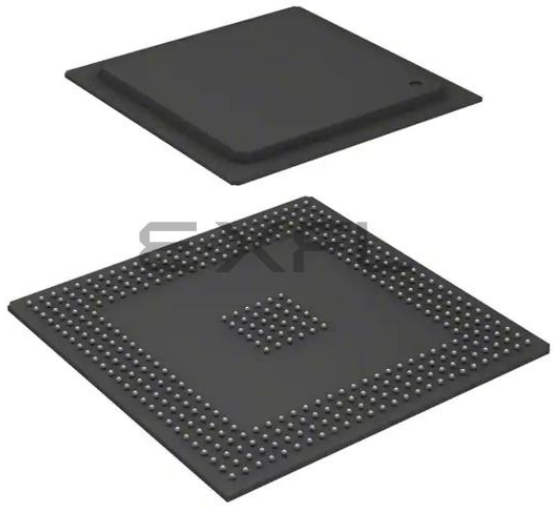

Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Not For New Designs
Core Processor	PowerPC
Core Size	32-Bit Single-Core
Speed	40MHz
Connectivity	CANbus, EBI/EMI, SCI, SPI, UART/USART
Peripherals	POR, PWM, WDT
Number of I/O	56
Program Memory Size	512KB (512K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	32K x 8
Voltage - Supply (Vcc/Vdd)	2.5V ~ 2.7V
Data Converters	A/D 32x10b
Oscillator Type	External
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	388-BBGA
Supplier Device Package	388-PBGA (27x27)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/mpc564czp40

Contents

Table 2-1. MPC561/MPC563 Signal Descriptions (continued)

Signal Name	No. of Signals	Type	Function after Reset[1]	Description
IRQ4 / AT2 / SGPIOC4	1	I	IRQ4	Interrupt Request 4. One of the eight external signals that can request, by means of the internal interrupt controller, a service routine from the RCPU.
		O		Address Type 2. A bit from the address type bus which indicates one of the 16 “address types” to which the address applies. The address type signals are valid at the rising edge of the clock in which the special transfer start (STS) is asserted.
		I/O		Port SGPIOC4. Allows the signal to be used as a general-purpose input/output.
IRQ5 / MODCK1 / SPGIOC5	1	I	MODCK1 until reset negates, then IRQ5	Interrupt Request 5. One of the eight external signals that can request, by means of the internal interrupt controller, a service routine from the RCPU.
		I		Mode Clock 1. Sampled at the negation of PORESET/TRST in order to configure the phase-locked loop (PLL)/clock mode of operation.
		I/O		Port SGPIOC5. Allows the signal to be used as a general-purpose input/output.
IRQ[6:7] / MODCK[2:3]	2	I	MODCK[2:3] until reset negates, then IRQ[6:7]	Interrupt Request [6:7]. One of the eight external signals that can request, by means of the internal interrupt controller, a service routine from the RCPU.
		I		Mode Clock [2:3]. Sampled at the negation of PORESET/TRST in order to configure the PLL/clock mode of operation.
CS[0:3]	4	O	CS[0:3]	Chip Select [0:3]. These output signals enable peripheral or memory devices at programmed addresses if defined appropriately in the memory controller. CS0 or CS3 can be configured to be the global chip select for the boot device.
WE[0:3] / BE[0:3] / AT[0:3]	4	O	Controlled by RCW[ATWC]. See Table 6-8.	Write Enable[0:3]/Byte Enable[0:3]. This output signal is asserted when a write access to an external slave controlled by the memory controller is initiated by the MPC561/MPC563. It can be optionally asserted on all read and write accesses. See WEBS bit definition in Table 10-8. WEn/BEn are asserted when data lanes shown below contain valid data to be stored by the slave device. – WE0/BE0 is asserted if the data lane DATA[0:7] contains valid data to be stored by the slave device. • WE1/BE1 is asserted if the data lane DATA[8:15] contains valid data to be stored by the slave device. • WE2/BE2 is asserted if the data lane DATA[16:23] contains valid data to be stored by the slave device. • WE3/BE3 is asserted if the data lane DATA[24:31] contains valid data to be stored by the slave device.
		O		Address Type [0:3]. Indicates one of the 16 address types to which the address applies. The address type signals are valid at the rising edge of the clock in which the special transfer start (STS) is asserted.

Table 2-6. PDMCR2 Field Description (continued)

Bits	Name	Description
26	PCS4EN[1]	Controls the pad B_T2CLK/PCS4. 0 B_T2CLK function will be selected. 1 PCS4 function will be selected.
27	PCS5EN	Controls the pad A_T2CLK/PCS5. 0 A_T2CLK function will be selected. 1 PCS5 function will be selected.
28	PCS6EN	Controls the pad ETRIG1/PCS6. 0 ETRIG1 function will be selected. 1 PCS6 function will be selected.
29	PCS7EN	Controls the pad ETRIG2/PCS7. 0 ETRIG2 function will be selected. 1 PCS7 function will be selected.
30:31	—	Reserved

[1] Refer to Table 2-9 for information regarding enhanced PCS functionality.

Table 2-7. TCNC Pad Functionalities

TCNC Values	TXD2/QGPO2/ C_CNTX0	RXD2/QGPI2/ C_CNRX0	MPIO32B11/ C_CNRX0	MPIO32B12/ C_CNTX0
00	TXD2/QGPO2	RXD2/QGPI2	MPIO32B11	MPIO32B12
x1	TXD2/QGPO2	RXD2/QGPI2	C_CNRX0	C_CNTX0
10	C_CNTX0	C_CNRX0	MPIO32B11	MPIO32B12

Table 2-8. PPMPAD Pad Functionalities

PPMPAD Values	MPIO32B14/ PPM_RX0	MPIO32B15/ PPM_TX0	MPWM3/ PPM_RX1	MPWM2/ PPM_TX1	MPIO32B10/ PPM_TSYNC	MPIO32B1/ PPM_TCLK
x00	MPIO32B14	MPIO32B15	MPWM3	MPWM2	MPIO32B10	MPIO32B13
001	MPIO32B14	PPM_TX0	MPWM3	MPWM2	PPM_TSYNC	PPM_TCLK
010	PPM_RX0	MPIO32B15	MPWM3	MPWM2	PPM_TSYNC	PPM_TCLK
011	PPM_RX0	PPM_TX0	MPWM3	MPWM2	PPM_TSYNC	PPM_TCLK
101	MPIO32B14	PPM_TX0	MPWM3	PPM_TX1	PPM_TSYNC	PPM_TCLK
110	PPM_RX0	MPIO32B15	PPM_RX1	MPWM2	PPM_TSYNC	PPM_TCLK
111	PPM_RX0	PPM_TX0	PPM_RX1	PPM_TX1	PPM_TSYNC	PPM_TCLK

Table 2-9. Enhanced PCS Functionality

PCS_IN[3:0]	PCS_OUT[7:0] IF PCSV = 0	PCS_OUT[7:0] IF PCSV = 1
0000	00000001	11111110
0001	00000010	11111101

9.5.7.3 Bus Busy

$\overline{BB}$ assertion indicates that the current bus master is using the bus. New masters should not begin transfer until this signal is negated. The bus owner should not relinquish or negate this signal until the transfer is complete. To avoid contention on the $\overline{BB}$ line, the master should three-state this signal when it gets a logical one value. This requires the connection of an external pull-up resistor to ensure that a master that acquires the bus is able to recognize the $\overline{BB}$ line negated, regardless of how many cycles have passed since the previous master relinquished the bus. Refer to Figure 9-25.

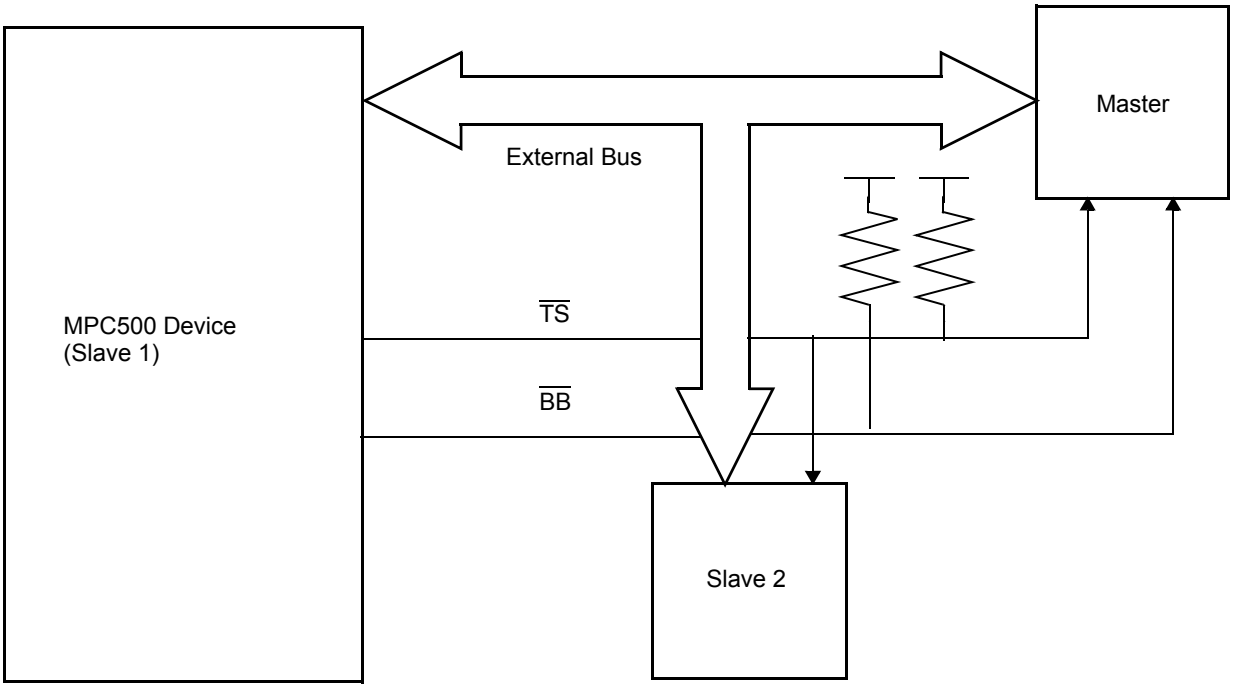

Figure 9-25. Master Signals Basic Connection

Show cycles are activated by properly setting the SIUMCR register bits. Refer to Section 6.2.2.1.1, “SIU Module Configuration Register (SIUMCR).” Construction visibility is controlled by the ISCT_SER bits in the ICTRL register. Refer to Table 23-26. Data visibility is controlled by the LSHOW bits of the L2U_MCR register. Refer to Table 11-7.

In a burst show cycle only the first data beat is shown externally. Refer to Table 9-8 for show cycle transaction encodings.

Instruction show cycle bus transactions have the following characteristics (see Figure 9-41):

- One clock cycle
- Address phase only; in decompression on mode part of the compressed address is driven on data lines together with address lines. The external bus interface adds one clock delay between a read cycle and such show cycle.
- $\overline{\text{STS}}$ assertion only (no $\overline{\text{TA}}$ assertion)

The compressed address is driven on the external bus in the following manner:

- ADDR[0:29] = the word base address;
- DATA[0] = operating mode:
  - 0 = decompression off mode;
  - 1 = decompression on mode;
- DATA[1:4] = bit pointer

See Chapter 4, “Burst Buffer Controller 2 Module” and Appendix A, “MPC562/MPC564 Compression Features” for more details about decompression mode.

	MSB 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	LSB 15
Field	CIE2	PIE2	SSE2	MQ2					RESUME	BQ2						
SRESET	0	0	0	0_0000					0	111_1111						
Addr	0x30 480E (QACR2_A), 0x30 4C0E (QACR2_B)															

Figure 13-11. Control Register 2 (QACR2)

Table 13-12. QACR2 Bit Descriptions

Bits	Name	Description
0	CIE2	Queue 2 Completion Software Interrupt Enable. CIE2 enables an interrupt upon completion of queue 2. The interrupt request is initiated when the conversion is complete for the CCW in queue 2. 0 Disable the queue completion interrupt associated with queue 2 1 Enable an interrupt after the conversion of the sample requested by the last CCW in queue 2
1	PIE2	Queue 2 Pause Software Interrupt Enable. PIE2 enables an interrupt when queue 2 enters the pause state. The interrupt request is initiated when conversion is complete for a CCW that has the pause bit set. 0 Disable the pause interrupt associated with queue 2 1 Enable an interrupt after the conversion of the sample requested by a CCW in queue 2 which has the pause bit set
2	SSE2	Queue 2 Single-Scan Enable Bit. SSE2 enables a single-scan of queue 2 to start after a trigger event occurs. The SSE2 bit may be set to a one during the same write cycle when the MQ2 bits are set for one of the single-scan queue operating modes. The single-scan enable bit can be written as a one or a zero, but is always read as a zero. The SSE2 bit enables a trigger event to initiate queue execution for any single-scan operation on queue 2. The QADC64E clears the SSE2 bit when the single-scan is complete. Refer to Table 13-13 for more information. 0 Trigger events are not accepted for single-scan modes 1 Accept a trigger event to start queue 2 in a single-scan mode
3:7	MQ2	Queue 2 Operating Mode. The MQ2 field selects the queue operating mode for queue 2. Refer to Table 13-13 for more information.
8	RESUME	0 After suspension, begin executing with the first CCW in queue 2 or the current sub-queue 1 After suspension, begin executing with the aborted CCW in queue 2

continuously and queue 2, being lower in priority, never gets executed. The short interval of time between a queue 1 completion and the subsequent trigger event is not sufficient to allow queue 2 execution to begin.

The software initiated continuous-scan mode is a useful choice with queue 2 for converting channels that do not need to be synchronized to anything, or for the slow-to-change analog channels. Interrupts are normally not used with the software initiated continuous-scan mode. Rather, the software reads the latest conversion result from the result table at any time. Once initiated, software action is not needed to sustain conversions of channel.

13.5.4.4.2 External Trigger Continuous-Scan Mode

The QADC64E provides external trigger signals for both queues. When the external trigger software initiated continuous-scan mode is selected, a transition on the associated external trigger signal initiates queue execution. The polarity of the external trigger signal is programmable, so that the software can select a mode which begins queue execution on the rising or falling edge. Each CCW is read and the indicated conversions are performed until an end-of-queue condition is encountered. When the next external trigger edge is detected, the queue execution begins again automatically. Software initialization is not needed between trigger events.

When a pause bit is encountered in external trigger continuous-scan mode, another trigger event is required for queue execution to continue. Software involvement is not needed to enable queue execution to continue from the paused state.

Some applications need to synchronize the sampling of analog channels to external events. There are cases when it is not possible to use software initiation of the queue scan sequence, since interrupt response times vary.

13.5.4.4.3 External Gated Continuous-Scan Mode

The QADC64E provides external gating for queue 1 only. When external gated continuous-scan mode is selected, the input level on the associated external trigger signal enables and disables queue execution. The polarity of the external gated signal is fixed so a high level opens the gate and a low level closes the gate. Once the gate is open, each CCW is read and the indicated conversions are performed until the gate is closed. When the gate opens again, the queue execution automatically begins again from the beginning of the queue. Software initialization is not needed between trigger events. If a pause in a CCW is encountered, the pause flag *will not set*, and execution continues without pausing.

The purpose of external gated continuous-scan mode is to continuously collect digitized samples while the gate is open and to have the most recent samples available. It is up to the programmer to ensure that the queue is large enough so that a maximum gate open time will not reach an end-of-queue. However it is useful to take advantage of a smaller queue in the manner described in the next paragraph.

In the event that the queue completes before the gate closes, a completion flag will be set and the queue will roll over to the beginning and continue conversions until the gate closes. If the gate remains open and the completion flag is not cleared, when the queue completes a second time the trigger overrun flag will be set and the queue will roll-over again. The queue will continue to execute until the gate closes or the mode is disabled.

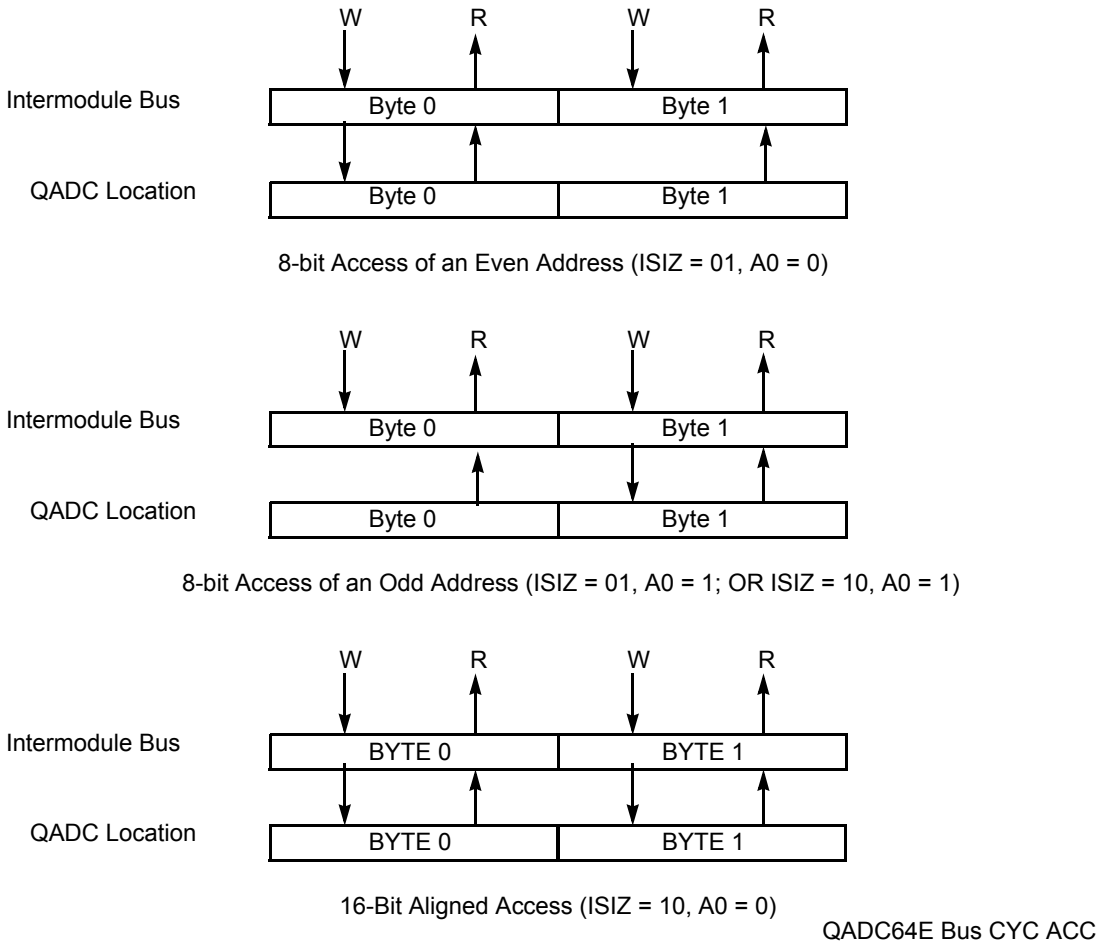

Figure 13-26. Bus Cycle Accesses

Byte access to an even address of a QADC64E location is shown in the top illustration of Figure 13-26. In the case of write cycles, byte 1 of the register is not disturbed. In the case of a read cycle, the QADC64E provides both byte 0 and byte 1.

Byte access to an odd address of a QADC64E location is shown in the center illustration of Figure 13-26. In the case of write cycles, byte 0 of the register is not disturbed. In the case of read cycles, the QADC64E provides both byte 0 and byte 1.

16-bit accesses to an even address read or write byte 0 and byte 1 as shown in the lowest illustration of Figure 13-26. The full 16 bits of data is written to and read from the QADC64E location with each access.

16-bit accesses to an odd address require two bus cycles; one byte of two different 16-bit QADC64E locations is accessed. The first bus cycle is treated by the QADC64E as an 8-bit read or write of an odd address. The second cycle is an 8-bit read or write of an even address. The QADC64E address space is organized into 16-bit even address locations, so a 16-bit read or write of an odd address obtains or provides the lower half of one QADC64E location, and the upper half of the following QADC64E location.

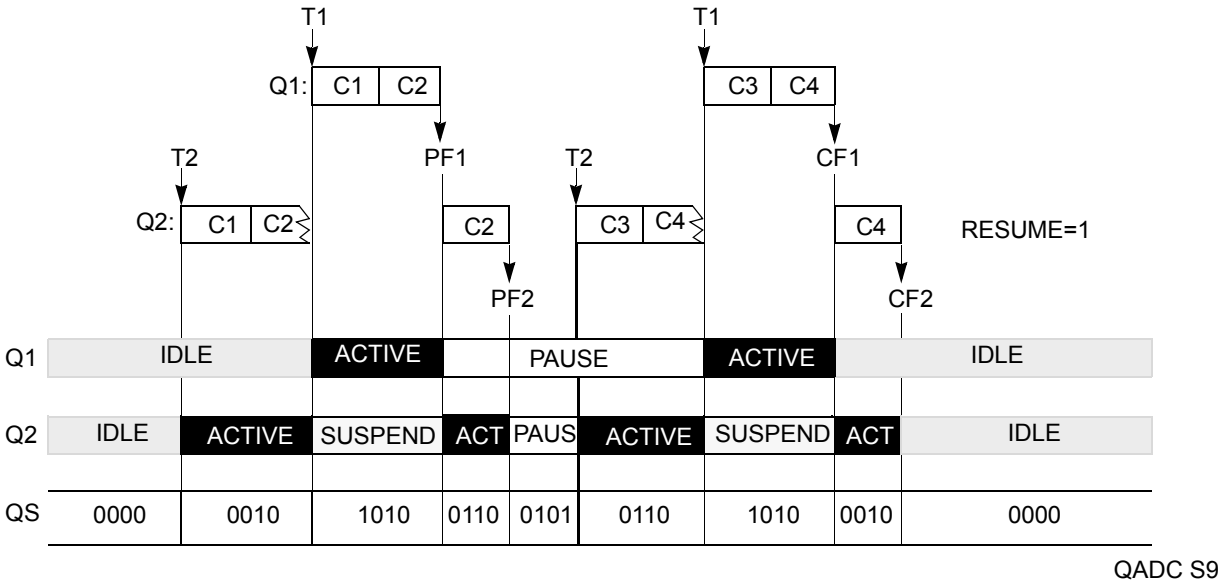

Figure 13-35. CCW Priority Situation 9

Situations S10 and S11 (Figure 13-36 and Figure 13-37) show that when an additional trigger event is detected for queue 2 while the queue is suspended, the trigger overrun error bit is set, the same as if queue 2 were being executed when a new trigger event occurs. Trigger overrun on queue 2 thus permits the software to know that queue 1 is taking up so much QADC64E time that queue 2 trigger events are being lost.

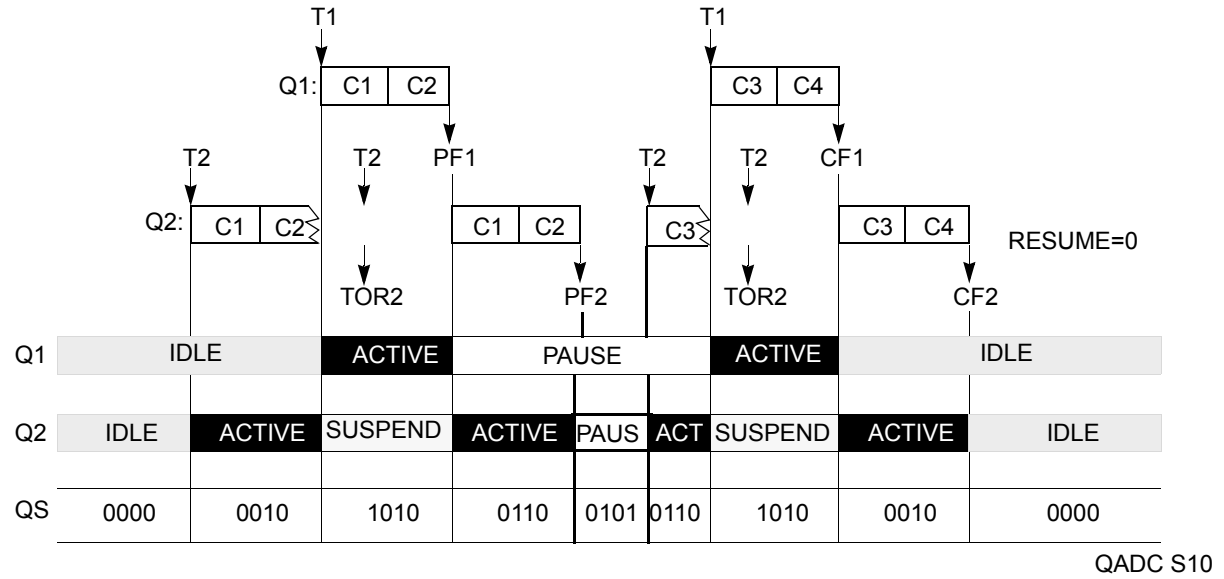

Figure 13-36. CCW Priority Situation 10

Therefore, conversion time requires a minimum of 14 QCLK clocks (seven µs with a 2.0-MHz QCLK). If the maximum final sample time period of 8 QCLKs is selected, the total conversion time is 20 QCLKs (10 µs with a 2.0-MHz QCLK)

Figure 14-21 illustrates the timing for conversions.

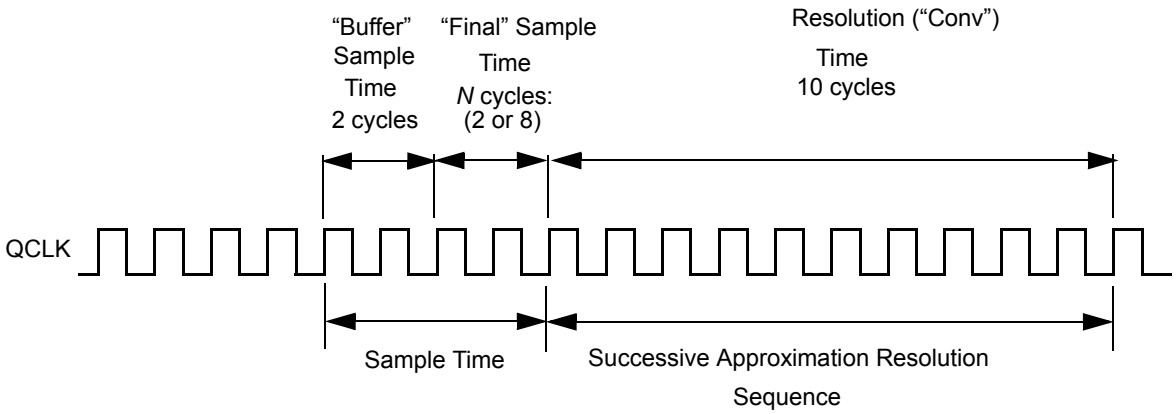

Figure 14-21. Conversion Timing

14.3.12 Channel Decode and Multiplexer

The internal multiplexer selects one of the 16 analog input signals for conversion. The selected input is connected to the sample buffer amplifier. The multiplexer also includes positive and negative stress protection circuitry, which prevents deselected channels from affecting the selected channel when current is injected into the deselected channels. Refer to Appendix F, "Electrical Characteristics," for specific current levels.

14.3.13 Sample Buffer Amplifier

The sample buffer is used to raise the effective input impedance of the A/D converter, so that external components (higher bandwidth or higher impedance) are less critical to accuracy. The input voltage is buffered onto the sample capacitor to reduce crosstalk between channels.

14.3.14 Digital to Analog Converter (DAC) Array

The digital to analog converter (DAC) array consists of binary-weighted capacitors and a resistor-divider chain. The reference voltages, V_{RH} and V_{RL}, are used by the DAC to perform ratiometric conversions. The DAC also converts the following three internal channels:

- V_{RH} — Reference voltage high
- V_{RL} — Reference voltage low
- $(V_{RH} - V_{RL})/2$ — Reference voltage

The DAC array serves to provide a mechanism for the successive approximation A/D conversion.

MSB 0	1	2	3	4	5	6	LSB 7
CONT	BITSE	DT	DSCK	PCS3	PCS2	PCS1	PCS0[1]
—	—	—	—	—	—	—	—
CONT	BITSE	DT	DSCK	PCS3	PCS2	PCS1	PCS0[1]
Command Control				Peripheral Chip Select			

The PCS0 bit represents the dual-function PCS0/$\overline{SS}$.

Figure 15-17. CR[0:F] — Command RAM 0x30 51C0, 0x30 51DF

Table 15-19. Command RAM Bit Descriptions

Bits	Name	Description
0	CONT	Continue 0 Control of chip selects returned to PORTQS after transfer is complete. 1 Peripheral chip selects remain asserted after transfer is complete.
1	BITSE	Bits per transfer enable 0 Eight bits 1 Number of bits set in BITS field of SPCR0.
2	DT	Delay after transfer 0 Delay after transfer is 17 ÷ f_{SYS}. 1 SPCR1 DTL[7:0] specifies delay after transfer PCS valid to SCK.
3	DSCK	PCS to SCK Delay 0 PCS valid to SCK delay is one-half SCK. 1 SPCR1 DSCKL[6:0] specifies delay from PCS valid to SCK.
4:7	PCS[3:0]	Peripheral chip selects. Use peripheral chip-select bits to select an external device for serial data transfer. More than one peripheral chip select may be activated at a time, and more than one peripheral chip can be connected to each PCS pin, provided proper fanout is observed. PCS0 shares a pin with the slave select ($\overline{SS}$) signal, which initiates slave mode serial transfer. If $\overline{SS}$ is taken low when the QSPI is in master mode, a mode fault occurs.

Refer to Section 15.6.5, “Master Mode Operation” for more information on the command RAM.

15.6.3 QSPI Pins

Seven pins are associated with the QSPI. When not needed by the QSPI, they can be configured for general-purpose I/O. Table 15-20 identifies the QSPI pins and their functions. Register DDRQS determines whether the pins are designated as input or output. The user must initialize DDRQS for the QSPI to function correctly.

Figure 17-2. MIOS14 Memory Map

17.5 MIOS14 I/O Ports

Each signal of each submodule can be used as an input, output, or I/O port:

Table 17-2. MIOS14 I/O Ports

Submodule	Number of Pins per Module	Type
MPIOSM	16	I/O
MMCSM	2	I
MDASM	1	I/O
MPWMSM	1	I/O

17.6 MIOS14 Bus Interface Submodule (MBISM)

The MIOS14 bus interface submodule (MBISM) is used as an interface between the MIOB (modular I/O bus) and the IMB3. It allows the CPU to communicate with the MIOS14 submodules.

17.6.1 MIOS14 Bus Interface (MBISM) Registers

Table 17-3 is the address map for the MBISM submodule.

	MSB 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	LSB 15
0x30 6800	MIOS14 Test and Signal Control Register (MIOS14TPCR)															
0x30 6802	MIOS14 Vector Register (MIOS14VECT) -Reserved															
0x30 6804	MIOS14 Module-Version Number Register (MIOS14VNR)															
0x30 6806	MIOS14 Module Control Register (MIOS14MCR)															
0x30 6808	Reserved															
0x30 680A	Reserved															
0x30 680C	Reserved															
0x30 680E	Reserved															

Figure 17-3. MBISM Registers

17.6.1.1 MIOS14 Test and Signal Control Register (MIOS14TPCR)

This register is used for MIOS14 factory testing and to control the VF and VFLS Signal usage. Control of other multiplexed functions is in the PDMCR2 register.

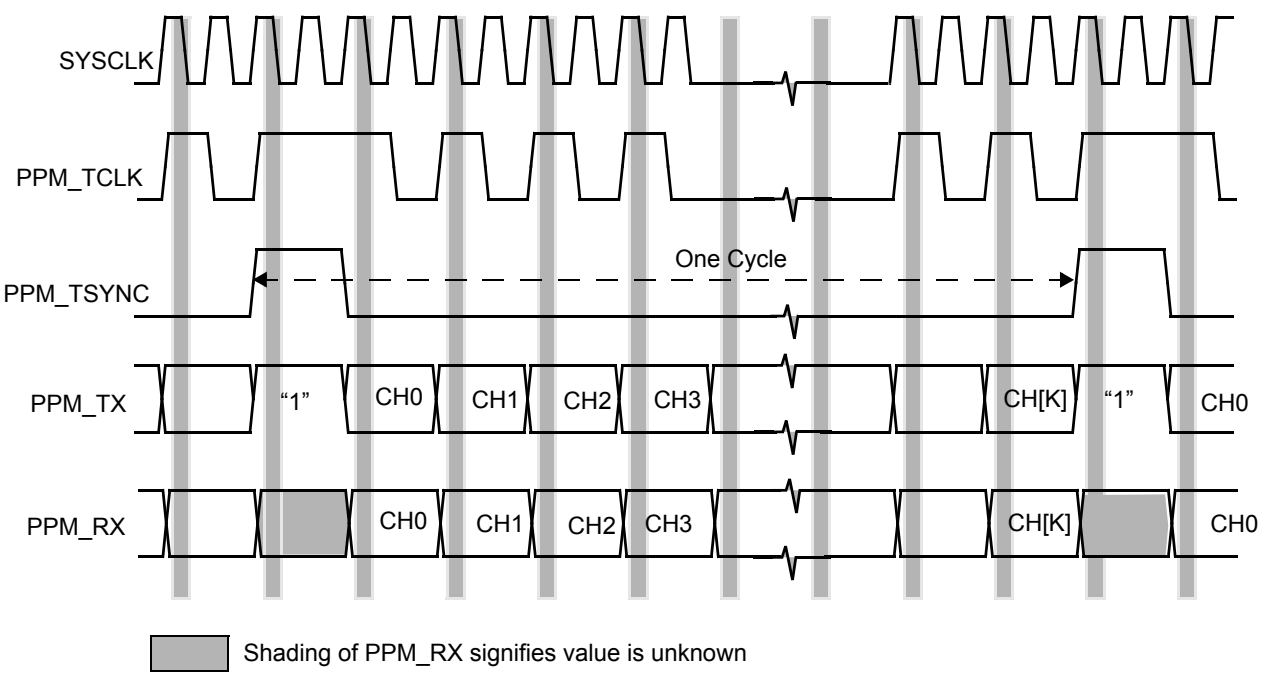

Figure 18-6. One Transmit and Receive Cycle in SPI Mode

18.3.1.3 PPM Control Settings

As data is transferred through the PPM module it must be sampled at a rate which guarantees its validity. This sample rate is a multiple of PPM_TCLK and is defined by the SAMP[0:2] field of the PPMPCR register. For transmit operations, the sample rate is the rate at which TX_DATA receives data from the internal modules. For receive operations, it is the rate at which the internal modules read RX_SHIFTER. The register RX_DATA is updated from RX_SHIFTER on completed receive (PPM_TSYNC) cycles.

Table 19-18. TPUMCR2 Bit Descriptions (continued)

Bits	Name	Description
8	SOFT RST	Soft reset. The TPU3 performs an internal reset when both the SOFT RST bit in the TPUMCR2 and the STOP bit in TPUMCR are set. The CPU must write zero to the SOFT RST bit to bring the TPU3 out of reset. The SOFT RST bit must be asserted for at least nine clocks. 0 Normal operation 1 Puts TPU3 in reset until bit is cleared NOTE: Do not attempt to access any other TPU3 registers when this bit is asserted. When this bit is asserted, it is the only accessible bit in the register.
9:10	ETBANK	Entry table bank select. This field determines the bank where the microcoded entry table is situated. After reset, this field is 0b00. This control bit field is write once after reset. ETBANK is used when the microcode contains entry tables not located in the default bank 0. To execute the ROM functions on this MCU, ETBANK[1:0] must be 00. Refer to Table 19-19. NOTE: This field should not be modified by the programmer unless necessary because of custom microcode.
11:13	FPSCK	Filter prescaler clock. The filter prescaler clock control bit field determines the ratio between system clock frequency and minimum detectable pulses. The reset value of these bits is zero, defining the filter clock as four system clocks. Refer to Table 19-20.
14	T2CF	T2CLK pin filter control. When asserted, the T2CLK input pin is filtered with the same filter clock that is supplied to the channels. This control bit is write once after reset. 0 Uses fixed four-clock filter 1 T2CLK input pin filtered with same filter clock that is supplied to the channels
15	DTPU	Disable TPU3 pins. When the disable TPU3 control pin is asserted, pin TP15 is configured as an input disable pin. When the TP15 pin value is zero, all TPU3 output pins are three-stated, regardless of the pins function. The input is not synchronized. This control bit is write once after reset. 0 TP15 functions as normal TPU3 channel 1 TP15 pin configured as output disable pin. When TP15 pin is low, all TPU3 output pins are in a high-impedance state, regardless of the pin function.

Table 19-19. Entry Table Bank Location

ETBANK	Bank
00	0
01	1
10	2
11	3

Table 19-20. System Clock Frequency/Minimum Guaranteed Detected Pulse

Filter Control	Divide By	20 MHz	33 MHz	40 MHz	56 MHz
000	4	200 ns	121 ns	100 ns	71 ns
001	8	400 ns	242 ns	200 ns	143 ns
010	16	800 ns	485 ns	400 ns	286 ns
011	32	1.6 μs	970 ns	800 ns	571 ns
100	64	3.2 μs	1.94 μs	1.60 μs	1.14 μs

Chapter 20
Dual-Port TPU3 RAM (DPTRAM)

The dual-port RAM (DPTRAM) module with TPU3 microcode storage support consists of a control register block and an 8-Kbyte array of static RAM, which can be used either as a microcode storage for TPU3 or as a general-purpose memory. The MPC561/MPC563 has one DPTRAM module. The module serves two TPU3 modules (A and B).

The DPTRAM module acts as a common memory on the IMB3 and allows the transfer of data to the two TPU3 modules. Therefore, the DPTRAM interface includes an IMB3 bus interface and two TPU3 interfaces. When the DPTRAM is being used in microcode mode, the array is only accessible to the TPU3 via a separate local bus, and not via the IMB3.

In the MPC561/MPC563, the DPTRAM base address register (RAMBAR) must be set to a particular value to fit into the IMB memory map of the part. The DPTRAM RAMBAR register *must* be programmed to 0xFFA0.

The DPTRAM module is powered by V_{DD} in normal operation. The entire array may be used as standby RAM if standby power is supplied via the IRAMSTBY pin of the MPC561/MPC563. IRAMSTBY must be supplied by an external source.

The DPTRAM may also be used as the microcode control store for up to two TPU3 modules when placed in a special emulation mode. In this mode the DPTRAM array may only be accessed by either or both of the TPU3 units simultaneously via separate emulation buses, and not via the IMB3.

The DPTRAM contains a multiple input signature calculator (MISC) in order to provide RAM data corruption checking. The MISC reads the DPTRAM address and generates a 32-bit data-dependent signature. This signature can then be checked by the host.

NOTE

The RCPU cannot perform instruction fetches from any module on the IMB3 (including the DPTRAM). Only data accesses are permitted.

20.1 Features

- Eight Kbytes of static RAM
- Accessible by the CPU only if neither TPU3 is in emulation mode
- Low-power stop operation
  - — Entered by setting the STOP bit in the DPTMCR
  - — Does not enter low-power state while in TPU3 emulation mode for protection
- TPU3 microcode mode

- Run-time access to on-chip memory map and MPC500 special purpose registers (SPRs) via the READI read/write access protocol. This feature supports accesses for runtime internal visibility, calibration constant acquisition and tuning, and external rapid prototyping for powertrain automotive development systems.
- Watchpoint messaging via the auxiliary port
- Nine or 16 full-duplex auxiliary signal interface for medium and high visibility throughput
  - One of two modes selected during reset: full port mode (FPM) and reduced port mode (RPM).
    - FPM comprises 16 signals and RPM comprises nine signals
  - Auxiliary output port
    - One MCKO (message clock out) signal
    - Two or eight MDO (message data out) signals
    - One $\overline{\text{MSEO}}$ (message start/end out) signal
  - Auxiliary input port
    - One $\overline{\text{MCKI}}$ (message clock in) signal
    - One or two MDI (message data in) signals
    - One $\overline{\text{MSEI}}$ (message start/end in) signal
    - One $\overline{\text{EVTI}}$ (event in) signal
    - One $\overline{\text{RSTI}}$ (reset in) signal
- All features configurable and controllable via the auxiliary port
- Security features for production environment
- Support of existing RCPU development access protocol via the auxiliary port
- READI module can be reset independent of system reset
- Parametrics:
  - Two bits are downloaded per clock in full port mode. For example, with input clock running at 28 MHz, this translates to a download rate of 56 Mbits/s.
  - One bit is downloaded per clock in reduced port mode. For example, with input clock running at 28 MHz, this translates to a download rate of 28 Mbits/s.
  - Eight bits are uploaded per clock in full port mode. For example, with system clock running at 56 MHz, this translates to a upload rate of 448 Mbits/s.
  - Two bits are uploaded per clock in reduced port mode. For example, with system clock running at 56 MHz, this translates to a upload rate of 112 Mbits/s.

24.1.1 Functional Block Diagram

The functional block diagram of the READI module is shown in Figure 24-1.

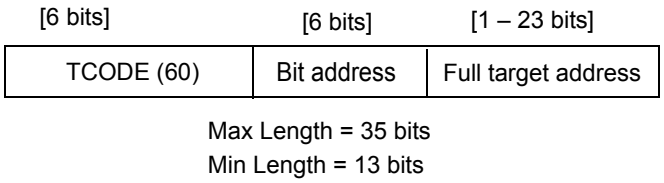

Figure 24-28. Direct Branch Synchronization Message Format with Compressed Code (PTSM = 0)

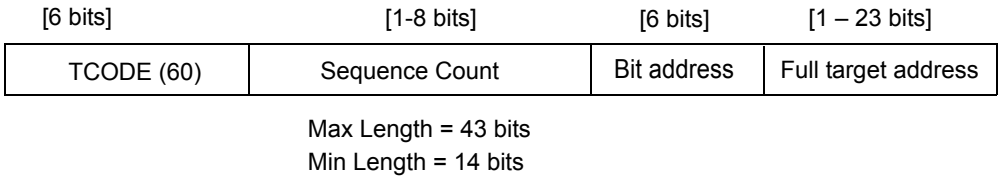

Figure 24-29. Direct Branch Synchronization Message Format with Compressed Code (PTSM = 1)

24.8.2.4.4 Indirect Branch Synchronization Message with Compressed Code

For compressed code support, six additional bits indicate the starting bit address within the word of the compressed instruction. The program trace indirect branch synchronization with compressed code message has the following formats depending on the setting of MC[PTSM]:

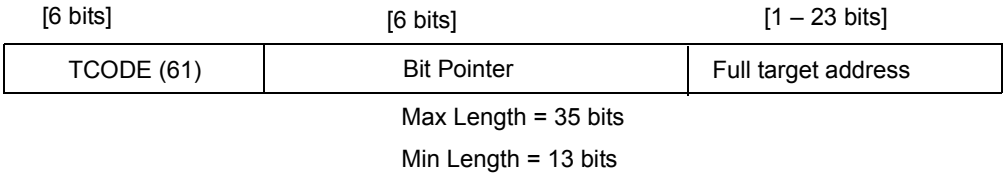

Figure 24-30. Indirect Branch Synchronization Message Format with Compressed Code (PTSM - 0)

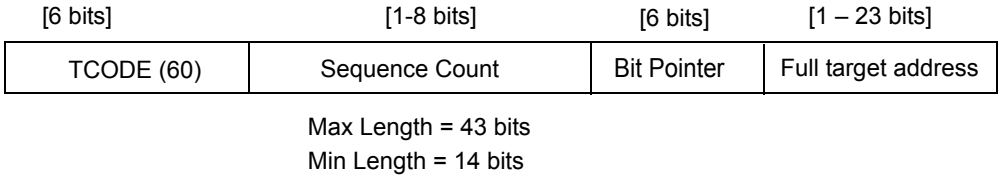

Figure 24-31. Indirect Branch Synchronization Message Format with Compressed Code (PTSM = 1)

Bit pointer format is shown in Figure 24-22 and bit address format is described in Table 24-26.

24.8.2.4.5 Resource Full Message

When more than 256 instructions have run without a branch being taken a program trace resource full message will be generated that indicates the maximum I-CNT value has been reached. The I-CNT field has a maximum width of 8 bits.

Table 25-2. MPC563 Boundary Scan Bit Definition (continued)

BSDL Bit	Cell Type	Pin/Port Name	BSDL Function	Safe Value	Control Cell	Disable Value	Disable Result	Pin Function	Pad Type
133	BC_7	B_AN58_PQA6	bidir	0	132	0	Z	IO	5vsa
134	BC_2	*	controlr	0					
135	BC_7	B_AN59_PQA7	bidir	0	134	0	Z	IO	5vsa
136	BC_2	*	controlr	0					
137	BC_7	ETRIG2_PCS7	bidir	0	136	0	Z	IO	5vfa
138	BC_2	*	controlr	0					
139	BC_7	ETRIG1_PCS6	bidir	0	138	0	Z	IO	5vfa
140	BC_2	*	controlr	0					
141	BC_7	MDA11	bidir	0	140	0	Z	IO	5vsa
142	BC_2	*	controlr	0					
143	BC_7	MDA12	bidir	0	142	0	Z	IO	5vsa
144	BC_2	*	controlr	0					
145	BC_7	MDA13	bidir	0	144	0	Z	IO	5vsa
146	BC_2	*	controlr	0					
147	BC_7	MDA14	bidir	0	146	0	Z	IO	5vsa
148	BC_2	*	controlr	0					
149	BC_7	MDA15	bidir	0	148	0	Z	IO	5vsa
150	BC_2	*	controlr	0					
151	BC_7	MDA27	bidir	0	150	0	Z	IO	5vsa
152	BC_2	*	controlr	0					
153	BC_7	MDA28	bidir	0	152	0	Z	IO	5vsa
154	BC_2	*	controlr	0					
155	BC_7	MDA29	bidir	0	154	0	Z	IO	5vsa
156	BC_2	*	controlr	0					
157	BC_7	MDA30	bidir	0	156	0	Z	IO	5vsa
158	BC_2	*	controlr	0					
159	BC_7	MDA31	bidir	0	158	0	Z	IO	5vsa
160	BC_2	*	controlr	0					
161	BC_7	MPWM0_MDI1	bidir	0	160	0	Z	IO	26v5vs
162	BC_2	*	controlr	0					
163	BC_7	MPWM1_MDO2	bidir	0	162	0	Z	IO	26v5vs
164	BC_2	*	controlr	0					
165	BC_7	MPWM2_PPM_TX1	bidir	0	164	0	Z	IO	26v5vs
166	BC_2	*	controlr	0					
167	BC_7	MPWM3_PPM_RX1	bidir	0	166	0	Z	IO	26v5vs
168	BC_2	*	controlr	0					

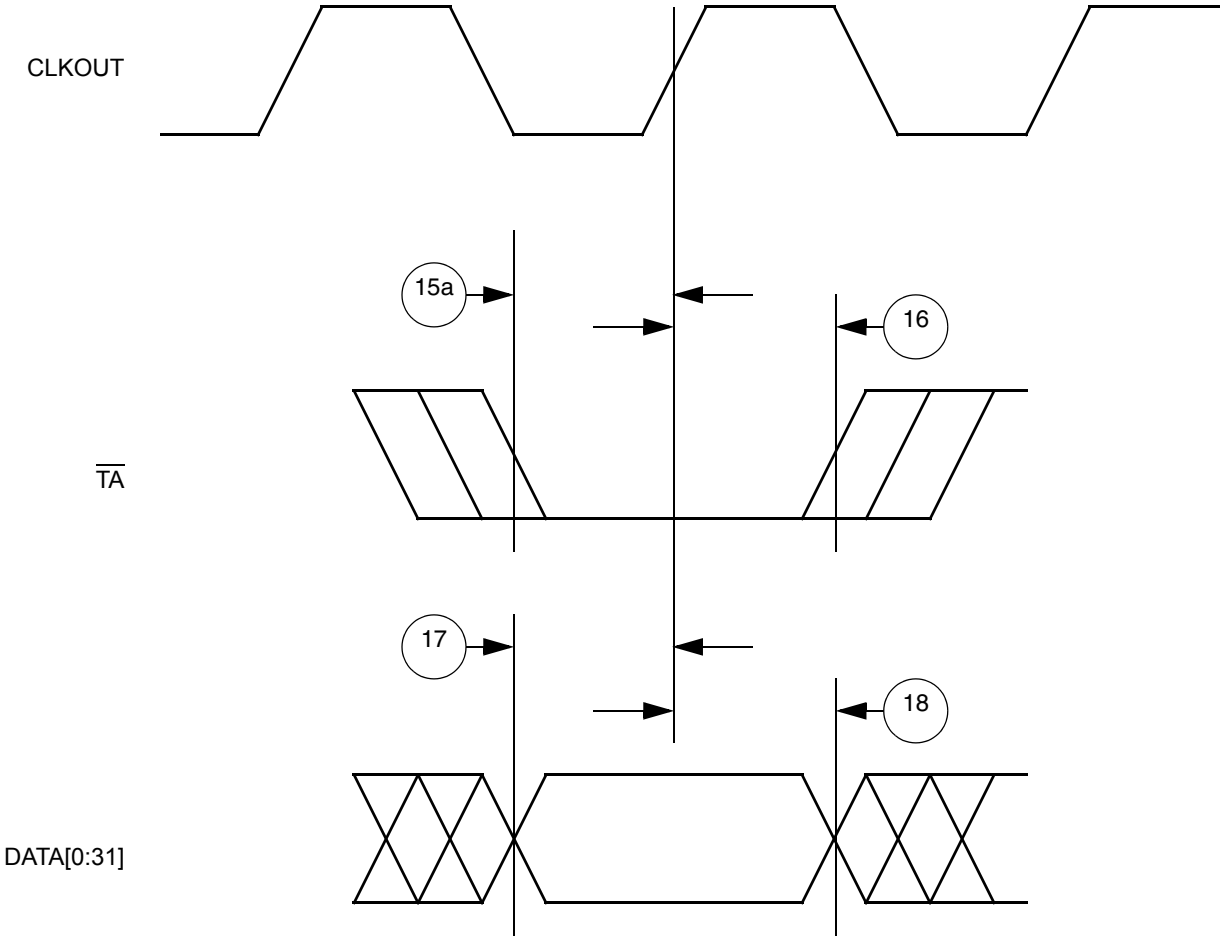

Figure F-15. Input Data Timing In Normal Case