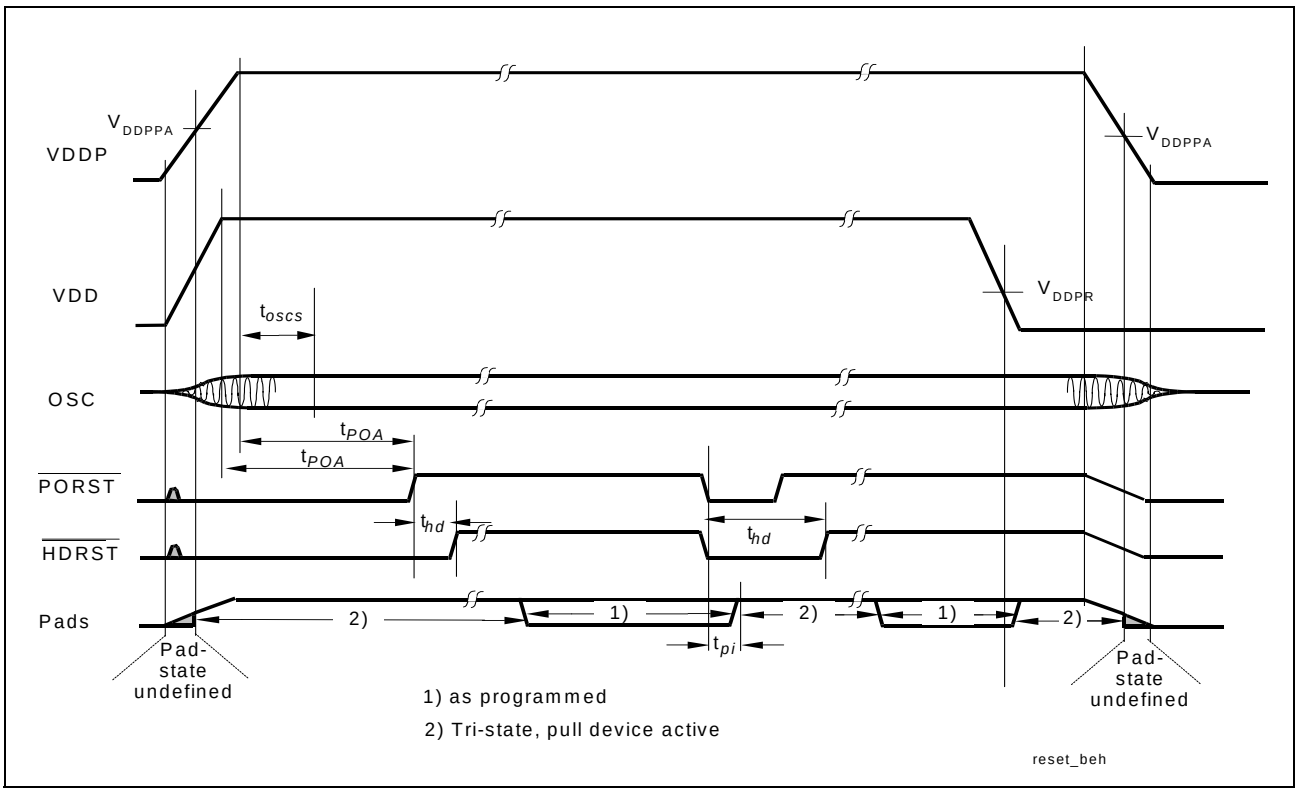
Power, Pad and Reset Timing

Parameter	Symbol		Limit Values		Unit
			min.	max.	
Min. VDDP voltage to ensure defined pad states	V_{DDPPA}	CC	xxxx	–	V
Oscillator start-up time[1)]	t_{OSCS}	CC	–	30	ms
Minimum $\overline{\text{PORST}}$ active time after power supplies are stable at operating levels	t_{POA}	CC	50	–	ms
$\overline{\text{HRST}}$ pulse width	t_{HD}	CC	1024 cycles[2)]		f_{SYS}
Ports inactive after any reset active[3)]	t_{PI}	CC	–	30	ns

1) Not measured, guaranteed by device characterization

2) Any $\overline{\text{HDRST}}$ activation is internally prolonged to 1024 FPI bus clock cycles

3) Not measured, guaranteed by design characterization



PLL Parameters

Phase Locked Loop (PLL)

When PLL operation is configured (*PLL_CLC.LOCK = 1*) the on-chip phase locked loop is enabled and provides the master clock. The PLL multiplies the input frequency by the factor **F** ($f_{MC} = f_{OSC} \times$ **F**) which results from the input divider, the multiplication factor (N Factor), and the output divider (**F** = *NDIV+1 / (PDIV+1 × KDIV+1)*). The PLL circuit synchronizes the master clock to the input clock. This synchronization is done smoothly, i.e. the master clock frequency does not change abruptly.

Due to this adaptation to the input clock the frequency of f_{MC} is constantly adjusted so it is locked to f_{OSC}. The slight variation causes a jitter of f_{MC} which also affects the duration of individual TCMs.

The timing listed in the AC Characteristics refers to TCPs. Because f_{CPU} is derived from f_{MC}, the timing must be calculated using the minimum TCP possible under the respective circumstances.

The actual minimum value for TCP depends on the jitter of the PLL. As the PLL is constantly adjusting its output frequency so it corresponds to the applied input frequency (crystal or oscillator) the relative deviation for periods of more than one TCP is lower than for one single TCP (see formula and **Figure 23**).

This is especially important for bus cycles using waitstates and e.g. for the operation of timers, serial interfaces, etc. For all slower operations and longer periods (e.g. pulse train generation or measurement, lower baudrates, etc.) the deviation caused by the PLL jitter is negligible.

The value of the accumulated PLL jitter depends on the number of consecutive VCO output cycles within the respective timeframe. The VCO output clock is divided by the output prescaler (K = *KDIV*+1) to generate the master clock signal f_{MC}. Therefore, the number of VCO cycles can be represented as K × N, where N is the number of consecutive f_{MC} cycles (TCM).

For a period of N × TCM the accumulated PLL jitter is defined by the corresponding deviation D_N:

D_N [ns] = ±(1.5 + 6.32 × N / f_{MC}); f_{MC} in [MHz], N = number of consecutive TCMs.

So, for a period of 3 TCMs @ 20 MHz and K = 12: D_3 = ±(1.5 + 6.32 × 3 / 20) = 2.448 ns.

This formula is applicable for K × N < 95. For longer periods the K×N=95 value can be used. This steady value can be approximated by: D_{Nmax} [ns] = ±(1.5 + 600 / (K × f_{MC})).

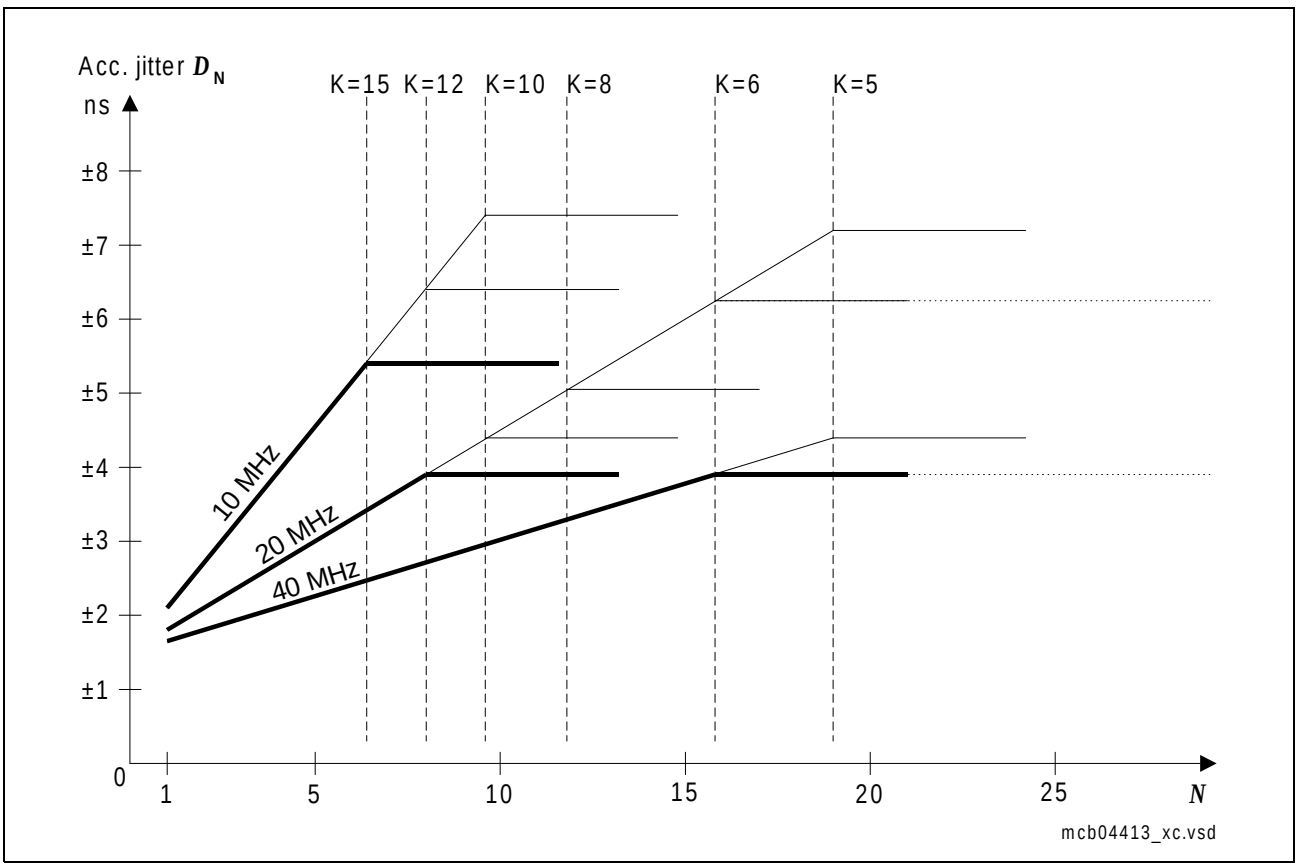


Figure 23 Approximated Accumulated PLL Jitter

Note: The bold lines indicate the minimum accumulated jitter which can be achieved by selecting the maximum possible output prescaler factor K.

Different frequency bands can be selected for the VCO, so the operation of the PLL can be adjusted to a wide range of input and output frequencies:

Table 9 VCO Bands for PLL Operation

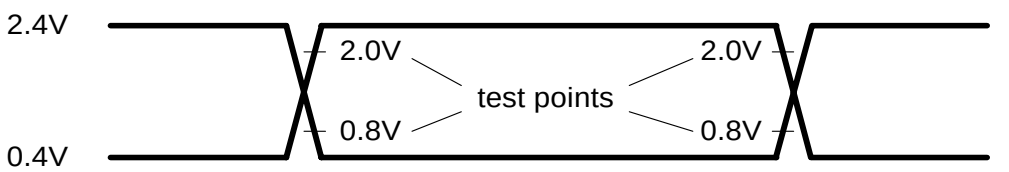
PLL_CLC.VCOSEL	VCO Frequency Range	Base Frequency Range [1]
00	400 ... 500 MHz	250 ... 320 MHz
01	500 ... 600 MHz	300 ... 400 MHz
10	600 ... 700 MHz	350 ... 480 MHz
11	Reserved [2]	

1) *Base Frequency Range is the free running operation frequency of the PLL, when no input clock is available.*

2) *This option can not be used.*

AC Characteristics

(Operating Conditions apply)



AC inputs during testing are driven at 2.4V for a logic “1” and 0.4V for a logic “0”. Timing measurements are made at V_{IHmin} for a logic “1” and V_{ILmax} for a logic “0”.

Figure 24 Input/Output Waveforms for AC Tests - for GPIO, Dedicated and EBU pins

Input Clock Timing
(Operating Conditions apply)

Parameter		Symbol	Limits		Unit
			min	max	
Oscillator clock frequency	with PLL	f_{OSC} SR	4	40	MHz
Input clock frequency driving at XTAL1	with PLL	f_{OSCDD} SR	-	40	MHz
Input Clock Duty Cycle (t_1 /t_2)		SR	45	55	%

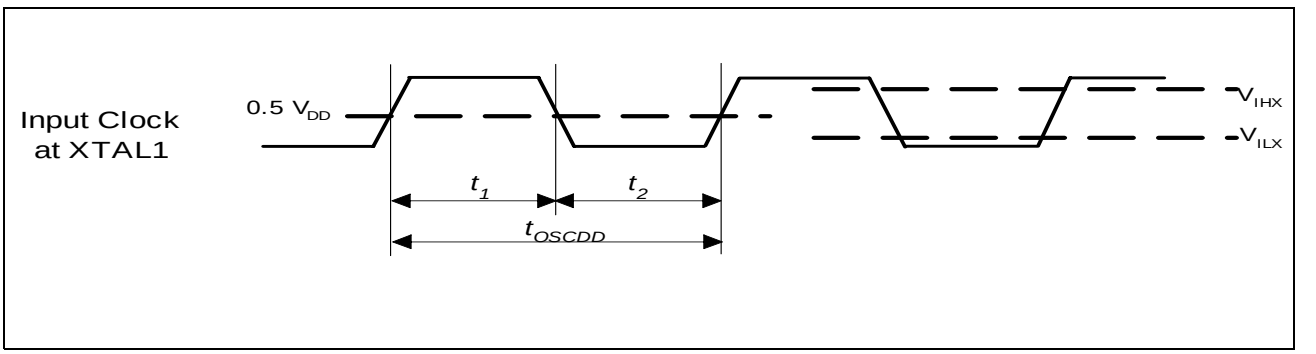


Figure 25 Input Clock Timing

Port Timing

(Operating Conditions apply; C_L = 50 pF)

Parameter	Symbol	Limits		Unit
		min	max	
Port data valid from TRCLK ↓ [1]	t_1 CC	–	13	ns

[1] Port data is output with respect to the FPI clock. The TRCLK is used as a reference here since the FPI clock is not available as an external pin and TRCLK is same frequency as CPU clock. Port lines maintain its state for at least 2 CPU clocks.

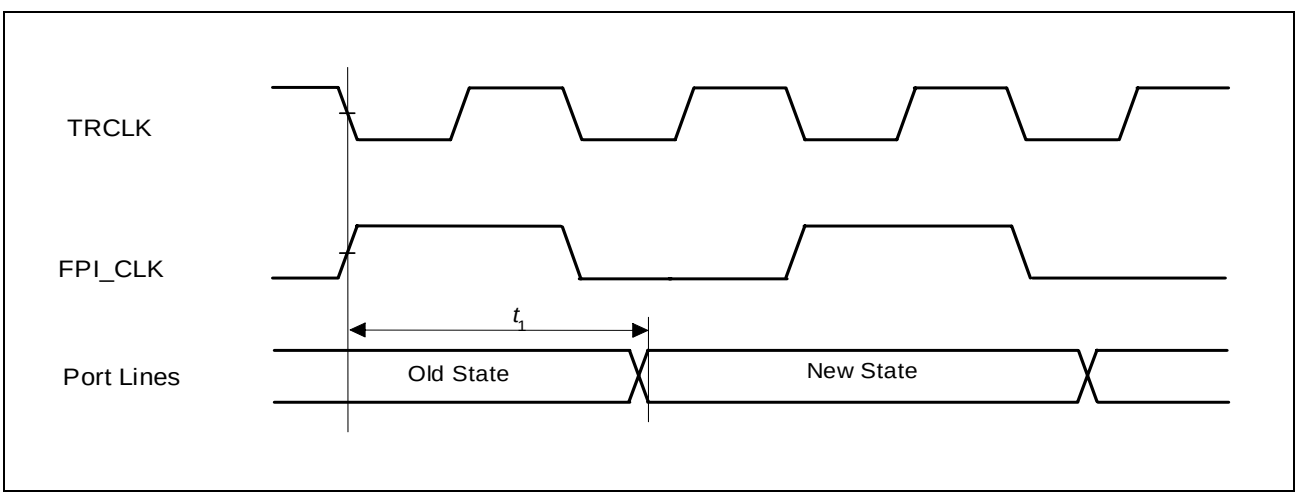


Figure 26 Port Timing

Timing for EBU_LMB Clock Outputs

SDCLKO Output Clock Timing

(Operating Conditions apply; CL = 50 pF)

Parameter	Symbol		Limits			Unit
			min		max	
SDCLKO period	t_1	CC	10	–	–	ns
SDCLKO high time	t_2	CC	4.5	–	–	ns
SDCLKO low time	t_3	CC	3	–	–	ns
SDCLKO rise time	t_4	CC	–	–	2.5	ns
SDCLKO fall time	t_5	CC	–	–	2.5	ns
SDCLKO duty cycle $t_2/(t_2 + t_3)$	DC	CC	45	50	55	%

BFCLKO Output Clock Timing

(Operating Conditions apply; C_L = 50 pF)

Parameter	Symbol		Limit Values			Unit
			min.	typ.	max.	
Clock period	t_1	CC	20	–	–	ns
BFCLKO high time	t_2	CC	9	–	–	ns
BFCLKO low time	t_3	CC	9	–	–	ns
BFCLKO rise time	t_4	CC	–	–	3.5	ns
BFCLKO fall time	t_5	CC	–	–	2.5	ns
BFCLKO duty cycle $t_2/(t_2 + t_3)$[1)]	DC	CC	45	50	55	%

1) This duty cycle is not applicable when BFCON.extclock equals to 10 (1/3 of LMBCLK frequency)

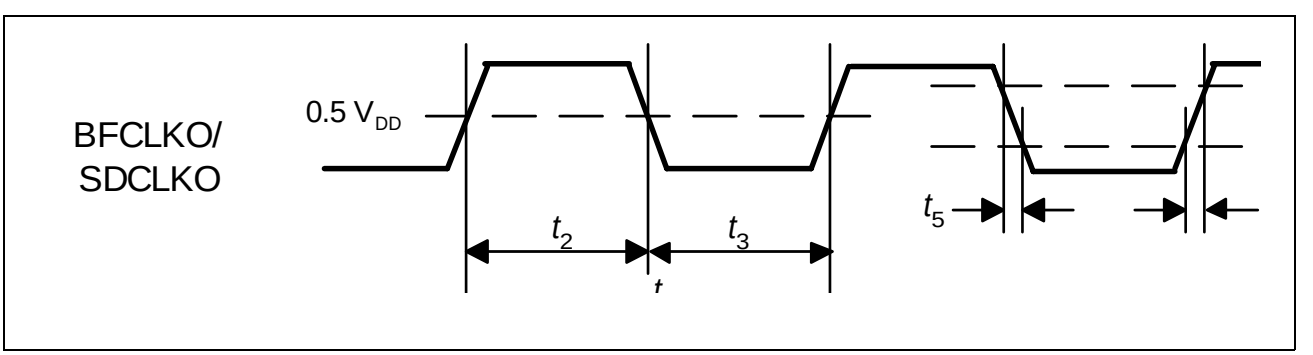


Figure 27 EBU Clock Output Timing

Timing for SDRAM Access Signals

(Operating Conditions apply; C_L = 50 pF[1])

Parameter	Symbol		Limits		Unit
			min	max	
CKE output valid time from SDCLKO	t_1	CC	–	8.0	ns
CKE output hold time from SDCLKO	t_2	CC	1.0	–	ns
Address output valid time from SDCLKO	t_3	CC	–	8.0	ns
Address output hold time from SDCLKO	t_4	CC	1.0	–	ns
$\overline{CSx}$, $\overline{RAS}$, $\overline{CAS}$, RD/$\overline{WR}$, $\overline{BC(3:0)}$ output valid time from SDCLKO	t_5	CC	–	8.0	ns
$\overline{CSx}$, $\overline{RAS}$, $\overline{CAS}$, RD/$\overline{WR}$, $\overline{BC(3:0)}$ output hold time from SDCLKO	t_6	CC	1.0	–	ns
AD(31:0) output valid time from SDCLKO	t_7	CC	–	8.0	ns
AD(31:0) output hold time from SDCLKO	t_8	CC	1.0	–	ns
AD(31:0) input setup time to SDCLKO	t_9	SR	4.0	–	ns
AD(31:0) input hold time from SDCLKO	t_{10}	SR	3.0	–	ns

1) If application conditions other than 50 pf capacitive load are used, then the proper correlation factor should be used for your specific application condition. For design team, the load should be set according to the system requirement.

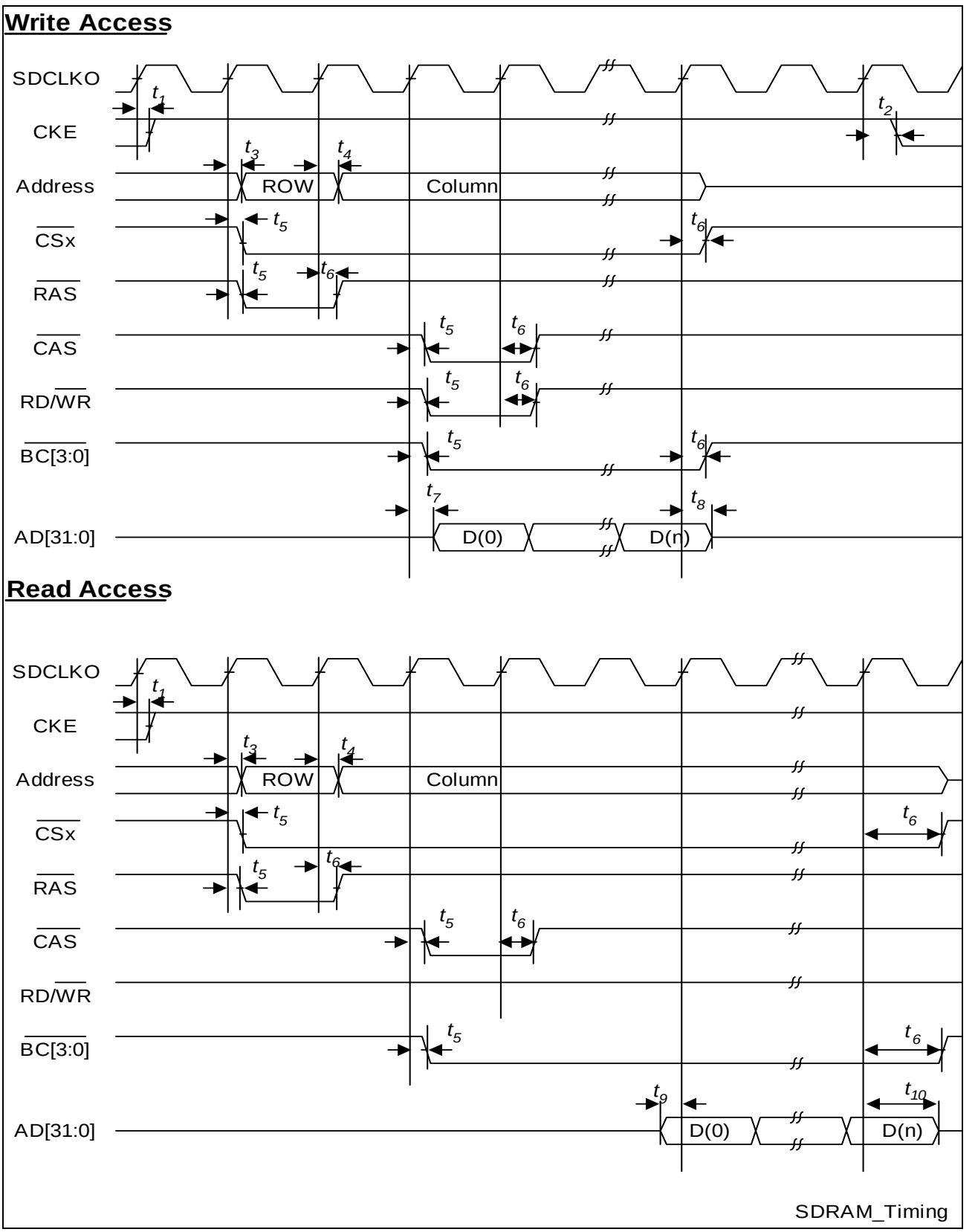


Figure 28 SDRAM Access Timing

Timing for Burst Flash Access Signals
Operating Conditions apply; C_L = 50 pF)

Parameter	Symbol		Limits		Unit
			min	max	
Address output valid time from BFCLKO	t_1	CC	–	11.0	ns
Address output hold time from BFCLKO	t_2	CC	10.0	–	ns
$\overline{\text{CSx}}$ output valid time from BFCLKO	t_3	CC	–	9.0	ns
$\overline{\text{RD}}$ output valid time from BFCLKO	t_4	CC	–	10.0	ns
$\overline{\text{ADV}}$ output valid time from BFCLKO	t_5	CC	–	10.0 (7.0)	ns
$\overline{\text{ADV}}$ output hold time from BFCLKO	t_6	CC	3.0 (0.0)	–	ns
$\overline{\text{BAA}}$ output valid time from BFCLKO	t_7	CC	–	10.0 (7.0)	ns
$\overline{\text{BAA}}$ output hold time from BFCLKO	t_8	CC	3.0	–	ns
AD(31:0) input setup time to BFCLKO	t_9	SR	5.0	–	ns
AD(31:0) input hold time from BFCLKO	t_{10}	SR	3.0	–	ns
$\overline{\text{WAIT}}$ input setup time to BFCLKO	t_{11}	SR	5.0	–	ns
$\overline{\text{WAIT}}$ input hold time from BFCLKO	t_{12}	SR	3.0	–	ns

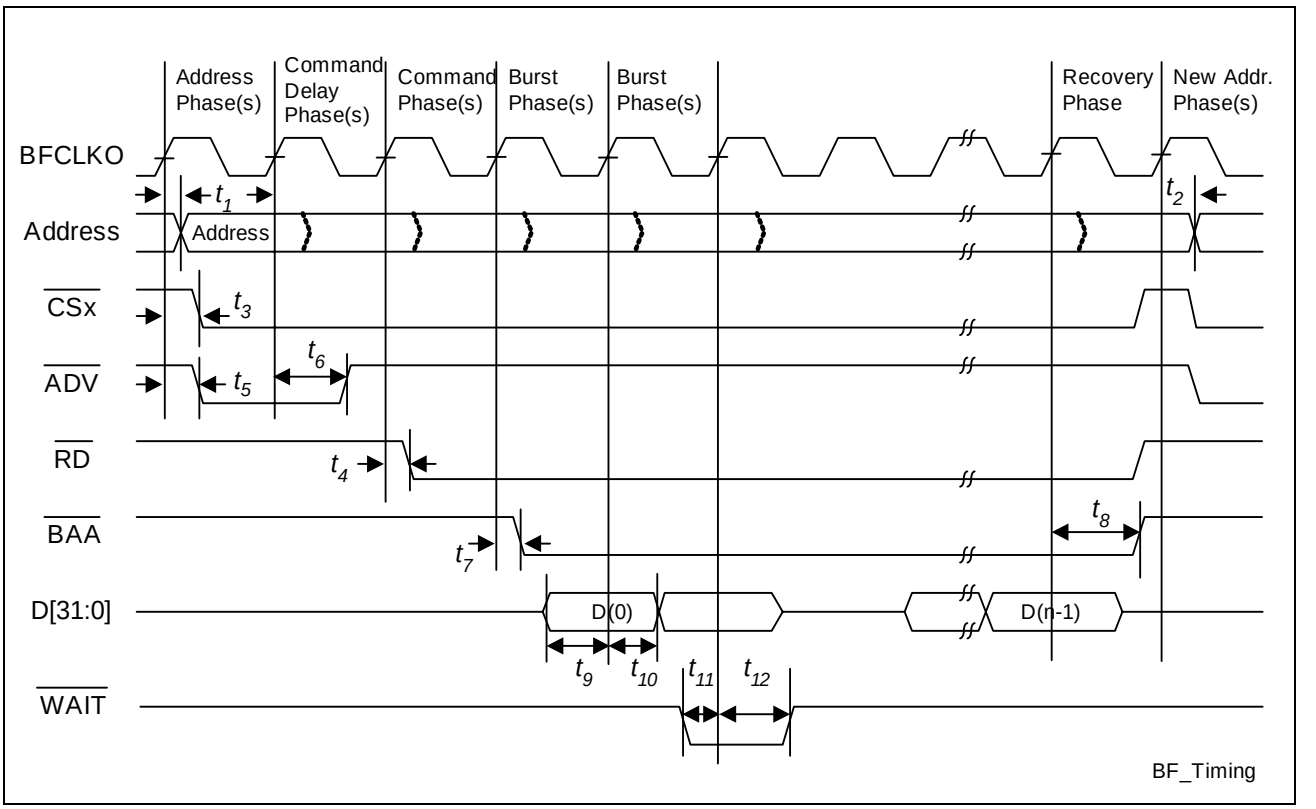


Figure 29 Burst Flash Access Timing

Note: Output delays are always referenced to BFCLKO. The reference clock for input characteristics depends on bit BFCON.FDBKEN.
BFCON.FDBKEN = 0: BFCLKO is the input reference clock.
BFCON.FDBKEN = 1: BFCLKI is the input reference clock (EBULMB clock feedback enabled)

Timing for Demultiplexed Access Signals

(Operating Conditions apply; C_L = 50 pF) [1)]

Parameter	Symbol		Limits		Unit
			min	max	
ALE, $\overline{CSx}$, RD/$\overline{WR}$, $\overline{RD}$, MR/$\overline{W}$, $\overline{BC(3:0)}$ output valid time from output clock ↓	t_1	CC	–	3.2	ns
ALE, $\overline{CSx}$, RD/$\overline{WR}$, $\overline{RD}$, MR/$\overline{W}$, $\overline{BC(3:0)}$ output hold time from output clock ↓	t_2	CC	0.0	–	ns
Address output valid time from output clock ↓	t_3	CC	–	3.5	ns
Address output hold time from output clock ↓	t_4	CC	0.0	–	ns
$\overline{WAIT}$ input setup time to output clock ↓	t_7	SR	10.6	–	ns
$\overline{WAIT}$ input hold time from output clock ↓	t_8	SR	0.0	–	ns
AD(31:0) output valid time from output clock ↓	t_9	CC	–	2.6	ns
AD(31:0) output hold time from output clock ↓	t_{10}	CC	0.0	–	ns
AD(31:0) input setup time to output clock ↓	t_{11}	SR	1.3	–	ns
AD(31:0) input hold time from output clock ↓	t_{12}	SR	0.9	–	ns
$\overline{RMW}$ output valid time from output clock ↓	t_{13}	CC	–	6.3	ns
$\overline{RMW}$ output hold time from output clock ↓	t_{14}	CC	1.3	–	ns
$\overline{ADV}$ width	t_{15}	CC	10	–	ns
AD(31:0) output hold time from RD/$\overline{WR}$ ↓	t_{16}	CC	0	–	ns

1) The purpose for characterization of Asynchronous access is to provide the performance of all of the signals to user. User can decide whether an extra cycle is needed or not based on above parameters to generate signals with correct timing sequence. It is user's responsibility to program the correct phase length according to the memory/peripheral device specification and EBU specification.

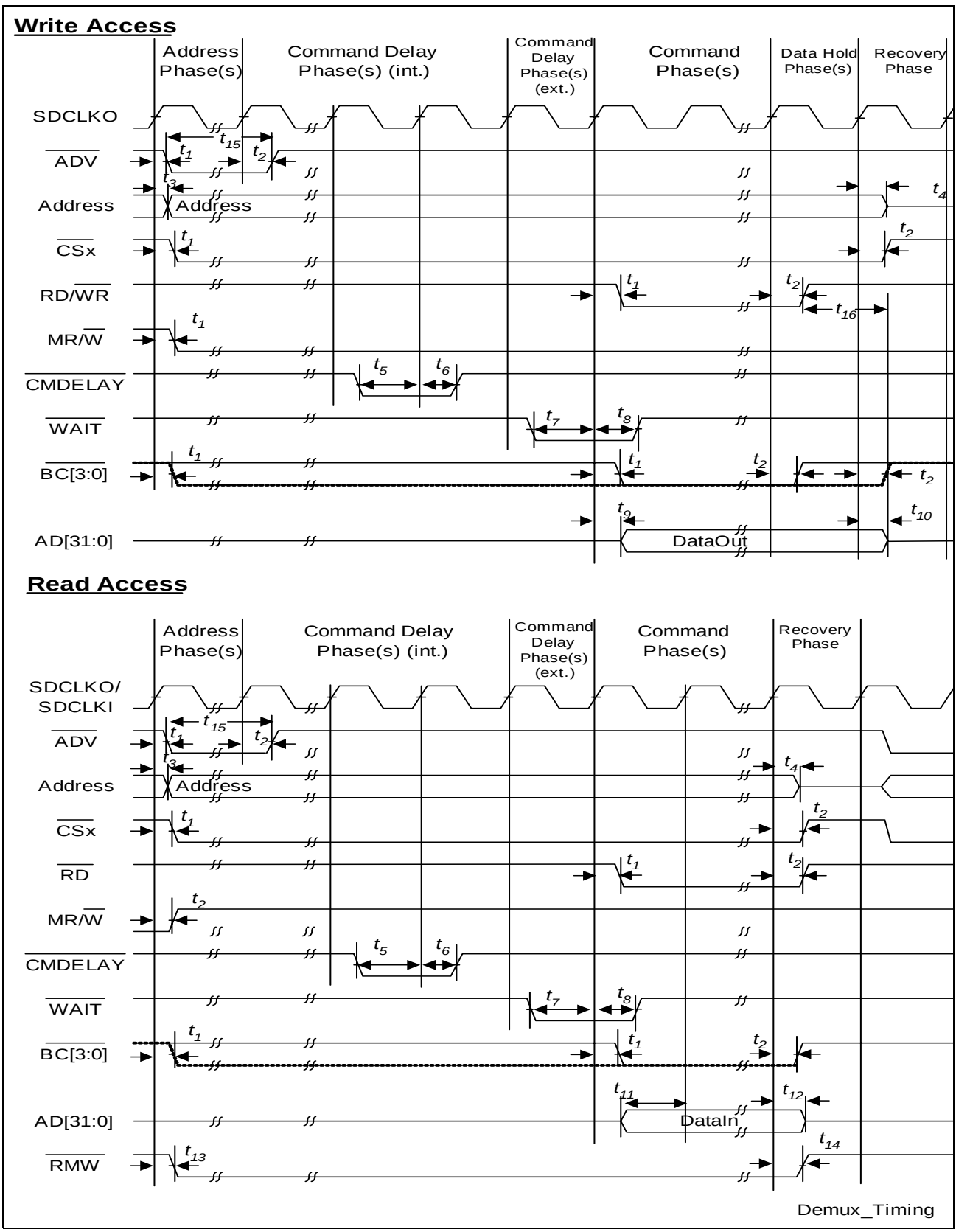


Figure 30 Demultiplexed Asynchronous Device Access Timing

Timing for Multiplexed Access Signals

(Operating Conditions apply; C_L = 50 pF)[1)]

Parameter	Symbol		Limits		Unit
			min	max	
ALE, $\overline{\text{CSx}}$, RD/$\overline{\text{WR}}$, $\overline{\text{RD}}$, MR/$\overline{\text{W}}$, $\overline{\text{BC(3:0)}}$ output valid time from output clock ↓	t_1	CC	–	3.2	ns
ALE, $\overline{\text{CSx}}$, RD/$\overline{\text{WR}}$, $\overline{\text{RD}}$, MR/$\overline{\text{W}}$, $\overline{\text{BC(3:0)}}$ output hold time from output clock ↓	t_2	CC	0.0	–	ns
AD(31:0) output valid time from output clock ↓	t_3	CC	–	2.6	ns
AD(31:0) output hold time from output clock ↓	t_4	CC	0.0	–	ns
AD(31:0) input setup time to output clock ↓	t_5	SR	1.4	–	ns
AD(31:0) input hold time from output clock ↓	t_6	SR	0.8	–	ns
$\overline{\text{WAIT}}$ input setup time to output clock ↓	t_9	SR	10.6	–	ns
$\overline{\text{WAIT}}$ input hold time from output clock ↓	t_{10}	SR	0.0	–	ns
$\overline{\text{RMW}}$ output valid time from output clock ↓	t_{11}	CC	–	6.3	ns
$\overline{\text{RMW}}$ output hold time from output clock ↓	t_{12}	CC	1.3	–	ns
$\overline{\text{ADV}}$ width	t_{13}	CC	10.0	–	ns
AD(31:0) output hold time from RD/$\overline{\text{WR}}$ ↓	t_{14}	CC	0	–	ns

1) The purpose for characterization of Asynchronous access is to provide the performance of all of the signals to user. User can decide whether an extra cycle is needed or not based on above parameters to generate signals with correct timing sequence. It is user's responsibility to program the correct phase length according to the memory/peripheral device specification and EBU Specification.

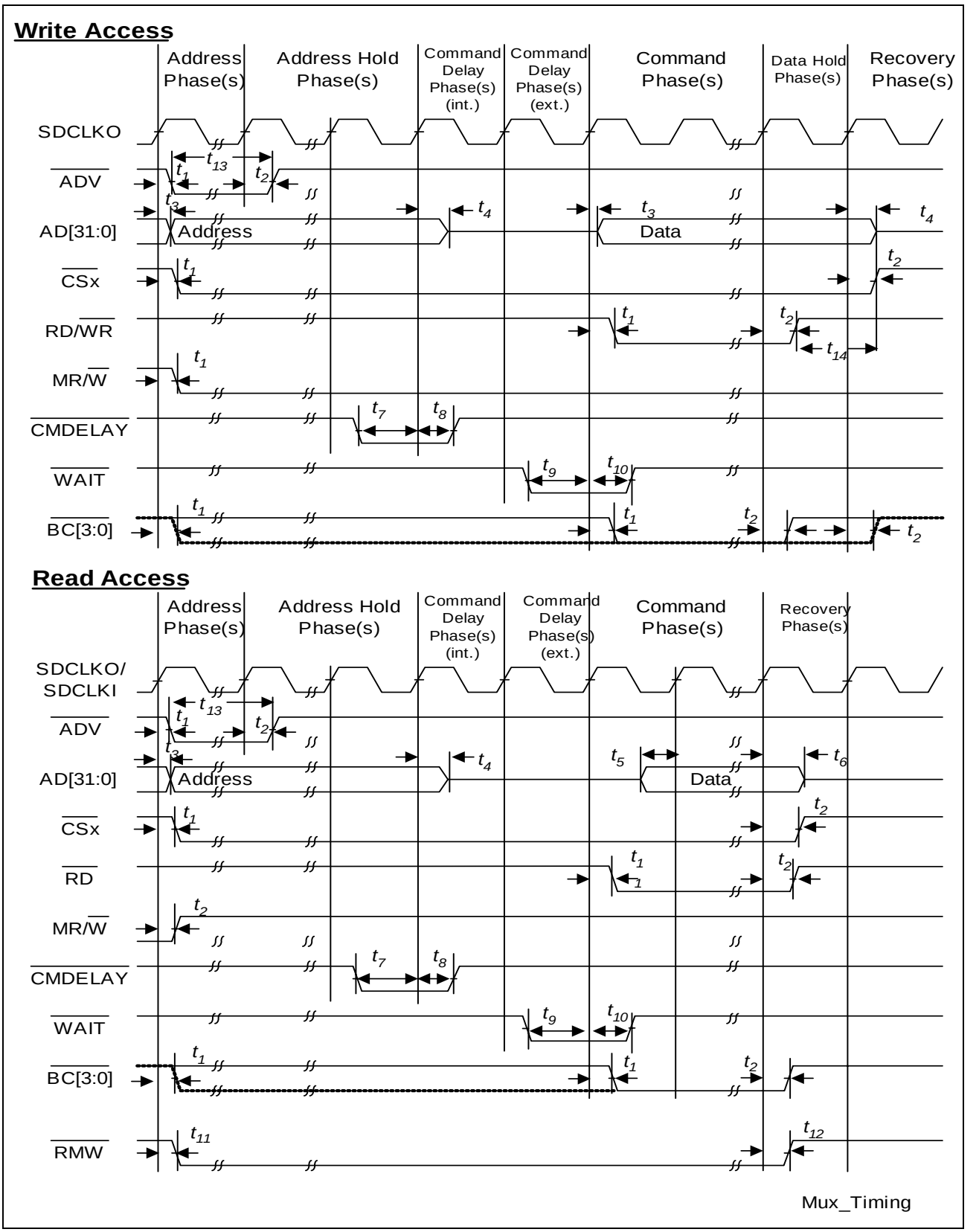


Figure 31 Write Access in Multiplexed Access

Timing for External Bus Arbitration Signals

(Operating Conditions apply; C_L = 50 pF)

Parameter	Symbol		Limits		Unit
			min	max	
$\overline{\text{HOLD}}$ input setup time to output clock	t_1	SR	7.3	–	ns
$\overline{\text{HOLD}}$ input hold time from output clock	t_2	SR	0.0	–	ns
$\overline{\text{HLDA}}$ output valid time from output clock	t_3	CC	–	6.2	ns
$\overline{\text{HLDA}}$ output hold time from output clock	t_4	CC	1.0	–	ns
$\overline{\text{HLDA}}$ input setup time to output clock	t_5	SR	7.4	–	ns
$\overline{\text{HLDA}}$ input hold time from output clock	t_6	SR	0.0	–	ns
$\overline{\text{BREQ}}$ output valid time from output clock	t_7	CC	–	6.4	ns
$\overline{\text{BREQ}}$ output hold time from output clock	t_8	CC	1.0	–	ns
$\overline{\text{CSx}}$ drive from EBUCLK	t_9	CC	–	3.1	ns
$\overline{\text{CSx}}$ high-impedance from EBUCLK	t_{10}	CC	–	3.1	ns
Other signals high-impedance from EBUCLK	t_{11}	CC	–	3.2	ns
Other signals drive from EBUCLK	t_{12}	CC	–	3.2	ns

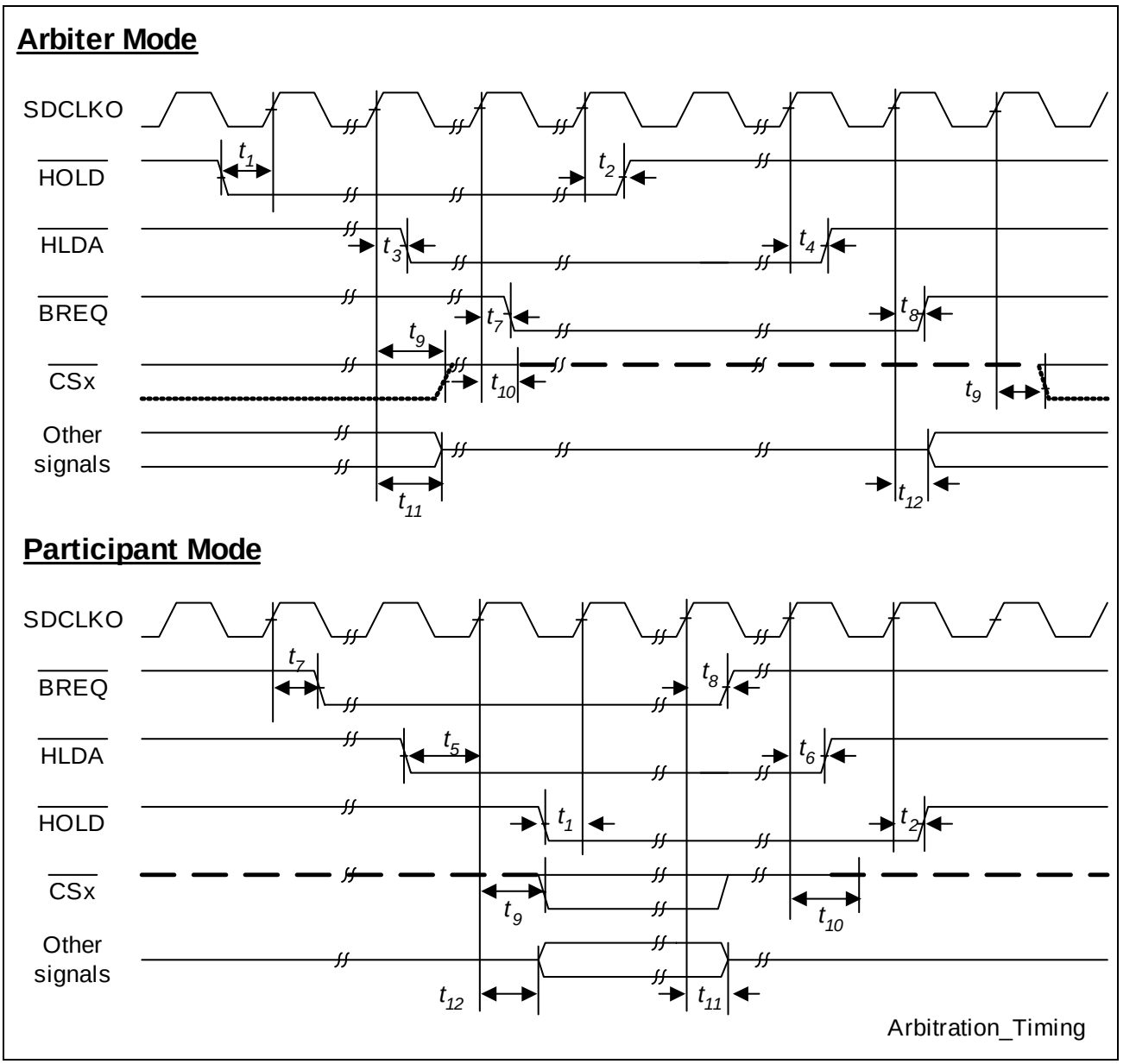


Figure 32 External Bus Arbitration Timing

Timing for Ethernet Signals
(Operating Conditions apply; C_L = 50 pF)

Parameter	Symbol		Limits		Unit
			min	max	
ETXCLK period (10 Mbps Ethernet)	t_1	SR	400.0	–	ns
ETXCLK high time (10 Mbps Ethernet)	t_2	SR	140	260	ns
ETXCLK low time (10 Mbps Ethernet)	t_3	SR	140	260	ns
ETXCLK period (100 Mbps Ethernet)	t_1	SR	40.0	–	ns
ETXCLK high time (100 Mbps Ethernet)	t_2	SR	14	26	ns
ETXCLK low time (100 Mbps Ethernet)	t_3	SR	14	26	ns
ERXCLK period (10 Mbps Ethernet)	t_1	SR	400.0	–	ns
ERXCLK high time (10 Mbps Ethernet)	t_2	SR	140	260	ns
ERXCLK low time (10 Mbps Ethernet)	t_3	SR	140	260	ns
ERXCLK period (100 Mbps Ethernet)	t_1	SR	40.0	–	ns
ERXCLK high time (100 Mbps Ethernet)	t_2	SR	14	26	ns
ERXCLK low time (100 Mbps Ethernet)	t_3	SR	14	26	ns
ERXD(3:0) input setup to ERXCLK	t_4	SR	10.0	–	ns
ERXD(3:0) input hold from ERXCLK	t_5	SR	–	10.0	ns
ERXDV input setup to ERXCLK	t_4	SR	10.0	–	ns
ERXDV input hold from ERXCLK	t_5	SR	–	10.0	ns
ERXER input setup to ERXCLK	t_4	SR	10.0	–	ns
ERXER input hold from ERXCLK	t_5	SR	–	10.0	ns
ETXD(3:0) output valid from ETXCLK	t_6	CC	–	25.0	ns
ETXEN output valid from ETXCLK	t_6	CC	–	25.0	ns
ETXER output valid from ETXCLK	t_6	CC	–	25.0	ns
EMDC clock period	t_7	CC	400.0	–	ns
EMDC high time	t_8	CC	160	–	ns
EMDC low time	t_9	CC	160	–	ns
EMDIO input setup to EMDC (sourced by STA)	t_{10}	SR	10.0	–	ns
EMDIO input hold from EMDC (sourced by STA)	t_{11}	SR	–	10.0	ns
EMDIO output valid from EMDC (sourced by PHY)	t_{12}	CC	–	300.0	ns

Note: Any other parameters which are not stated here, please refer to ANSI/IEEE Std 802.3, Section 22.3.

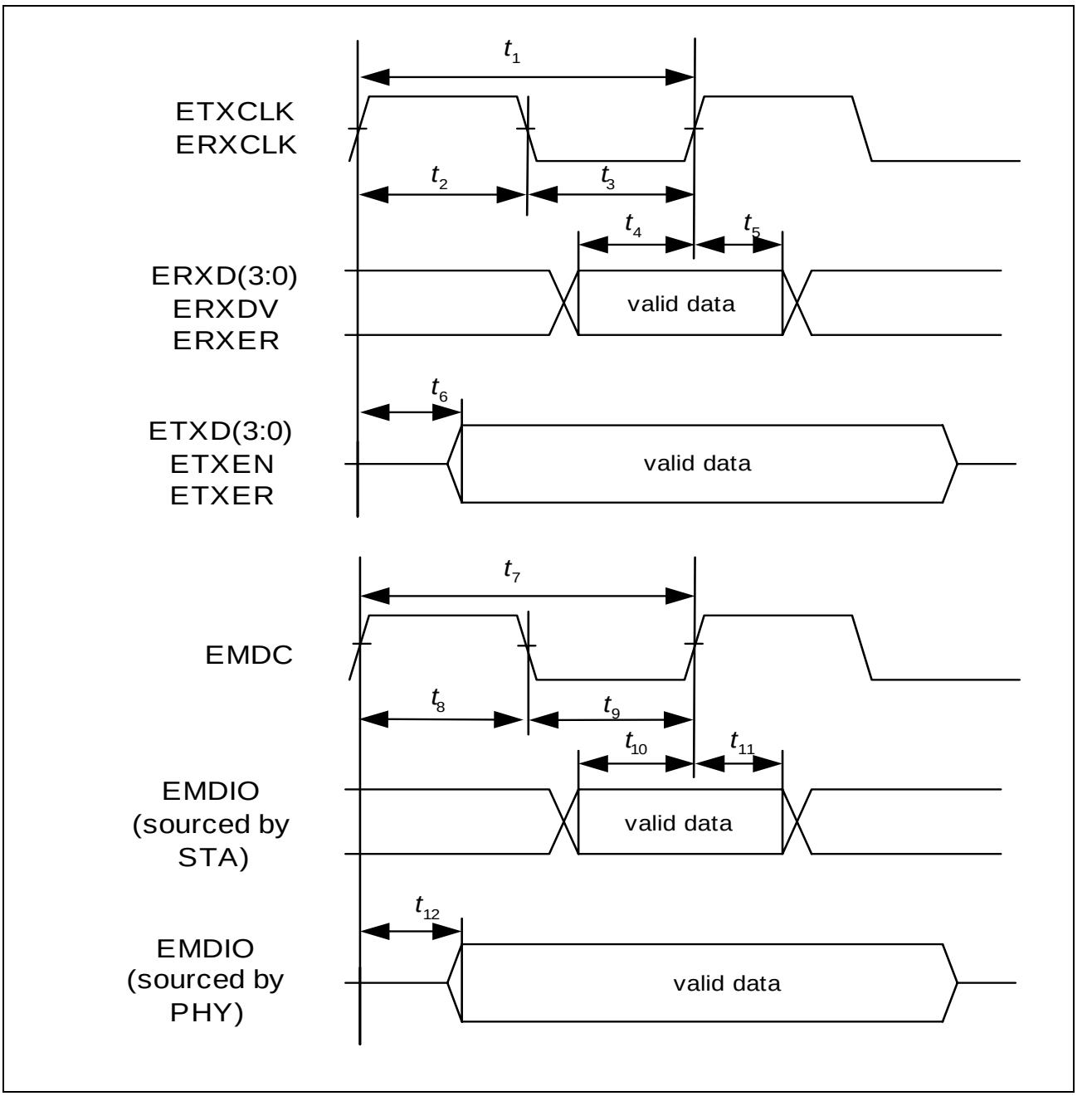


Figure 33 Ethernet Timing

SSC Master Mode Timing

(Operating Conditions apply; C_L = 50 pF)

Parameter	Symbol		Limit Values		Unit
			min.	max.	
SCLK clock frequency	$1 / t_{SCLK}$	CC	-	25	MHz
SCLK clock high time	t_1	CC	18	-	ns
SCLK clock low time	t_2	CC	18	-	ns
SCLK clock rise time	t_3	CC	-	11	ns
SCLK clock fall time	t_4	CC	-	11	ns
MTSR/SLSOx low/high from SCLK edge	t_5	CC	-	2.0	ns
MRST setup to SCLK edge	t_6	SR	7	-	ns
MRST hold from SCLK edge	t_7	SR	5	-	ns

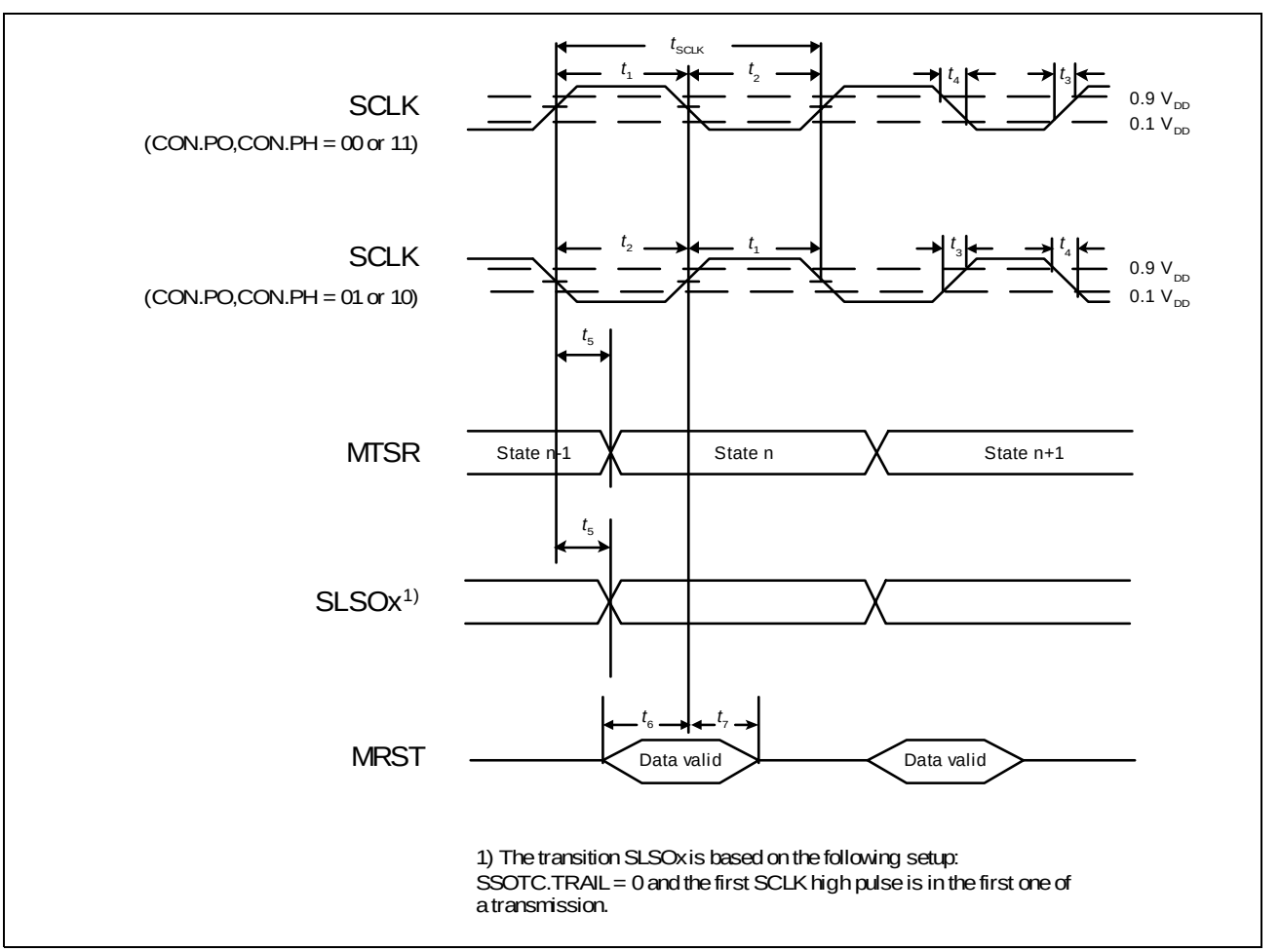


Figure 34 SSC Master Mode Timing

Timing for MLI Interface

(Operating Conditions apply; C_L = 50 pF)

Parameter	Symbol		Limit Values		Unit
			min.	max.	
TCLK/RCLK clock period	t_0	CC/SR	26.67	–	ns
TCLK high time	t_1	CC	9	–	ns
TCLK low time	t_2	CC	9	–	ns
TCLK rise time	t_3	CC	–	3	ns
TCLK fall time	t_4	CC	–	3	ns
TDATAx, TVALIDx outputs delay from TCLK ↑	t_5	CC	0	8	ns
TREADYx inputs setup to TCLK ↑	t_6	SR	tbd	–	ns
RDATAx, RVALIDx inputs setup to RCLK ↓	t_7	SR	5.3	–	ns
RDATAx, RVALIDx inputs hold from RCLK ↓	t_8	SR	tbd	–	ns
RREADYx outputs delay hold RCLK ↓	t_9	CC	tbd	tbd	ns

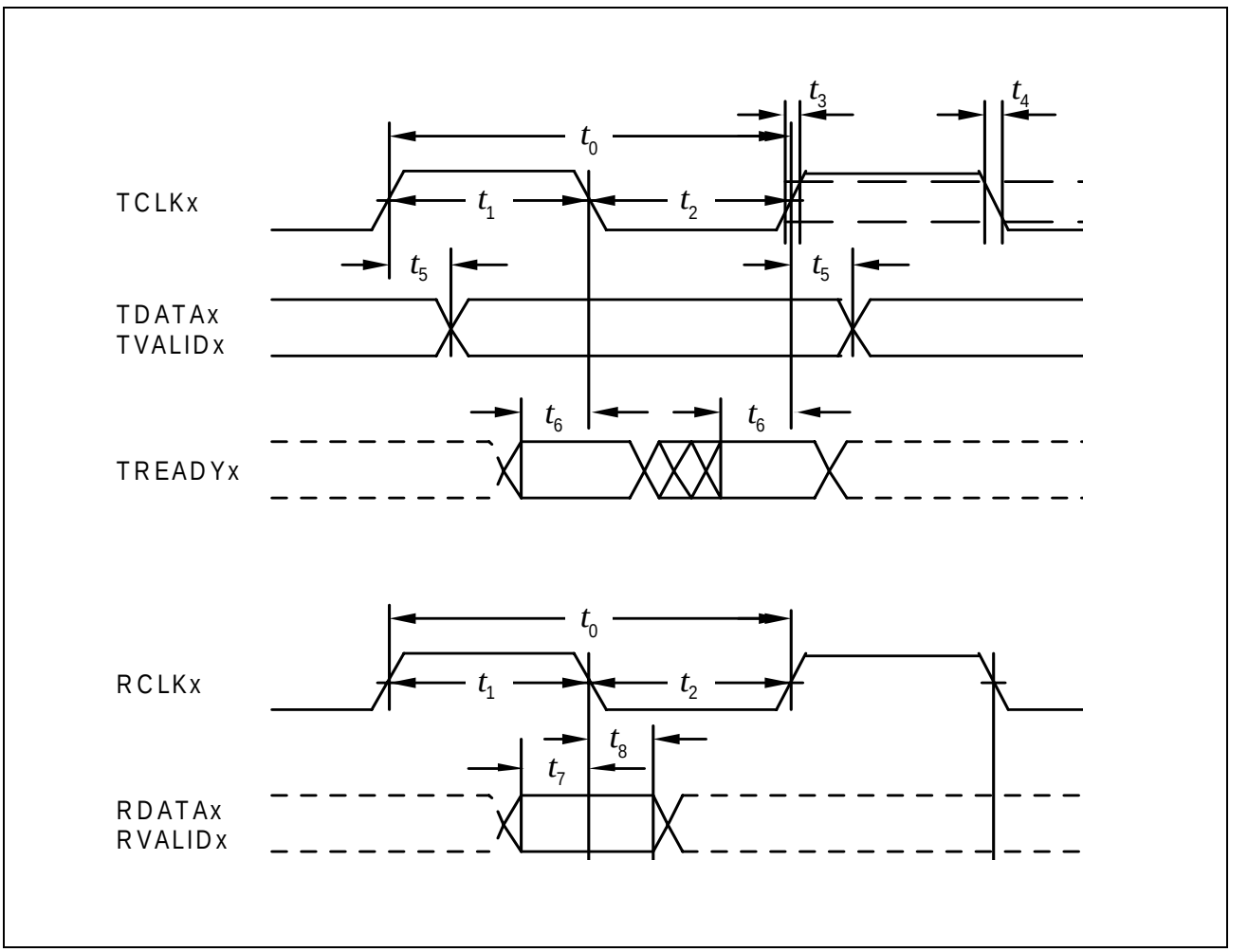


Figure 35 MLI Interface Timing

Note: The generation of RREADYx is in the input clock domain of the receiver. The reception of TXREADY is asynchronous to TCLKx (input synchronization with each edge of TCLKx). Meeting the setup time for TXREADY guarantees recognition of the TXREADY at a certain clock edge.

Timing for JTAG Signals

(Operating Conditions apply; C_L = 50 pF)

Parameter	Symbol	Limits		Unit
		min	max	
TCK clock period	t_{TCK} CC	50	–	ns
TCK high time	t_1 CC	10	–	ns
TCK low time	t_2 CC	29	–	ns
TCK clock rise time	t_3 CC	–	0.4	ns
TCK clock fall time	t_4 CC	–	0.4	ns

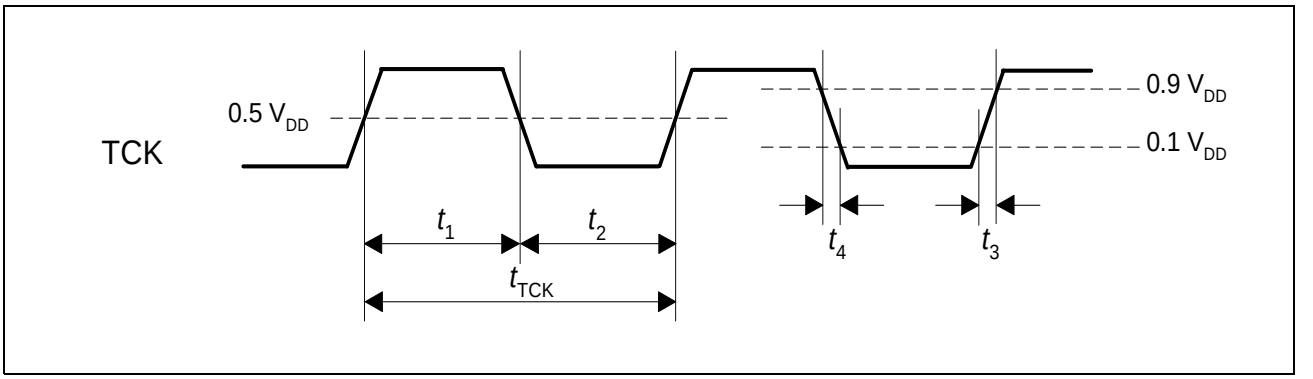


Figure 36 TCK Clock Timing

Parameter	Symbol		Limits		Unit
			min	max	
TMS setup to TCK ↑	t_1	SR	7.85	–	ns
TMS hold to TCK ↑	t_2	SR	3.0	–	ns
TDI setup to TCK ↑	t_1	SR	10.9	–	ns
TDI hold to TCK ↑	t_2	SR	3.0	–	ns
TDO valid output from TCK ↓	t_3	CC	–	10.7	ns
TDO high impedance to valid output from TCK ↓	t_4	CC	–	23.0	ns
TDO valid output to high impedance from TCK ↓	t_5	CC	–	26.0	ns

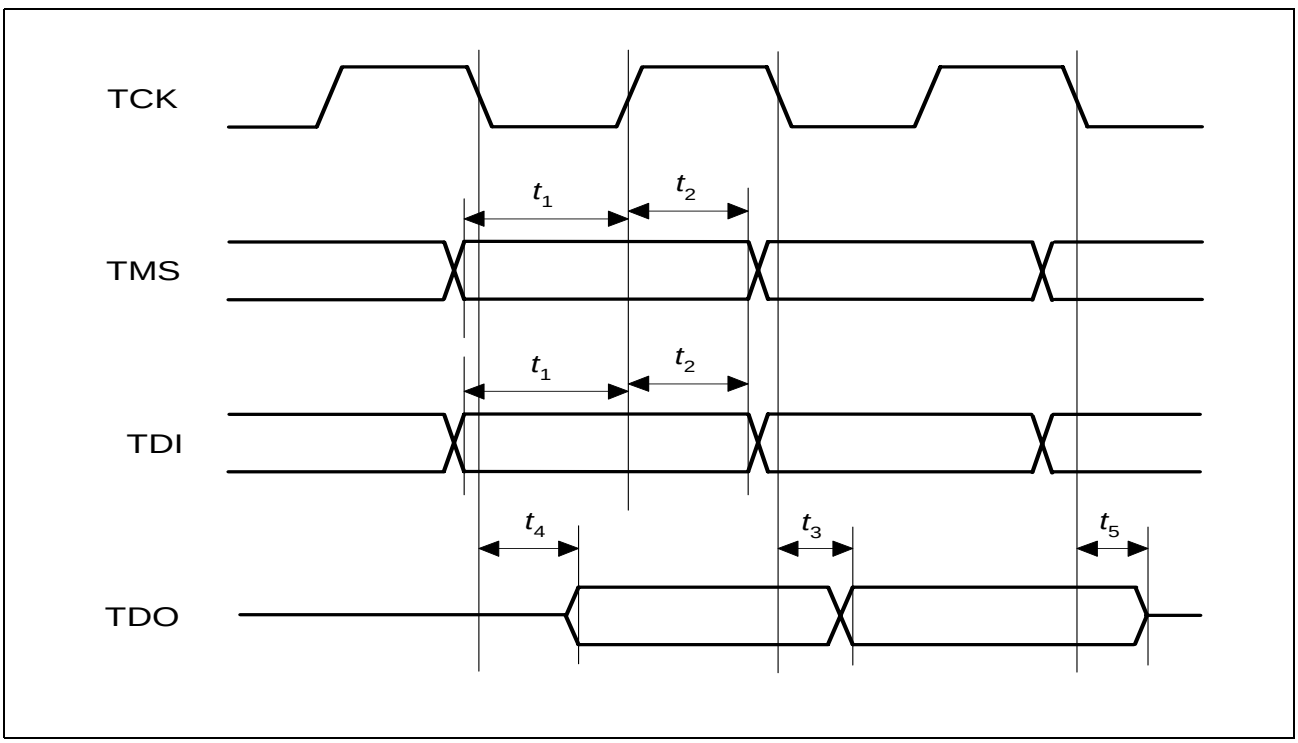
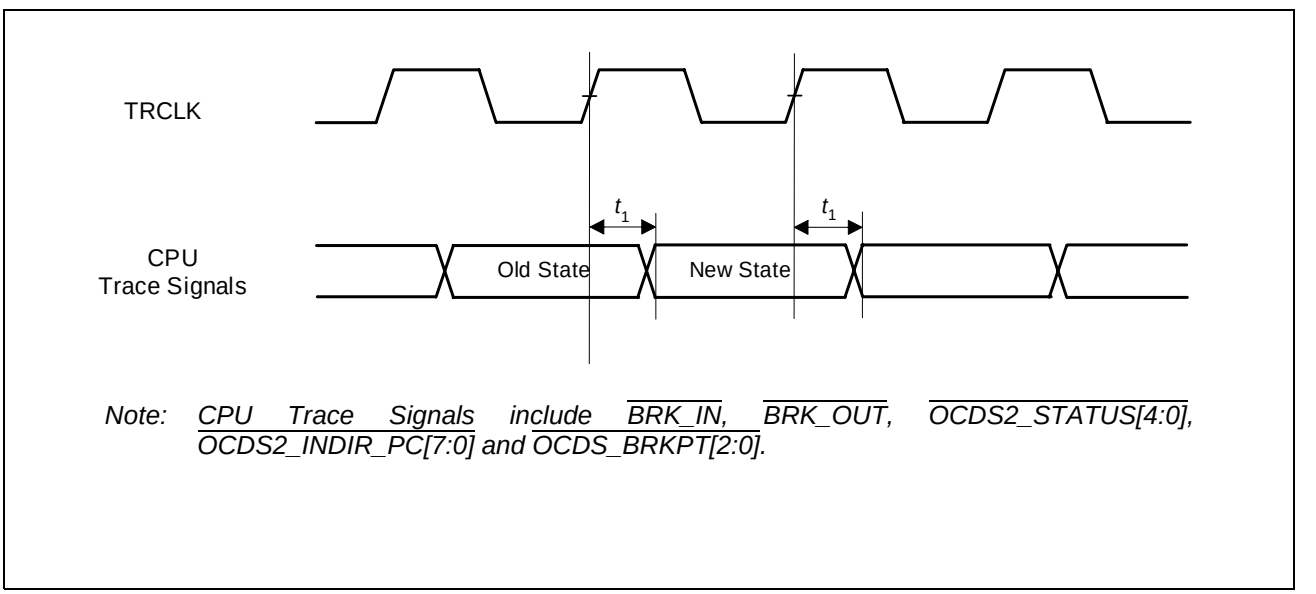


Figure 37 JTAG Timing

Timing for OCDS Trace and Breakpoint Signals

(Operating Conditions apply;C_L(TRCLK) = 25 pF, C_L = 50 pF)

Parameter	Symbol		Limits		Unit
			min	max	
$\overline{\text{BRK\_OUT}}$ valid from TRCLK ↓	t_1	CC	–	5.2	ns
$\overline{\text{OCDS2\_STATUS[4:0]}}$ valid from TRCLK ↓	t_1	CC	1.7	3.7	ns
$\overline{\text{OCDS2\_INDIR\_PC[7:0]}}$ valid from TRCLK ↓	t_1	CC	1.7	3.7	ns
$\overline{\text{OCDS2\_BRKPT[2:0]}}$ valid from TRCLK ↓	t_1	CC	1.7	3.7	ns



Note: CPU Trace Signals include $\overline{\text{BRK\_IN}}$, $\overline{\text{BRK\_OUT}}$, $\overline{\text{OCDS2\_STATUS[4:0]}}$, $\overline{\text{OCDS2\_INDIR\_PC[7:0]}}$ and $\overline{\text{OCDS\_BRKPT[2:0]}}$.

Figure 38 OCDS Trace Signals Timing

Timing for USB Transceiver Signals
(Operating Conditions apply; C_L = 50 pF)

Parameter	Symbol	Limits		Unit
		min	max	
Full speed mode rise time	t_{FR} CC	4	20	ns
Full speed mode fall time	t_{FF} CC	4	20	ns

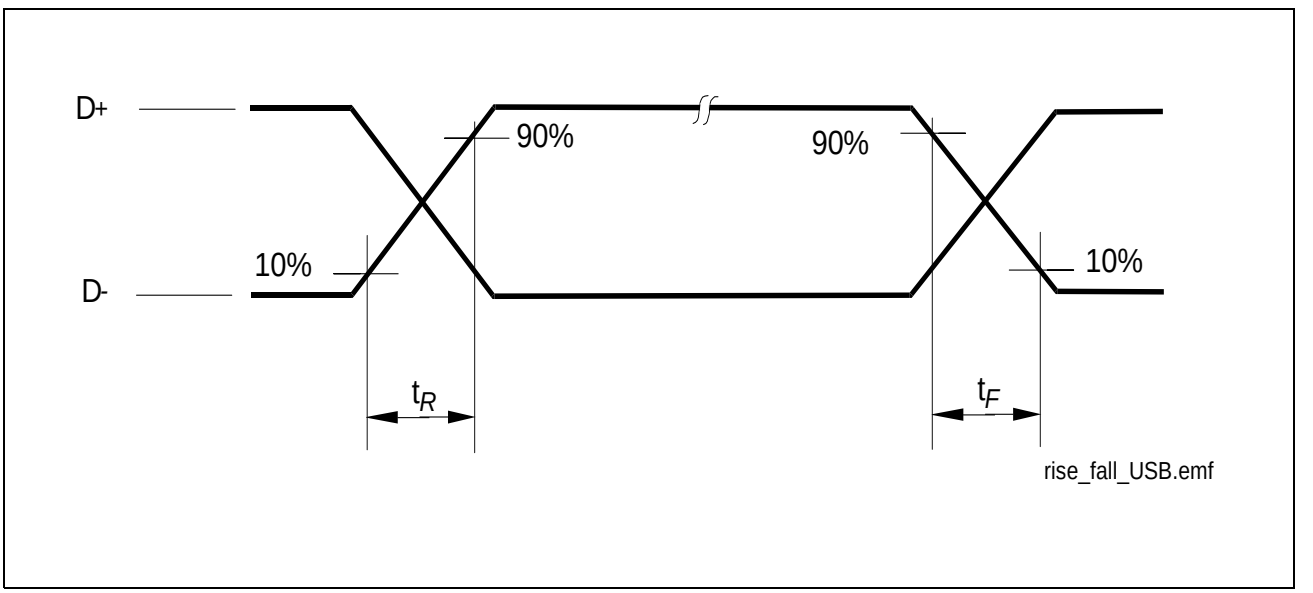


Figure 39 AC Testing: Input, Output Waveforms

Package Outline

Plastic Package, P-LBGA-208-2 (SMD)
(Low Profile Ball Grid Array Package)

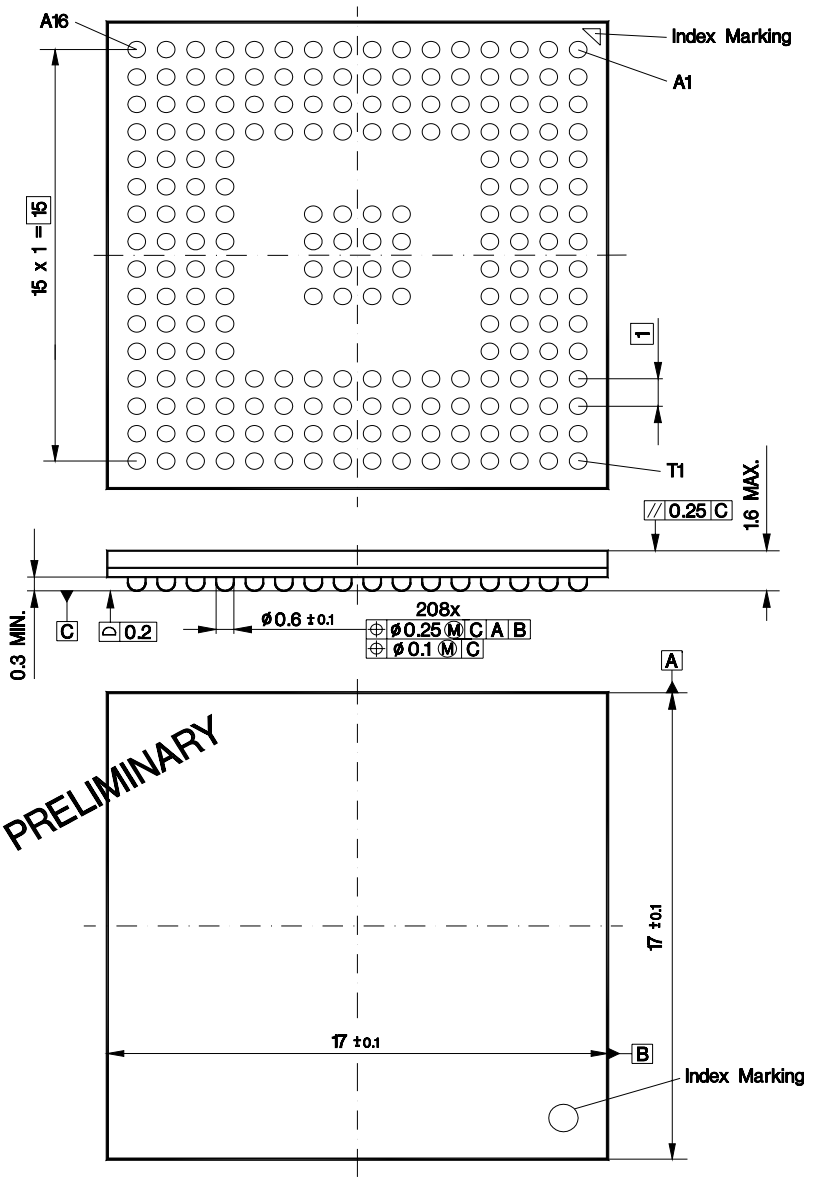


Figure 40 P-LBGA-208-2 Package

You can find all of our packages, sorts of packing and others in our Infineon Internet Page “Products”: http://www.infineon.com/products.

SMD = Surface Mounted Device

Dimensions in mm

Infineon goes for Business Excellence

“Business excellence means intelligent approaches and clearly defined processes, which are both constantly under review and ultimately lead to good operating results.
Better operating results and business excellence mean less idleness and wastefulness for all of us, more professional success, more accurate information, a better overview and, thereby, less frustration and more satisfaction.”

Dr. Ulrich Schumacher

http://www.infineon.com

Published by Infineon Technologies AG