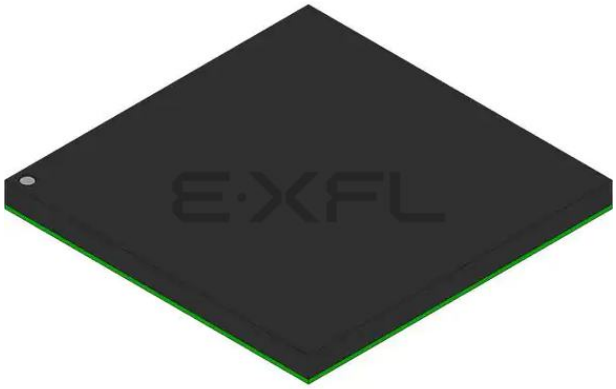


Welcome to E-XFL.COM

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	Active
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	16
Program Memory Size	-
Program Memory Type	ROMless
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/pro/item?MUrl=&PartUrl=mpc562czp40

Figures

Figures

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

Signal Name	No. of Signals	Type	Function after Reset[1]	Description
MPWM1 / MDO2	1	I/O	MPWM1 unless the Nexus (READI) port is enabled, then MDO2. See Section 2.5.	Pulse Width Modulation 1. This signal provides a variable pulse width output signal at a wide range of frequencies.
		O		READI Message Data Out. Message data out (MDO2) is an output signal used for uploading OTM, BTM, DTM, and read/write accesses. External latching of MDO occurs on the rising edge of MCKO. Eight MDO signals are implemented.
MPWM2 / PPM_TX1	1	I/O	MPWM2	Pulse Width Modulation 2. This signal provides a variable pulse width output signal at a wide range of frequencies.
		O		PPMTX1. Transmit data from PPM channel number 1.
MPWM3 / PPM_RX1	1	I/O	MPWM3	Pulse Width Modulation 3. This signal provides a variable pulse width output signal at a wide range of frequencies.
		I		PPMRX1. Receive data to the PPM channel number 1.
MPWM16	1	I/O	MPWM16	Pulse Width Modulation 16. This signal provides a variable pulse width output at a wide range of frequencies. Clock Input: MPWM16 can provide a clock input to modulus clock submodule, MMCSM8
MPWM17 / MDO3	1	I/O	MPWM17 unless the Nexus (READI) port is enabled. See Section 2.5.	Pulse Width Modulation 17. This signal provides variable pulse width outputs at a wide range of frequencies. Load Input: PWM17 can provide a load input to modulus clock submodule, MMCSM8
		O		READI Message Data Out. Message data out (MDO3) is an output signal used for uploading OTM, BTM, DTM, and read/write accesses. External latching of MDO occurs on rising edge of MCKO. Eight MDO signals are implemented.
MPWM[18:19] / MDO[6:7]	2	I/O	MPWM[18:19]	Pulse Width Modulation [18:19]. These signals provide variable pulse width output signals at a wide range of frequencies. Clock and Load Input: • MPWM18 can provide clock inputs to modulus counter submodule MMCSM24 • MPWM19 can provide load inputs to modulus counter submodule MMCSM24
		O		READI Message Data Out. Message data out (MDO[6:7]) are output signals used for uploading OTM, BTM, DTM, and read/write accesses. External latching of MDO occurs on rising edge of MCKO. Eight MDO signals are implemented.

Table 3-5. FPSCR Bit Descriptions (continued)

Bits	Name	Description	
12	VXVC	Floating-point invalid operation exception for invalid compare.	Sticky bit
13	FR	Floating-point fraction rounded. The last floating-point instruction that potentially rounded the intermediate result incremented the fraction.	Not sticky
14	FI	Floating-point fraction inexact. The last floating-point instruction that potentially rounded the intermediate result produced an inexact fraction or a disabled exponent overflow.	Not sticky
[15:19]	FPRF	Floating-point result flags. This field is based on the value placed into the target register even if that value is undefined. Refer to Table 3-6 for specific bit settings. 15 Floating-point result class descriptor (C). Floating-point instructions other than the compare instructions may set this bit with the FPCC bits, to indicate the class of the result. 16-19 Floating-point condition code (FPCC). Floating-point compare instructions always set one of the FPCC bits to one and the other three FPCC bits to zero. Other floating-point instructions may set the FPCC bits with the C bit, to indicate the class of the result. Note that in this case the high-order three bits of the FPCC retain their relational significance indicating that the value is less than, greater than, or equal to zero. 16 Floating-point less than or negative (FL or <) 17 Floating-point greater than or positive (FG or >) 18 Floating-point equal or zero (FE or =) 19 Floating-point unordered or NaN (FU or ?)	Not sticky
20	—	Reserved	—
21	VXSOFT	Floating-point invalid operation exception for software request. This bit can be altered only by the mcrfs, mtfsfi, mtfsf, mtfsb0, or mtfsb1 instructions. The purpose of VXSOFT is to allow software to cause an invalid operation condition for a condition that is not necessarily associated with the execution of a floating-point instruction. For example, it might be set by a program that computes a square root if the source operand is negative.	Sticky bit
22	VXSQRT	Floating-point invalid operation exception for invalid square root. This guarantees that software can simulate fsqrt and frsqrte, and can provide a consistent interface to handle exceptions caused by square root operations.	Sticky bit
23	VXCVI	Floating-point invalid operation exception for invalid integer convert.	Sticky bit
24	VE	Floating-point invalid operation exception enable.	—
25	OE	Floating-point overflow exception enable.	—
26	UE	Floating-point underflow exception enable. This bit should not be used to determine whether denormalization should be performed on floating-point stores.	—
27	ZE	Floating-point zero divide exception enable.	—
28	XE	Floating-point inexact exception enable.	—

The interrupt may be delayed by other higher priority exceptions or if the MSR[EE] bit is cleared when the exception occurs. MSR[EE] is automatically cleared by hardware to disable external interrupts when any exception is taken.

Upon detecting an external interrupt, the processor assigns it to the instruction at the head of the history buffer (after retiring all instructions that are ready to retire).

The enhanced interrupt controller mode is available for interrupt-driven applications on MPC561/MPC563. It allows the single external interrupt exception vector 0x500 to be split into up to 48 different vectors corresponding to 48 interrupt sources to speed up interrupt processing. It also supports a low priority source masking feature in hardware to handle nested interrupts more easily. See Section 6.1.4, "Enhanced Interrupt Controller," and Chapter 4, "Burst Buffer Controller 2 Module."

The register settings for the external interrupt exception are shown in Table 3-26.

Table 3-26. Register Settings following External Interrupt

Register	Bits	Setting Description
Save/Restore Register 0 (SRR0)[1]	All	Set to the effective address of the instruction that the processor would have attempted to execute next if no interrupt conditions were present.
Save/Restore Register 1 (SRR1)	[0:15]	Cleared to 0
	[16:31]	Loaded from bits [16:31] of MSR. In the current implementation, bit 30 of the SRR1 is never cleared, except by loading a zero value from MSR[RI]
Machine State Register (MSR)	IP	No change
	ME	No change
	LE	Set to value of ILE bit prior to the exception
	DCMPEN	This bit is set according to (BBCMCR[EN_COMP] AND BBCMCR[EXC_COMP])
	Other	Cleared to 0

[1] If the exception occurs during an instruction fetch in Decompression On mode, the SRR0 register will contain an address in compressed format.

When an external interrupt is taken, instruction execution resumes at offset 0x00500 from the physical base address indicated by MSR[IP].

3.15.4.6 Alignment Exception (0x00600)

The following conditions cause an alignment exception:

- The operand of a floating-point load or store instruction is not word-aligned.
- The operand of a load or store multiple instruction is not word-aligned.
- The operand of lwarx or stwcx. is not word-aligned.

Alignment exceptions use the SRR0 and SRR1 to save the machine state and the DSISR to determine the source of the exception.

The register settings for alignment exceptions are shown in Table 3-27.

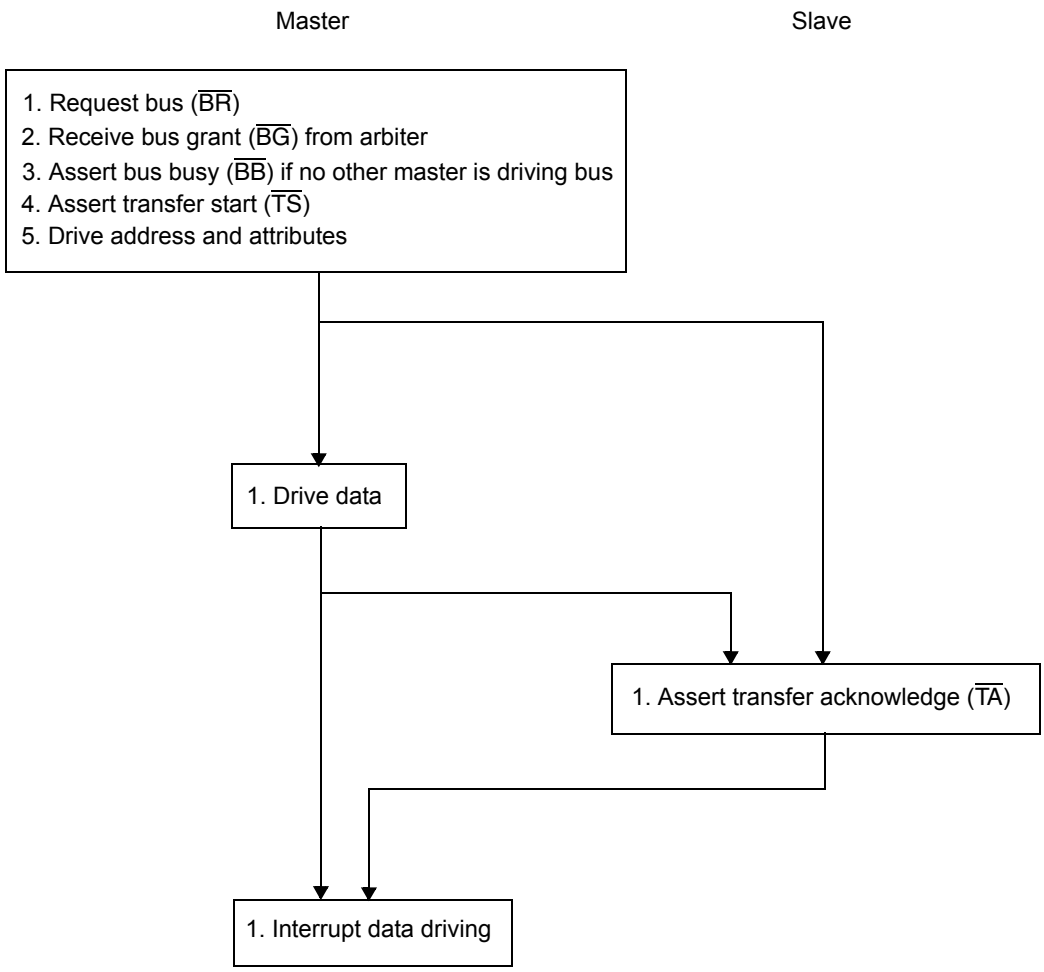


Figure 9-7. Basic Flow Diagram of a Single Beat Write Cycle

a 16-bit boot EPROM and $\overline{CS1}$ is used for a 32-bit SRAM. The $\overline{WE}/\overline{BE}$[0:3] signals are used both to program the EPROM and to enable write access to various bytes in the RAM.

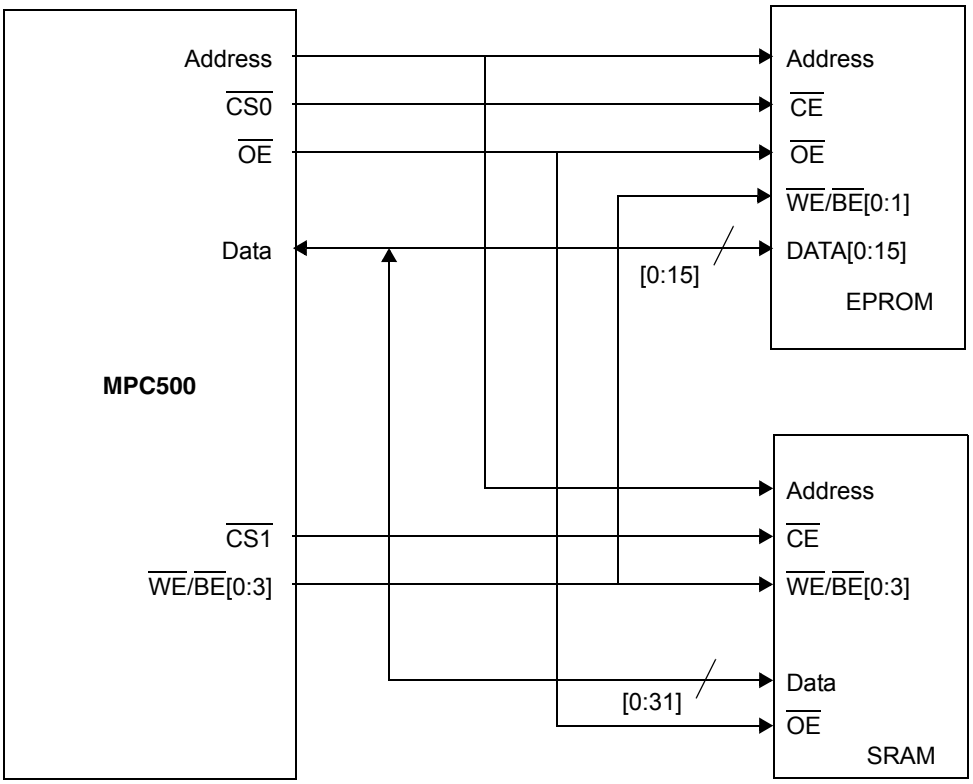


Figure 10-3. MPC561/MPC563 Simple System Configuration

10.2 Memory Controller Architecture

The memory controller consists of a basic machine that handles the memory access cycle: the general-purpose chip-select machine (GPCM).

When any of the internal masters request a new access to external memory, the address of the transfer (with 17 bits having a mask) and the address type (with three bits having a mask) are compared to each one of the valid banks defined in the memory controller. Refer to Figure 10-4.

13.2 Key Features and Quick Reference Diagrams

This section gives an overview of the implementation of the two QADC64E modules on MPC561/MPC563. It can also be used as a quick reference guide while programming the modules.

13.2.1 Features of the QADC64E Legacy Mode Operation

- Internal sample and hold
- Directly supports up to four external multiplexers (for example the MC14051)
- Up to 41 analog input channels using QADC64E external multiplexing
- Programmable input sample time for various source impedances
- Minimum conversion time of 7 µs (with typical QCLK frequency, 2 MHz)
- Two conversion command queues with a total of 64 entries
- Sub-queues possible using pause mechanism
- Queue complete and pause software interrupts available on both queues
- Queue pointers indicate current location for each queue
- Automated queue modes initiated by
  - External edge trigger
  - Periodic/Interval timer, within QADC64E module
  - Software command
  - External gated trigger (Queue 1 only)
- Single-scan or continuous-scan of queues
- 64 result registers in each QADC64E module
- Output readable in three formats
  - Right-justified unsigned
  - Left-justified signed
  - Left-justified unsigned
- Unused analog channels on Port A can be used as digital input/output signals, unused analog channels on Port B can be used as digital input signals.

The analog section includes input signals, an analog multiplexer, and the sample and hold circuits. The analog conversion is performed by the digital-to-analog converter (DAC) resistor-capacitor array and a high-gain comparator.

The digital control section contains queue control logic to sequence the conversion process and interrupt generation logic. Also included are the periodic/interval timer, control and status registers, the conversion command word (CCW) table RAM, and the result table RAM.

The bus interface unit (BIU) allows the QADC64E to operate with the applications software through the IMB3 environment.

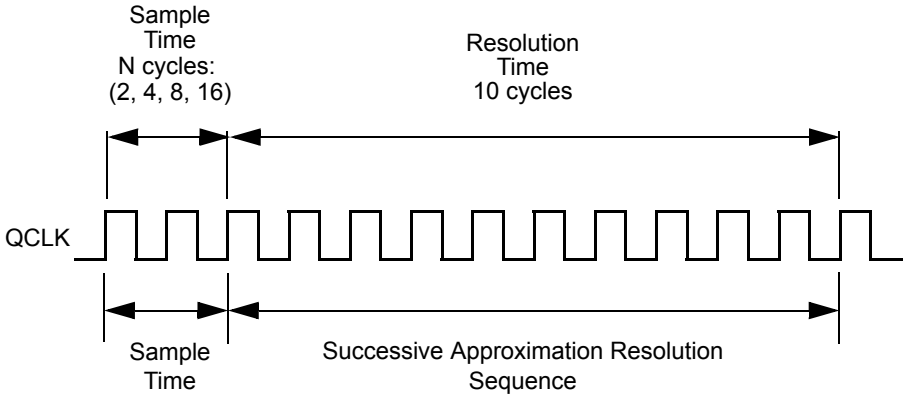


Figure 13-22. Bypass Mode Conversion Timing

13.4.2 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.

13.4.3 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.

13.4.4 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.

Resolution begins with the most significant bit (MSB) and works down to the least significant bit (LSB). The switching sequence is controlled by the comparator and successive-approximation register (SAR) logic.

- Sample capacitor — The sample capacitor is employed to sample and hold the voltage to be converted.

BIU components consist of:

- IMB3 buffers
- Address match and module select logic
- The BIU state machine
- Clock prescaler logic
- Data bus routing logic
- Interface to the internal module data bus

NOTE

Normal accesses from the IMB3 to the QADC64E require two clocks. However, if the CPU tries to access table locations while the QADC64E is accessing them, the QADC64E produces IMB3 wait states. From one to four IMB3 wait states may be inserted by the QADC64E in the process of reading and writing.

13.5.7.2 QADC64E Bus Accessing

The QADC64E supports 8-bit, 16-bit, and 32-bit data transfers, at even and odd addresses. Coherency of results read (ensuring that all results read were taken consecutively in one scan) is not guaranteed. For example, if a read of two consecutive 16-bit locations in a result area is made, the QADC64E could change one 16-bit location in the result area between the bus cycles. There is no holding register for the second 16-bit location. All read and write accesses that require more than one 16-bit access to complete occur as two or more independent bus cycles. Depending on bus master protocol, these accesses could include misaligned and 32-bit accesses.

Figure 13-26 shows the three bus cycles which are implemented by the QADC64E. The following paragraphs describe how the three types of accesses are used, including misaligned 16-bit and 32-bit accesses.

Chapter 14
QADC64E Enhanced Mode Operation

The two queued analog-to-digital converter (QADC) modules on the MPC561/MPC563 devices are 10-bit, unipolar, successive approximation converters. The modules can be configured to operate in one of two modes, legacy mode (for MPC555 compatibility) and enhanced mode. This chapter describes how the module operates in enhanced mode. Refer to Chapter 13, "QADC64E Legacy Mode Operation," for information regarding the QADC64E functionality in legacy mode.

For this revision of the QADC, the name QADC64E implies the enhanced version of the QADC64 module, not just enhanced mode of operation. For simplicity, the names QADC and QADC64E may be used interchangeably throughout this document.

14.1 QADC64E Block Diagram

Figure 14-1 displays the major components of the QADC64E modules on the MPC561/MPC563.

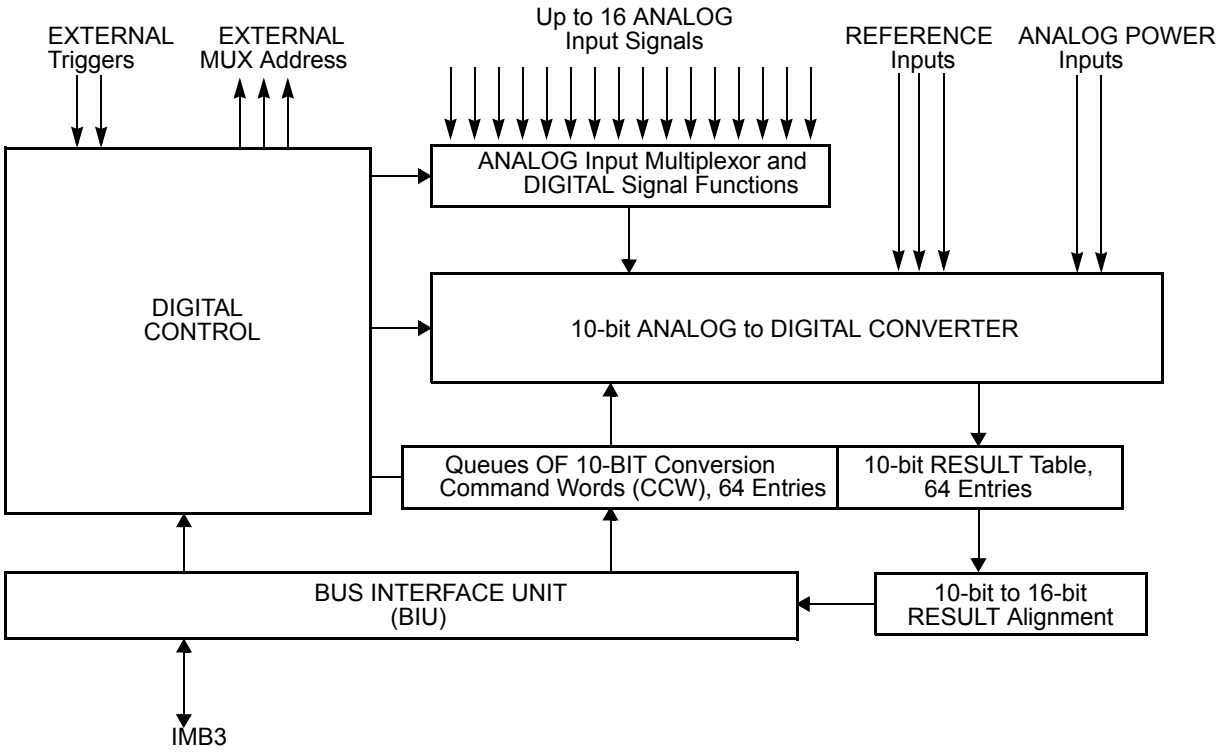


Figure 14-1. QADC64E Block Diagram

Table 14-21. QADC64E_B Multiplexed Channel Assignments and Signal Designations (continued)

Multiplexed Input Signals				Channel Number in CCW CHAN Field	
Port Signal Name	**Analog Signal Name**	**Other Functions / Descriptions**	**Signal Type**	**Binary**	**Decimal**
B_PQB4 B_PQB5 B_PQB6 B_PQB7	AN48 AN47 AN50 AN51	— — — —	Input/Output Input/Output Input/Output Input/Output	0110000 0110001 0110010 0110011	48 49 50 51
B_PQA0 B_PQA1 B_PQA2 B_PQA3	AN52 AN53 AN54 AN55	MA0 MA1 MA2 —	Input/Output Input/Output Input/Output Input/Output	0110100 0110101 0110110 0110111	52 53 54 55
B_PQA4 B_PQA5 B_PQA6 B_PQA7	AN56 AN57 AN58 AN59	— — — —	Input/Output Input/Output Input/Output Input/Output	0111000 0111001 0111010 0111011	56 57 58 59
VRL VRH/ALTREF[1] —	Low Ref High Ref —	— — (VRH – VRL)/2	Input Input —	0111100 0111101 0111110	60 61 62
—	—	End of Queue Code	—	0111111	63
—	RESERVED	—	—	1011000 to 1111111	64 to 127

[1] Whichever is selected in the CCW.

The channel field is programmed for channel 63 to indicate the end of the queue. Channels 60 to 62 are special internal channels. When one of the special channels is selected, the sampling amplifier is not used. The value of V_{RL}, V_{RH}, or $(V_{RH} - V_{RL})/2$ is placed directly onto the converter. Also for the internal special channels, programming any input sample time other than two has no benefit except to lengthen the overall conversion time.

14.3.10 Result Word Table

The result word table is a RAM, 64 words long and 10 bits wide. An entry is written by the QADC64E after completing an analog conversion specified by the corresponding CCW table entry. Software can read or write the result word table, but in normal operation, the software reads the result word table to obtain analog conversions from the QADC64E. Unimplemented bits are read as zeros, and write operations do not have any effect. See Figure 14-15 for a diagram of the result word table

While there is only one result word table, the data can be accessed in three different data formats:

- Right justified in the 16-bit word, with zeros in the higher order unused bits
- Left justified, with the most significant bit inverted to form a sign bit, and zeros in the unused lower order bits
- Left justified, with zeros in the lower order unused bits

17.8 MIOS14 Modulus Counter Submodule (MMCSM)

The MMCSM is a versatile counter submodule capable of performing complex counting and timing functions, including modulus counting, in a wide range of applications. The MMCSM may also be configured as an event counter, allowing the overflow flag to be set after a predefined number of events (internal clocks or external events), or as a time source for other submodules.

NOTE

The MMCSM can also operate as a free running counter by loading the modulus value of zero.

The main components of the MMCSM are an 8-bit prescaler counter, an 8-bit prescaler register, a 16-bit up-counter register, a 16-bit modulus latch register, counter loading and interrupt flag generation logic.

The contents of the modulus latch register is transferred to the counter under the following three conditions:

1. When an overflow occurs
2. When an appropriate transition occurs on the external load signal
3. When the program writes to the counter register. In this case, the value is first written into the modulus register and immediately transferred to the counter.

Software can also write a value to the modulus register for later loading into the counter with one of the two first criteria.

A software control register selects whether the clock input to the counter is one of the prescaler outputs or the corresponding input signal. The polarity of the external input signal is also programmable.

The following sections describe the MMCSM in detail. A block diagram of the MMCSM is shown in Figure 17-10.

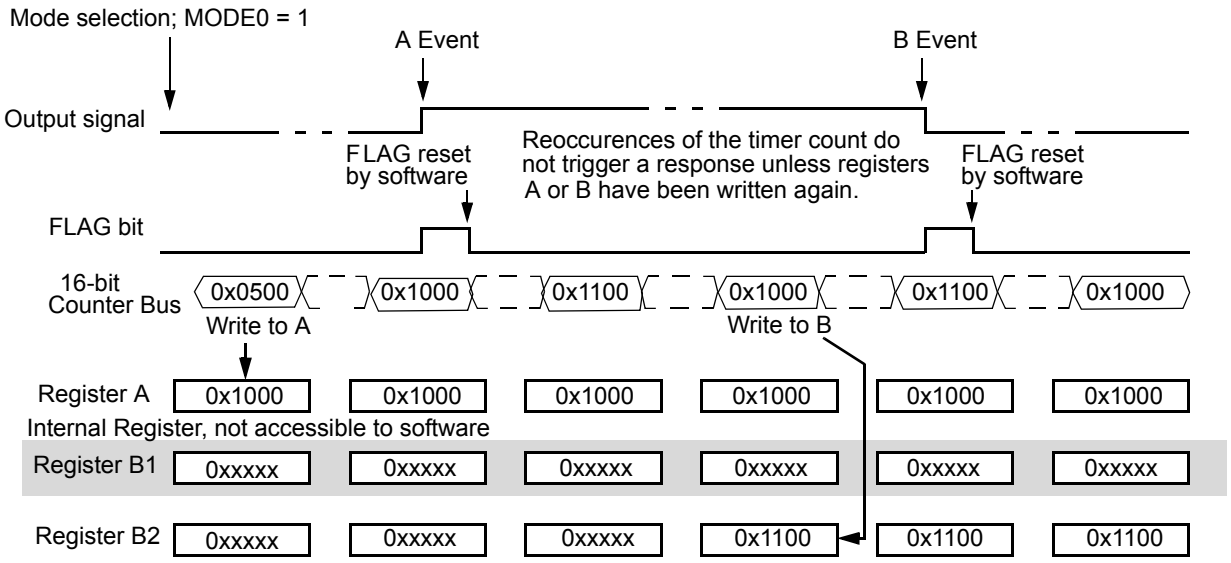


Figure 17-20. Single Shot Output Transition Example

17.9.3.5.3 Output Port Bit Operation

The output port bit operation is selected by leaving both channels disabled, (i.e., by writing to neither register A nor B). The EDPOL bit alone controls the output value. The same result can be achieved by keeping EDPOL at zero and using the FORCA and FORCB bits to obtain the desired output level.

17.9.3.6 Output Pulse Width Modulation (OPWM) Mode

OPWM mode is selected by setting MODE[0:3] to 1xxx. The MODE[1:3] bits allow some of the comparator bits to be masked.

This mode allows pulse width modulated output waveforms to be generated, with eight selectable frequencies. Frequencies are only relevant as such if the counter bus is driven by a counter as a time reference. Both channels (A and B) are used to generate one PWM output signal on the MDASM signal.

Channel B is accessed via register B1. Register B2 is not accessible. Channels A and B define respectively the leading and trailing edges of the PWM output pulse. The value in register B1 is transferred to register B2 each time a match occurs on either channel A or B.

NOTE

A FORCA or FORCB does not cause a transfer from B1 to B2.

The value loaded in register A is compared with the value on the 16-bit counter bus each time the counter bus is updated. When a match on A occurs, the FLAG line is activated and the output flip-flop is set. The value loaded in register B2 is compared with the value on the 16-bit counter bus each time the counter bus is updated. When a match occurs on B, the output flip-flop is reset.

As this register is read only, a write to this register has no other effect than generating a bus error if the bus error option is selected.

	MSB 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	LSB 15
Field	IRP15	IRP14	IRP13	IRP12	IRP1	—		IRP8	IRP7	IRP6	IRP5	IRP4	IRP3	IRP2	IRP1	IRP0
$\overline{\text{SRESET}}$	0000_0000_0000_0000															
Addr	0x30 6C06															

Figure 17-37. Interrupt Request Pending Register (MIOS14RPR0)

Table 17-37. MIOS14PR0 Bit Descriptions

Bits	Name	Description
0:4	IRP15:11	Pending Bits — MDASM pending bits [15:11]
5:6	—	Reserved
7:9	IRP8:6	Pending Bits — MMCSM pending bits [8:6]
10:15	IRP5:0	Pending Bits — PWMSM pending bits [5:0]

17.12.4 MIRSM1 Interrupt Registers

17.12.4.1 Interrupt Status Register (MIOS14SR1)

This register contains the flag bits that are raised when the submodules generate an interrupt. Each bit corresponds to a given submodule.

	MSB 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	LSB 15
Field	FLG 31	FLG 30	FLG 29	FLG 28	FLG 27	—		FLG 24	FLG 23	FLG 22	FLG 21	FLG 20	FLG 19	FLG 18	FLG 17	FLG 16
$\overline{\text{SRESET}}$	0000_0000_0000_0000															
Addr	0x30 6C40															

Figure 17-38. Interrupt Status Register (MIOS14SR1)

Table 17-38. MIOS14SR1 Bit Descriptions

Bits	Name	Description
0:4	FLG31:27	Flag Bits — MDASM flag bits [31:27]
5:6	—	Reserved
7:9	FLGL24:22	Flag Bits— MMCSM flag bits [24:22]
10:15	FLG21:16	Flag Bits — PWMSM flag bits [21:16]

Table B-1. SPR (Special Purpose Registers) (continued)

Address	Access	Symbol	Register	Size	Reset
SPR 529	S	EIBADR	External Interrupt Relocation Table Base Address Register See Table 4-9 for bit descriptions.	32	—
SPR 536	S	L2U_GRA	L2U Global Region Attribute Register See Table 11-10 for bit descriptions.	32	—
SPR 560	S	BBCMCR	BBC Module Configuration Register See Table 4-4 for bit descriptions.	32	H
SPR 568	S	L2U_MCR	L2U Module Configuration Register See Table 11-7 for bit descriptions.	32	—
SPR 630	S	DPDR	Development Port Data Register See Section 23.4.6 for bit descriptions.	32	—
SPR 638	S	IMMR	Internal Memory Mapping Register See Table 6-12 for bit descriptions.	32	H
SPR 784 – 787	S	MI_RBAx	MI Region x Base Address Register See Table 4-5 for bit descriptions.	32	—
SPR 792 – 795	S	L2U_RBAx	L2U Region x Base Address Register See Table 11-8 for bit descriptions.	32	—
SPR 816 – 819	S	MI_RAx	MI Region x Attribute Register See Table 4-6 for bit descriptions.	32	—
SPR 824 – 827	S	L2U_RAx	L2U Region x Attribute Register See Table 11-9 for bit descriptions.	32	—

Table B-2. UC3F Flash Array

Address	Access	Symbol	Register	Size	Reset
0x00 0000 — 0x07 FFFF	U,S	UC3F	UC3F Flash Array	32	—

Table B-3. DECRAM SRAM Array

Address	Access	Symbol	Register	Size	Reset
0x2F 8000 — 0x2F 87FF	U,S	DECRAM	DECRAM SRAM	32	—

Table B-4. BBC (Burst Buffer Controller Module)

Address	Access	Symbol	Register	Size	Reset
0x2F A000	S (read only)[1]	DCCR0	Decompressor Class Configuration Register See Table A-3 for bit descriptions.	32	—
0x2F A004	S	DCCR1	Decompressor Class Configuration Register See Table A-3 for bit descriptions.	32	—

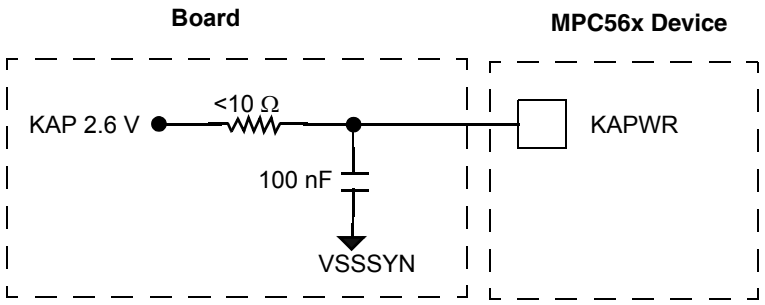
- EXTAL pad capacitance is C_{PAD} = ~7 pF

C_{SOCKET} is socket and board trace capacitance

- Socket capacitance C <=1pF
- Board trace capacitance C <=1pF. This should be low since the crystal must be located very close to the chip.

C.2.1 KAPWR Filtering

The KAPWR signal is the MPC561/MPC563 keep-alive power. KAPWR is used for the crystal oscillator circuit, and should be isolated from the noisy supplies. It is recommended that an RC filter be used on KAPWR, or bypass capacitors that are located as close as possible to the part.



Note: A filter cut off frequency of 500Hz is recommended, however this will result in a capacitor size of 33uF using a 10 Ohm resistor. This may be too expensive or large for the system. In this case the filter shown with cut- off frequency of 160kHz will suffice.

Figure C-4. RC Filter Example

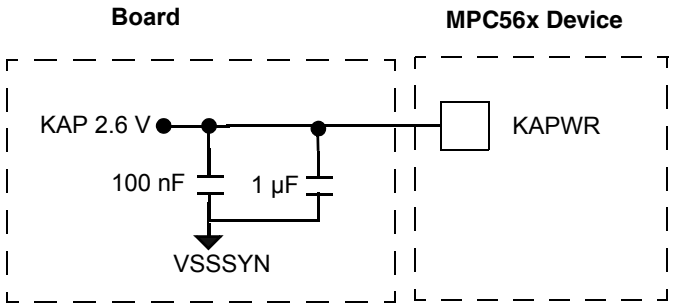


Figure C-5. Bypass Capacitors Example (Alternative)

C.2.2 PLL External Components

VDDSYN and VSSSYN are the PLL dedicated power supplies. These supplies must be used only for the PLL and isolated from all other noisy signals in the board. VDDSYN could be isolated with RC filter (see Figure C-6), or LC filter. The maximum noise allowed on VDDSYN, and VSSSYN is 50 mV with typical cut-off frequency of 500 Hz.

the SIOP clock channel + actual SIOP service time (= Td) and ensure that the baud rate is chosen such that HALF_PERIOD - Td is not less that the minimum setup time of the receiving device. A transmitting device must also hold data valid for a minimum time of Td after the clock.

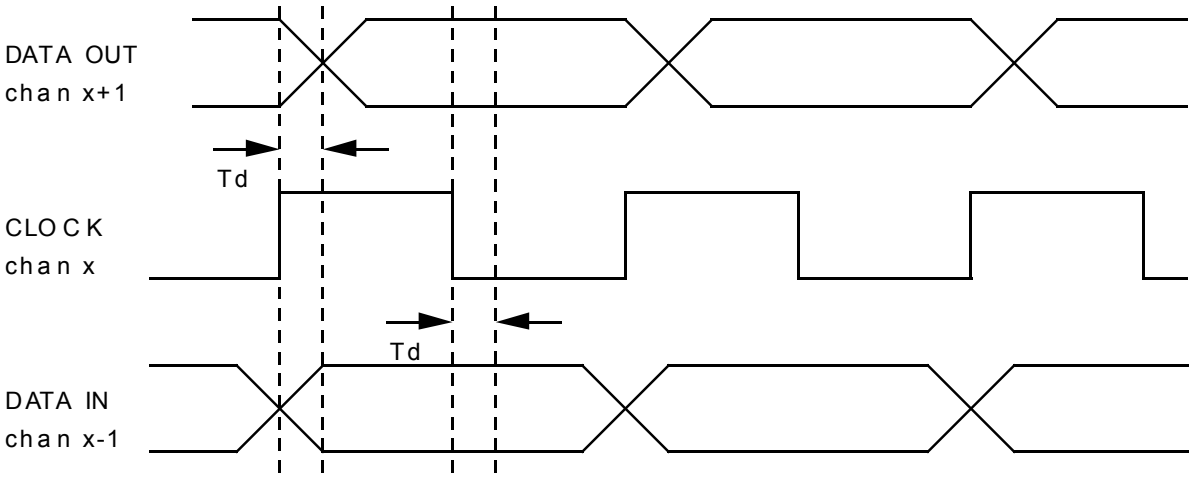


Figure D-33. SIOP Function Data Transition Example

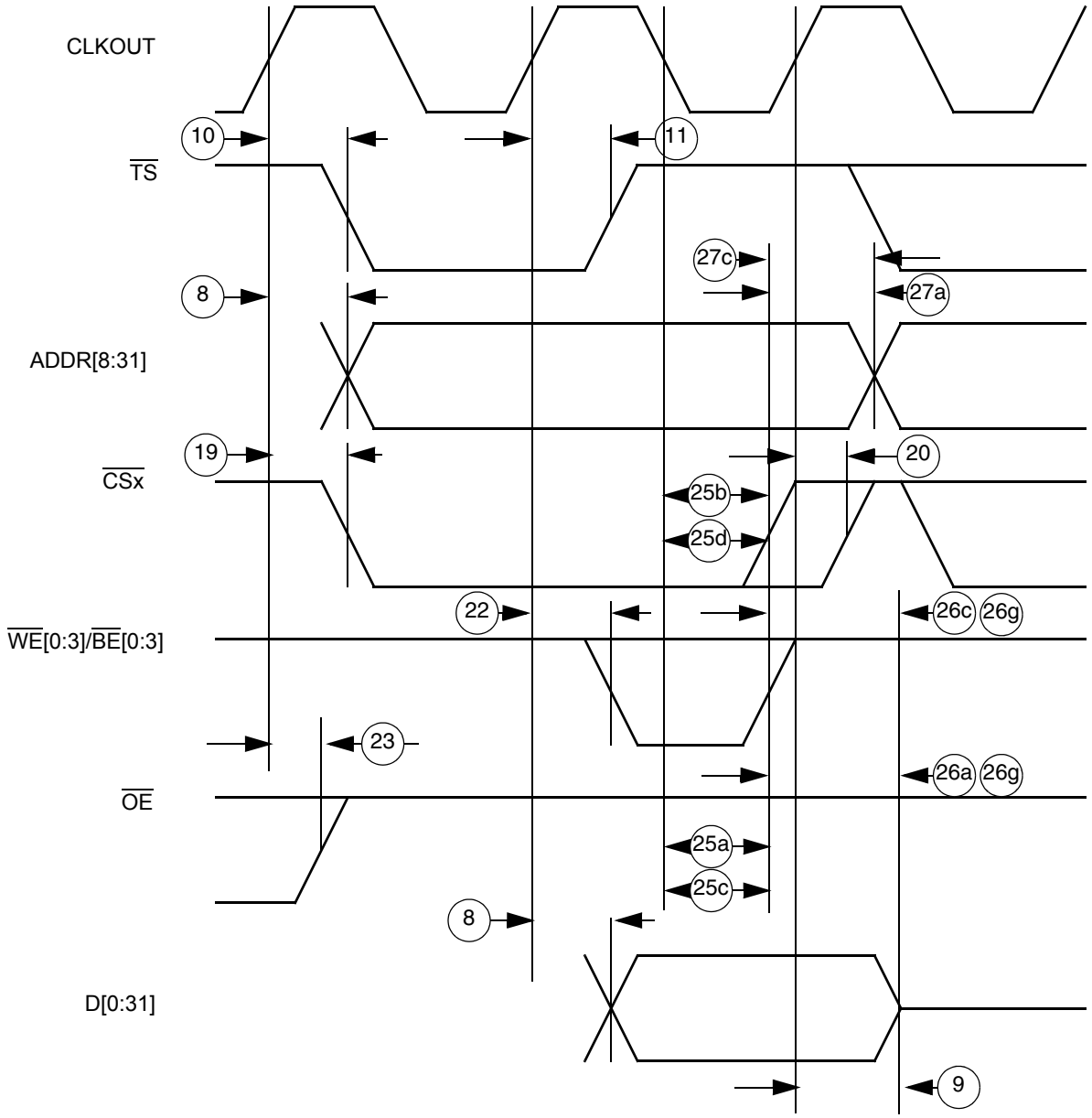


Figure G-22. External Bus Write Timing (GPCM Controlled – TRLX = '0', CSNT = '1')

G.19 TouCAN Electrical Characteristics

Table G-21. TouCAN Timing[1]

Note: ($T_A = T_L$ to T_H)

Num	Rating	Symbol	Min	Max	Unit
127	CNTX0 (Delay from ICLOCK)	t_{CNTX0}		19	ns
128	CNRX0 (Set-Up to ICLOCK rise)	t_{CNRX0}		0	ns
129	Rise Time Input Output – 50 pF load, SLRC1 bit of PDMCR = “0” 200 pF load, SLRC1 bit of PDMCR = “0” 50 pF, SLRC1 bit of PDMCR = “1”	 t_{RI} t_{RO}		 1 50 100 25	 µs ns ns ns
130	Fall Time Input Output– 50 pF load, SLRC1 bit of PDMCR = “0” 200 pF load, SLRC1 bit of PDMCR = “0” 50 pF, SLRC1 bit of PDMCR = “1”	 t_{FI} t_{FO}		 1 50 100 25	 µs ns ns ns
	Serial Pins (Maximum frequency)	t_F	1	—	MHz

[1] AC timing is shown is tested to the 3-V levels outlined in <XrefBlue>Table G-4 on page G-7.

G.20 PPM Timing Characteristics

Table G-22. PPM Timing

Note: ($T_A = T_L$ to T_H, 50 pF Load on all Pins)

Num	Rating	Symbol	Min	Max	Unit
131	Operating Frequency[1]	f_{OP}	$f_{SYS}/256$	$f_{SYS}/2^2$	Hz
132	Cycle Time	t_{CYC}	2*TC	256*TC[3]	ns
133	PPM Clock (PPM_TCLK) high or low time	t_{SW}	$(t_{cyc}/2) - (t_{RO} + t_{FI})$		ns
134	Sequential Transfer Delay	t_{TD}	9*tcyc	17*tcyc	ns
135	Data Setup Time (Inputs)	t_{SU}	30	—	ns
136	Data Hold Time (Inputs)	t_{HI}	0	—	ns
137	Data Valid (After PPM_TCLK Edge)	t_V	—	5	ns
138	Data Hold Time (Outputs)	t_{HO}	TC/2	—	ns