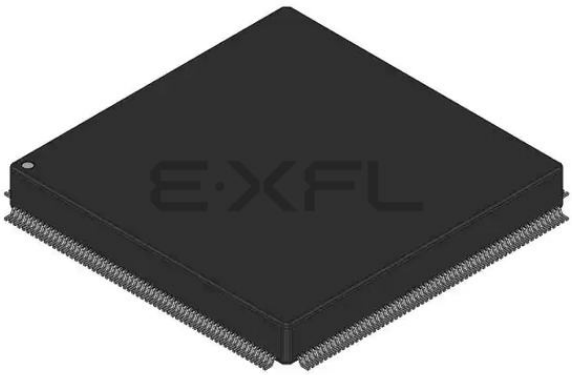


Welcome to E-XFL.COM

Understanding Embedded - Microprocessors

Embedded microprocessors are specialized computing chips designed to perform specific tasks within an embedded system. Unlike general-purpose microprocessors found in personal computers, embedded microprocessors are tailored for dedicated functions within larger systems, offering optimized performance, efficiency, and reliability. These microprocessors are integral to the operation of countless electronic devices, providing the computational power necessary for controlling processes, handling data, and managing communications.

Applications of Embedded - Microprocessors

Embedded microprocessors are utilized across a broad spectrum of applications, making them indispensable in

Details	
Product Status	Active
Core Processor	CPU32+
Number of Cores/Bus Width	1 Core, 32-Bit
Speed	33MHz
Co-Processors/DSP	Communications; CPM
RAM Controllers	DRAM
Graphics Acceleration	No
Display & Interface Controllers	-
Ethernet	10Mbps (1)
SATA	-
USB	-
Voltage - I/O	5V
Operating Temperature	0°C ~ 70°C (TA)
Security Features	-
Package / Case	240-BFQFP
Supplier Device Package	240-FQFP (32x32)
Purchase URL	https://www.e-xfl.com/pro/item?MUrl=&PartUrl=mc68360ai33l

Table 3-4. QUICC CPM Registers Memory Map

REGB + 692	SMCMR2	16	SMC2 Mode Register	0000		SMC2
REGB + 696	SMCE2	8	SMC2 or PIP Event Register	00		
REGB + 69a	SMCM2	8	SMC2 Mask Register	00		
REGB + 69C			Reserved			
REGB + 6A0	SPMODE	16	SPI Mode Register	0000	H	SPI
REGB + 6A6	SPIE	8	SPI Event Register	00		
REGB + 6AA	SPIM	8	SPI Mask Register	00		
REGB + 6AD	SPCOM	8	SPI Command Register	00		
REGB + 6B2	PIPC	16	PIP Configuration Register	0000	H	PIP
REGB + 6B6	PTPR	16	PIP Timing Parameters Register	0000		
REGB + 6B8	PBDIR	18	Port B Data Direction Register	xxx0 0000	H	
REGB + 6BC	PBPAR	18	Port B Pin Assignment Register	xxx0 0000	H	
REGB + 6C2	PBODR	16	Port B Open Drain Register	0000	H	
REGB + 6C4	PBDAT	18	Port B Data Register	xxxX XXXX		
REGB + 6c8 to REGB + 6df			Reserved			
REGB + 6E0	SIMODE	32	SI Mode Register	0000 0000	H	SI
REGB + 6E4	SIGMR	8	SI Global Mode Register	00	H	
REGB + 6E6	SISTR	8	SI Status Register	00	H	
REGB + 6E7	SICMR	8	SI Command Register	00		
REGB + 6E8		32	Reserved			
REGB+ 6EC	SICR	32	SI Clock Route	0000 0000	H	
REGB + 6F2	SIRP	32	SI RAM Pointers	0000 0000		
REGB + 6F6 to REGB + 6FF	RES		Reserved			
REGB + 700 to REGB + 7ff	SIRAM	256 Bytes	SI Routing RAM	XXXX		

Notes:
1.Reset value field.
2.H=Effected only upon $\overline{\text{RESETH}}$ assertion
3.S=Effected only upon $\overline{\text{RESETS}}$ assertion
4.Blank field = Effected by both $\overline{\text{RESETH}}$ or $\overline{\text{RESETS}}$ assertion.

4.6.8 Show Cycles

The QUICC can perform data transfers with its internal modules without using the external bus, but when debugging, it is desirable to have address and data information appear on the external bus. These external bus cycles, called show cycles, are distinguished by the fact that $\overline{AS}$ is not asserted externally. $\overline{DS}$ is used to signal address strobe timing in show cycles.

After reset, show cycles are disabled and must be enabled by writing to the SHEN bits in the module configuration register. When show cycles are disabled, the address bus, function codes, size, and read/write signals continue to reflect internal bus activity. However, $\overline{AS}$ and $\overline{DS}$ are not asserted externally, and the external data bus remains in a high impedance state. When show cycles are enabled, $\overline{DS}$ indicates address strobe timing and the external data bus contains data. The following paragraphs are a state-by-state description of show cycles, and Figure 4-45 illustrates a show cycle timing diagram. Refer to Section 10 Electrical Characteristics for specific timing information.

State 0 – During state 0, the address and function codes become valid, R/$\overline{W}$ is driven to indicate a show read or write cycle, and the size pins indicate the number of bytes to transfer. During a read, the addressed peripheral is driving the data bus, and the user must take care to avoid bus conflicts.

State 41 – One-half clock cycle later, $\overline{DS}$ (rather than $\overline{AS}$) is asserted to indicate that address information is valid.

State 42– No action occurs in state 42. The bus controller remains in state 42 (wait states will be inserted) until the internal read cycle is complete.

State 43– When $\overline{DS}$ is negated, show data is valid on the next falling edge of the system clock. The external data bus drivers are enabled so that data becomes valid on the external bus as soon as it is available on the internal bus.

State 0 – The address, function codes, read/write, and size pins change to begin the next cycle. Data from the preceding cycle is valid through state 0.

≠	Not equal to
>	Greater than
≥	Greater than or equal to
<	Less than
≤	Less than or equal to
Λ	Logical AND
V	Logical OR
⊕	Logical exclusive OR
~	Invert; operand is logically complemented
BCD	Binary-coded decimal, indicated by subscript Example: $Source_{10}$ is a BCD source operand.
LSW	Least significant word
MSW	Most significant word
{R/W}	Read/write indicator

In a description of an operation, a destination operand is placed to the right of source operands and is indicated by an arrow (⇒).

5.3.3 Instruction Summary

The instructions form a set of tools to perform the following operations:

Data Movement	Bit Manipulation
Integer Arithmetic	Binary-Coded Decimal Arithmetic
Logic	Program Control
Shift and Rotate	System Control

The complete range of instruction capabilities combined with the addressing modes described previously provide flexibility for program development. All CPU32+ instructions are summarized in Table 5-2.

The UNLK instruction removes a stack frame from the end of the list by loading an address into the SP and pulling the value at that address from the stack. When the instruction operand is the address of the link address at the bottom of a stack frame, the effect is to remove the stack frame from both the stack and the linked list.

5.3.6 Pipeline Synchronization with the NOP Instruction

Although the no operation (NOP) instruction performs no visible operation, it does force synchronization of the instruction pipeline, since all previous instructions must complete execution before the NOP begins.

5.4 PROCESSING STATES

This section describes the processing states of the CPU32+. It includes a functional description of the bits in the supervisor portion of the SR and an overview of actions taken by the processor in response to exception conditions.

5.4.1 State Transitions

The processor is always in one of four processing states: normal, background, exception, or halted.

When the processor fetches instructions and operands or executes instructions, it is in the normal processing state. The stopped condition, which the processor enters when a STOP or LPSTOP instruction is executed, is a variation of the normal state in which no further bus cycles are generated.

Background state is an alternate operational mode used for system debugging. Refer to 5.6 Development Support for more information.

Exception processing refers specifically to the transition from normal processing of a program to normal processing of system routines, interrupt routines, and other exception handlers. Exception processing includes the stack operations, the exception vector fetch, and the filling of the instruction pipeline caused by an exception. Exception processing ends when execution of an exception handler routine begins. Refer to 5.5 Exception Processing for comprehensive information.

A catastrophic system failure occurs if the processor detects a bus error or generates an address error while in the exception processing state. This type of failure halts the processor. For example, if a bus error occurs during exception processing caused by another bus error, the CPU32+ assumes that the system is not operational and halts.

The halted condition should not be confused with the stopped condition. After the processor executes a STOP or LPSTOP instruction, execution of instructions can resume when a trace, interrupt, or reset exception occurs.

5.4.2 Privilege Levels

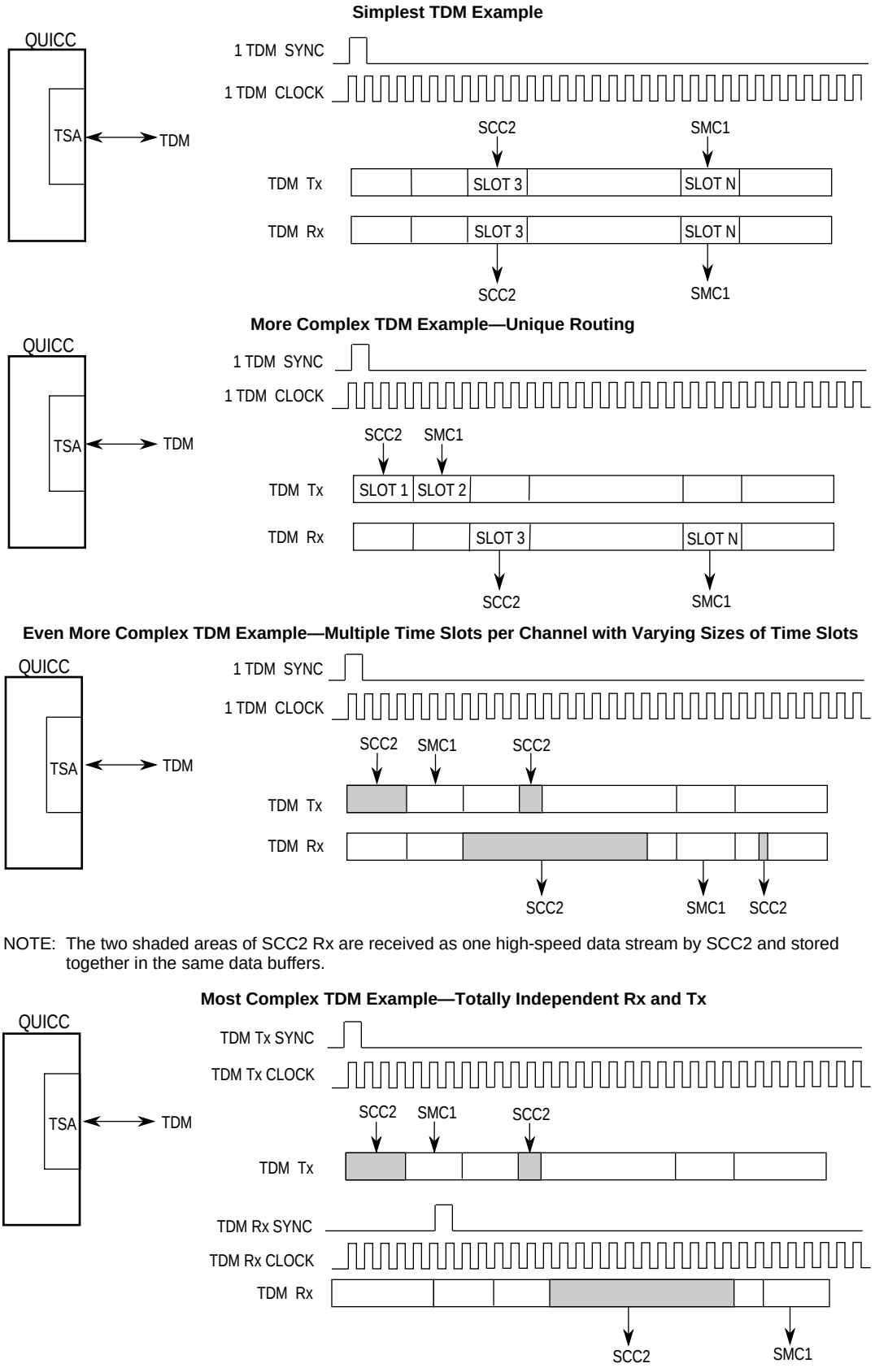
To protect system resources, the processor can operate with either of two levels of access—user or supervisor. Supervisor level is more privileged than user level. All instructions are

5.7.2.14 SAVE AND RESTORE OPERATIONS. The save and restore operations table indicates the number of clock periods needed for the processor to perform the specified state save or return from exception. Complete execution times and stack length are given. No additional tables are needed to calculate total effective execution time for these instructions. The total number of clock cycles is outside the parentheses. The numbers inside parentheses (r/p/w) are included in the total clock cycle number. All timing data assumes two-clock reads and writes.

Instruction	Head	Tail	Cycles
BERR on instruction	0	–2	<58(2/2/12)
BERR on exception	0	–2	48(2/2/12)
RTE (four-word frame)	1	–2	24(4/2/0)
RTE (six-word frame)	1	–2	26(4/2/0)
RTE (BERR on instruction)	1	–2	50(12/12/Y)
RTE (BERR on four-word frame)	1	–2	66(10/2/4)
RTE (BERR on six-word frame)	1	–2	70(12/2/6)

Y = If a bus error occurred during a write cycle, the cycle is rerun by the RTE.

< = Maximum time is indicated (certain data or mode combinations execute faster).
Timing is calculated with the CPU32+ in 16-bit mode.



NOTE: The two shaded areas of SCC2 Rx are received as one high-speed data stream by SCC2 and stored together in the same data buffers.

Figure 7-20. Various Configurations of a Single TDM Channel

Although TBPTR need never be written by the user in most applications, it may be modified by the user when the transmitter is disabled or when the user is sure that no transmit buffer is currently in use (e.g., after STOP TRANSMIT command is issued, or after a GRACEFUL STOP TRANSMIT command is issued, and the frame completes its transmission.)

7.10.7.6 OTHER GENERAL PARAMETERS. Additional parameters are listed in Table 7-5. These parameters do not need to be accessed by the user in normal operation, and are listed only because they may provide helpful information for experienced users and for debugging.

The Rx and Tx internal data pointers are updated by the SDMA channels to show the next address in the buffer to be accessed.

The Tx internal byte count is a down-count value that is initialized with the Tx BD data length and decremented with every byte read by the SDMA channels. The Rx internal byte count is a down-count value that is initialized with the MRBLR value and decremented with every byte written by the SDMA channels.

NOTE

To extract data from a partially full receive buffer, the CLOSE Rx BD command may be used.

The Rx internal state, Tx internal state, Rx temp, Tx temp, and reserved areas are for RISC use only.

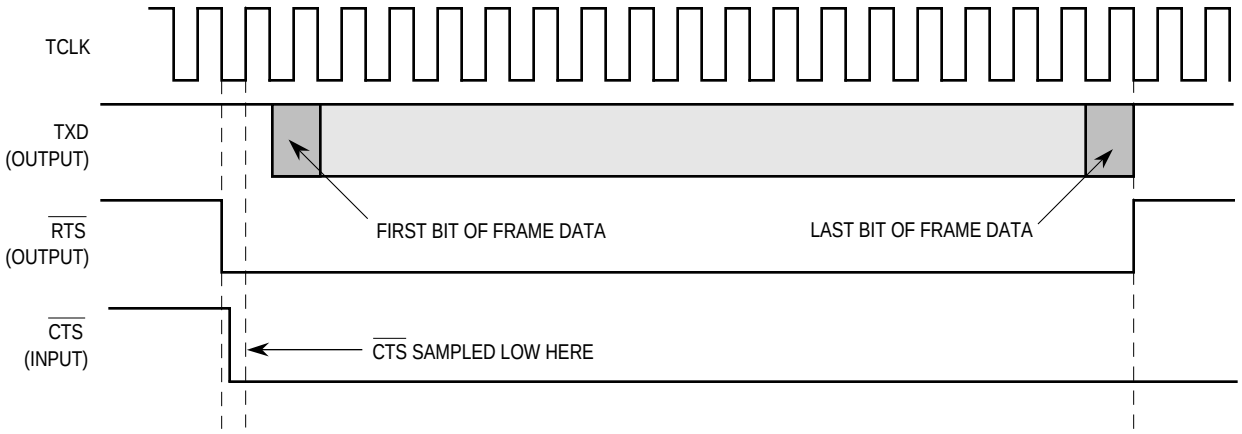
7.10.8 Interrupts from the SCCs

Interrupt handling for each of the SCC channels is configured on a global (per channel) basis in the CPM interrupt pending register, CPM interrupt mask register, and CPM in service register. Within each of these registers, one bit is used to either mask, enable, or report the presence of an interrupt in an SCC channel. The interrupt priority between the four SCCs is programmable in the CP interrupt configuration register. An SCC interrupt may be caused by a number of events. To allow interrupt handling for these (SCC-specific) events, further event registers are provided within the SCCs.

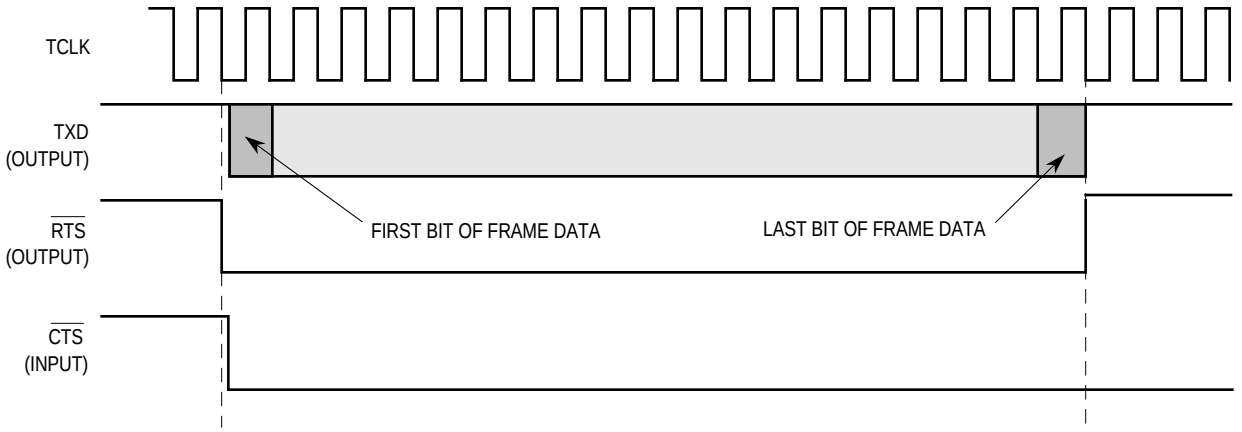
A number of events can cause the SCC to interrupt the processor. The events differ slightly according to the protocol selected. For a detailed description of the events see the specific protocol paragraphs. These events are handled independently for each channel by the SCC event register and the SCC mask register.

Events that can cause interrupts due to the $\overline{\text{CTS}}$ and $\overline{\text{CD}}$ modem lines are described in 7.14.9 Port C Pin Functions.

7.10.8.1 SCC EVENT REGISTER (SCCE). The 16-bit SCC event register is used to report events recognized by any of the SCCs. On recognition of an event, the SCC will set its corresponding bit in the SCC event register (regardless of the corresponding mask bit). The SCC event register appears to the user as a memory-mapped register and may be read at any time. A bit is cleared by writing a one (writing a zero does not affect a bit's value). More than one bit may be cleared at a time. This register is cleared at reset.



NOTE: CTSS = 0 in GSMR. CTSP is a don't care.



NOTE: CTSS = 1 in GSMR. CTSP is a don't care.

Figure 7-40. Output Delays from $\overline{\text{CTS}}$ Asserted for Synchronous Protocols

If the $\overline{\text{CTS}}$ pin is programmed to envelope the data, the $\overline{\text{CTS}}$ pin must remain asserted during frame transmission, or a CTS lost error occurs (see Figure 7-41). The negation of the $\overline{\text{CTS}}$ pin forces the $\overline{\text{RTS}}$ pin high, forcing the transmit data to the idle state. If the CTSS bit in the GSMR is zero, the $\overline{\text{CTS}}$ pin must be sampled by the SCC before a CTS lost is recognized. Otherwise, the negation of $\overline{\text{CTS}}$ immediately causes the CTS lost condition.

7.10.17.13 SCC STATUS REGISTER (SCCS). The SCCS is an 8-bit read-only register that allows the user to monitor real-time status conditions on the RXD line. The real-time status of the $\overline{CTS}$ and $\overline{CD}$ pins are part of the port C parallel I/O.

7	6	5	4	3	2	1	0
—	—	—	—	—	FG	CS	ID

Bits 7–3—Reserved

FG—Flags

While FG is cleared, the most recently received 8 bits are examined every bit time to see if a flag is present. FG is set as soon as an HDLC flag ($7E) is received on the line. Once FG is set, it will remain set at least 8 bit times, at which time the next 8 received bits are examined. If another flag occurs, then FG remains set for at least another eight bits; otherwise, FG is cleared and the search begins again.

The examination of the line is made after the data has been decoded by the DPLL.

0 = HDLC flags are not currently being received.
1 = HDLC flags are currently being received.

CS—Carrier Sense (DPLL)

This bit shows the real-time carrier sense of the line as determined by the DPLL, if it is used.

0 = The DPLL does not sense a carrier.
1 = The DPLL does sense a carrier.

ID—Idle Status

ID is set when the RXD pin is a logic one for 15 or more consecutive bit times; it is cleared after a single logic zero is received.

0 = The line is not currently idle.
1 = The line is currently idle.

7.10.17.14 SCC HDLC EXAMPLE #1. The following list is an initialization sequence for an SCC HDLC channel assuming an external clock is provided. SCC4 is used. The HDLC controller is configured with the $\overline{RTS4}$, $\overline{CTS4}$, and $\overline{CD4}$ pins active. The CLK7 pin is used for both the HDLC receiver and transmitter.

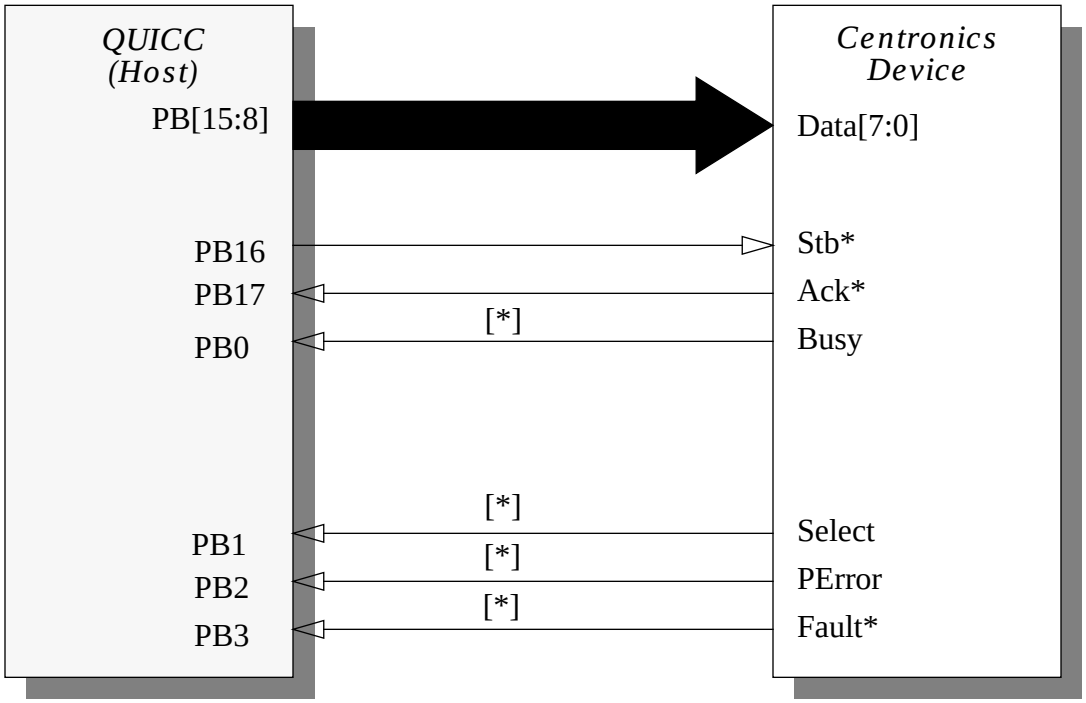
1. The SDCR (SDMA Configuration Register) should be initialized to $0740, rather than being left at its default value of $0000.
2. Configure the port A pins to enable the TXD4 and RXD4 pins. Write PAPAR bits 6 and 7 with ones. Write PADIR bits 6 and 7 with zeros. Write PAODR bits 6 and 7 with zeros.
3. Configure the port C pins to enable $\overline{RTS4}$, $\overline{CTS4}$, and $\overline{CD4}$. Write PCPAR bit 3 with one, and bits 10 and 11 with zeros. Write PCDIR bits 3, 10 and 11 with zeros. Write PCSO bits 10 and 11 with ones.
4. Configure port A to enable the CLK7 pin. Write PAPAR bit 14 with a one. Write PADIR bit 14 with a zero.

data), data will begin to be transmitted on the next falling edge of SMCLK after one character of ones is transmitted. If the transmit FIFO is loaded at some later time, the data will begin transmission after some multiple number of all-ones characters is transmitted. The transmitter will never lose synchronization again, regardless of the state of $\overline{\text{SMSYNx}}$, until the TEN bit is cleared by the user or the ENTER HUNT MODE command is issued.

If both the REN and TEN bits are set in SMCMR, the first falling edge of the $\overline{\text{SMSYNx}}$ pin causes both the transmitter and receiver to achieve synchronization. To re-synchronize the transmitter, the SMC transmitter may be disabled and reenabled, and the $\overline{\text{SMSYNx}}$ pin can be used again to re-synchronize just the transmitter. See 7.11.5 Disabling the SMCs on the Fly for a description of how to safely disable and reenable the SMC (simply clearing TEN and setting TEN may not be sufficient). The receiver may be re-synchronized in a similar fashion.

7.11.10.7 USING THE TSA FOR SYNCHRONIZATION. The TSA offers a method to synchronize the SMC channel internally without using the $\overline{\text{SMSYNx}}$ pin. This behavior is similar to that of the $\overline{\text{SMSYNx}}$ pin, except that the synchronization event is not the falling edge of the $\overline{\text{SMSYNx}}$ pin, but rather the first time slot for this SMC receiver/transmitter following the frame sync indication. See 7.8 Serial Interface with Time Slot Assigner for further information on configuring time slots for the SMCs and SCCs.

The TSA allows the SMC receiver and transmitter to be enabled simultaneously, yet synchronized separately, a capability not provided by the $\overline{\text{SMSYNx}}$ pin. See Figure 7-79 for an example of synchronization using the TSA.



[*] - optional **Figure 7-95. Centronics Transmitter Configuration**

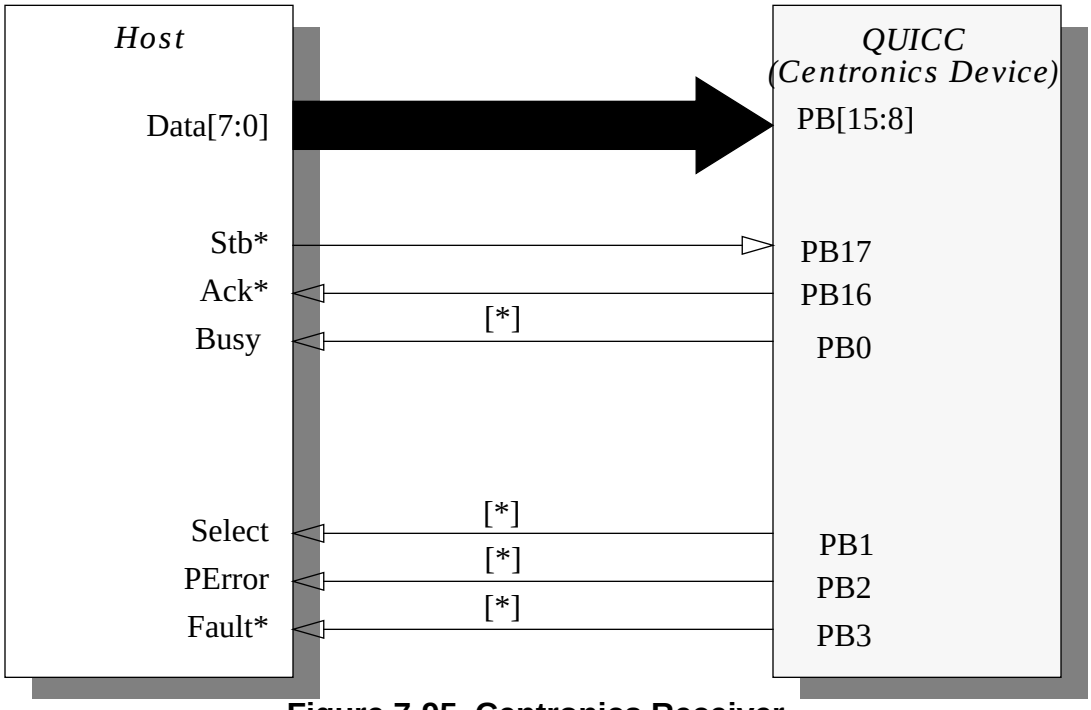


Figure 7-95. Centronics Receiver

7.13.8.1 CENTRONICS CONTROLLER KEY FEATURES. Super-set of the Centronics standard

- 8-bit or 16-Bit Data Transfer

For each DDx bit, the definition is as follows:

0 = General-purpose I/O. The peripheral functions of the pin are not used.
1 = Dedicated peripheral function. The pin is used by the internal module. The on-chip peripheral function to which it is dedicated may be determined by other bits such as these in the PBDIR.

7.14.8 Port B Example

PB0 can be configured as a general-purpose I/O pin or as an open-drain pin. It may also be the receiver reject pin for the SCC1 Ethernet CAM interface ($\overline{\text{RRJCT1}}$) or the SPI select input ($\overline{\text{SPISEL}}$). If the PB0 pin is not configured to connect to the $\overline{\text{RRJCT}}$ signal or the $\overline{\text{SPISEL}}$ signal, then the SCC and/or SPI receives V_{DD} on that signal.

NOTE

In the description of the PIP, the PB0 pin, as well as other port B pins, can also be used as PIP functions. However, the PIP does not affect the operation of port B unless it is enabled. Therefore, the PIP description does not need to be studied by users of port B unless the PIP will be used in the application.

7.14.9 Port C Pin Functions

Port C consists of 12 general-purpose I/O pins with interrupt capability on each pin. Refer to Table 7-21 for the description of all port C pin options.

Table 7-21. Port CPin Assignment

Signal	PCPAR = 0		PCPAR = 1		Input to On-Chip Peripherals
	PCDIR = 1 or PCSO = 0	PCDIR = 0 and PCSO = 1	PCDIR = 0	PCDIR = 1	
PC0	Port C0	—	RTS1	L1ST1	—
PC1	Port C1	—	RTS2	L1ST2	—
PC2	Port C2	—	$\overline{\text{RTS3}}$/L1RQB	L1ST3	—
PC3	Port C3	—	$\overline{\text{RTS4}}$/L1RQA	L1ST4	—
PC4	Port C4	CTS1	—		GND
PC5	Port C5	CD1	TGATE1		GND
PC6	Port C6	CTS2	—		GND
PC7	Port C7	CD2	TGATE2		GND
PC8	Port C8	CTS3	L1TSYNCB	SDACK2	$\overline{\text{CTS3}}$ and/or L1TSYNCB = GND
PC9	Port C9	CD3	L1RSYNCB		GND
PC10	Port C10	CTS4	L1TSYNCA	SDACK1	$\overline{\text{CTS4}}$ and/or L1TSYNCA = GND
PC11	Port C11	CD4	L1RSYNCA		GND

EXTx— External Request to the RISC

0 = PCx is a general-purpose interrupt I/O pin, with the direction controlled in PCDIR. If PCDIR configures this pin as an input, this pin can generate an interrupt to the CPU32+ core, as controlled by the PCINT bits.

1 = PCx becomes an external request to the RISC controller instead of being a general-purpose interrupt pin. The corresponding PCINT bits control when a request is generated.

NOTE

EXTx should only be set, if the user is instructed to do so, during the initialization of a Motorola-supplied RAM microcode.

7.14.10.5 PORT C INTERRUPT CONTROL REGISTER (PCINT). PCINT is a 16-bit read-write register. Each defined bit in the PCINT corresponds to a port C line to determine whether the corresponding port C line will assert an interrupt request upon a high-to-low change or any change on the pin. The PCINT is cleared at reset.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
—				EDM11	EDM10	EDM9	EDM8	EDM7	EDM6	EDM5	EDM4	EDM3	EDM2	EDM1	EDM0

Bits 15–12—Reserved.

These bits should be written with zeros.

EDMx—Edge Detect Mode for Line x

The corresponding port C line (PCx) will assert an interrupt request according to the following:

0 = Any change on PCx generates an interrupt request.

1 = High-to low change on PCx generates an interrupt request.

7.15 CPM INTERRUPT CONTROLLER (CPIC)

The CPIC is the focal point for all interrupts associated with the CPM. The CPIC accepts and prioritizes all the internal and external interrupt requests from all functional blocks associated with the CPM. It is also responsible for generating a vector during the CPU interrupt acknowledge cycle.

The CPIC contains has the following key features:

- Twenty-Eight Interrupt Sources (16 Internal and 12 External)
- Sources May Be Assigned to a Programmable Interrupt Level (1–7)
- Programmable Priority Between SCCs
- Two Priority Schemes for the SCCs
- Programmable Highest Priority Request
- Fully Nested Interrupt Environment
- Unique Vector Number for Each Interrupt Source

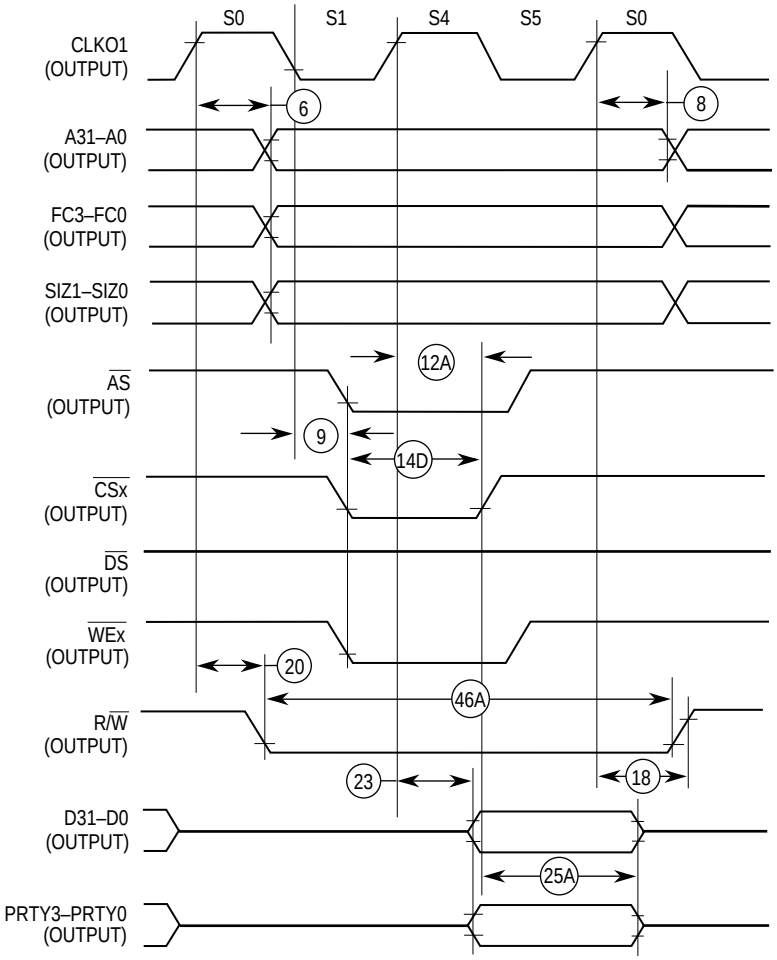


Figure 10-10. SRAM: Fast Termination Write Cycle (CSNTQ = 1)

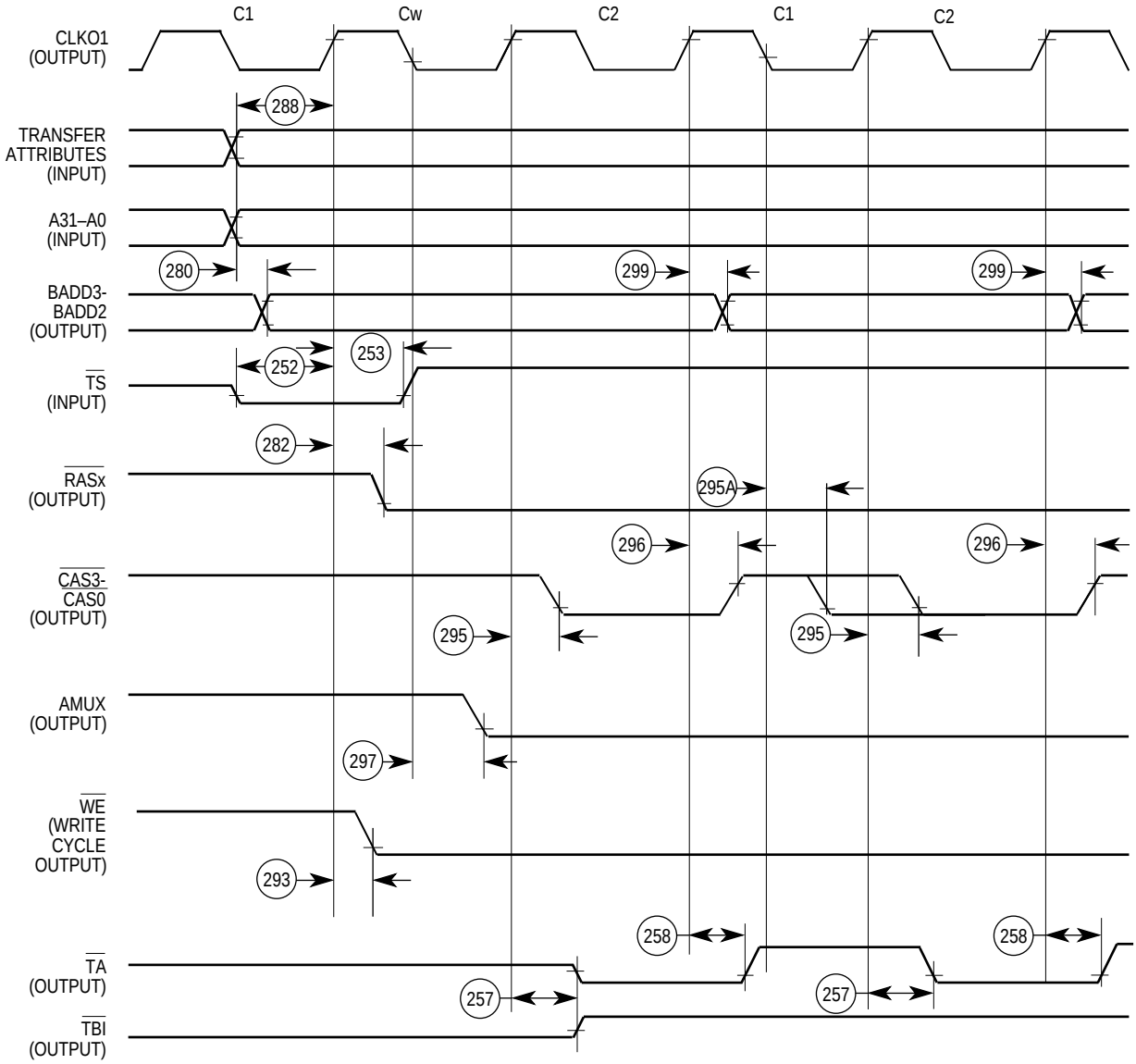


Figure 10-47. External MC68040 DRAM Burst Cycles Timing Diagram

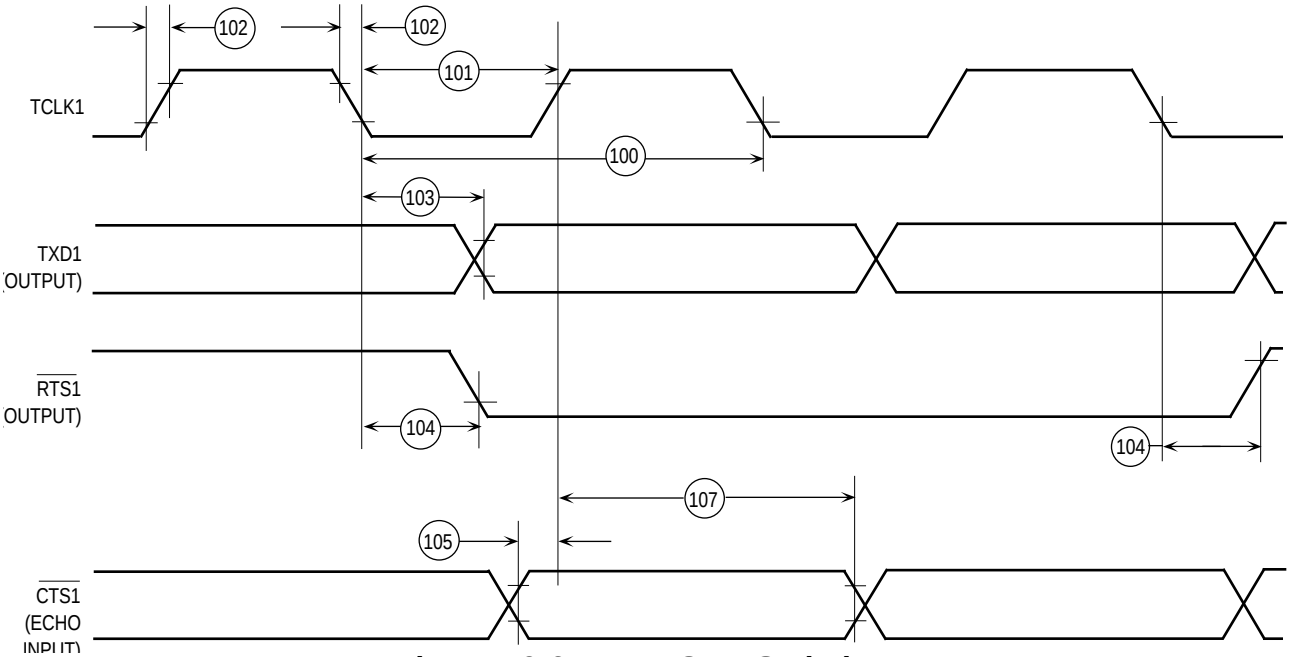


Figure 10-67. HDLC BUS Timing

10.26 ETHERNET ELECTRICAL SPECIFICATIONS

(The electrical specifications in this document are preliminary. See Figure 10-68–Figure 10-73)

Num.	Characteristic	3.3 V or 5.0 V 25.0 MHz		5.0 V 33.34 MHz		Unit
		Min	Max	Min	Max	
120	CLSN Width High	40	—	40	—	ns
121	RCLK1 Rise/Fall Time	—	15	—	15	ns
122	RCLK1 Width Low	CLKO1+ 5ns	—	CLKO1+ 5ns	—	
123[1]	RCLK1 Width high	CLKO1	—	CLKO1	—	
124	RXD1 Setup Time	20	—	20	—	ns
125	RXD1 Hold Time	5	—	5	—	ns
126	RENA Active Delay (from RCLK1 rising edge of the last data bit)	10	—	10	—	ns
127	RENA Width Low	100	—	100	—	ns
128	TCLK1 Rise/Fall Time	—	15	—	15	ns
129	TCLK1 Width Low	CLKO1+ 5ns	—	CLKO1+ 5ns	—	
130[1]	TCLK1 Width high	CLKO1	—	CLKO1	—	
131	TXD1 Active Delay (from TCLK1 rising edge)	10	50	10	50	ns
132	TXD1 Inactive Delay (from TCLK1 rising edge)	10	50	10	50	ns
133	TENA Active Delay (from TCLK1 rising edge)	10	50	10	50	ns
134	TENA Inactive Delay (from TCLK1 rising edge)	10	50	10	50	ns
135	RSTRT Active Delay (from TCLK1 falling edge)	10	50	10	50	ns
136	RSTRT Inactive Delay (from TCLK1 falling edge)	10	50	10	50	ns
137	RRJCT Width Low	1	—	1	—	CLKO1

- End Delimiter (ED) generation/checking
- Init Rx/Tx, Stop TX, Restart TX and Enter Hunt Mode Commands
- Token Rotation Timer, Idle Timer, Slot Timer and Time-out Timer
- Consumes 1280 bytes of the QUICC's internal memory.

C.6 ENHANCED ETHERNET FILTERING

The enhanced Ethernet filtering microcode has been created to add a bit more flexibility to the current Ethernet Controller. It allows the user to perfectly accept or reject frames with destination addresses that are contained in a table of 24 entries. In addition to the filtering capability, the microcode adds the ability to replace the externally sampled tag byte with one that is extracted from the address table.

C.6.1 Key Features

- Can accept or reject incoming frames if the destination MAC address is contained in a list of 24 preprogrammed entries.
- Allows a shorter list to obtain better peripheral performance.
- Can optionally tag incoming accepted frames with an 8-bit tag appended to the end of the frame (rather than using a CAM).
- The filtering can be turned off to allow tagging with no frame rejection.
- Can filter on group and individual addresses.
- Is a "small" microcode (consumes 768 bytes of DPRAM).
- Filtering is only supported on one Ethernet channel (SCC1), the other channel operates normally.

C.6.2 Performance

The overhead incurred by the filtering algorithm will depend upon the number of entries in the address table. Table C-5 shows possible configurations of other channels given the load added by a number of entries in the address table.

Table C-5. Channel Configuration

Number of addresses	Possible Configuration of Other Channels
1-8	1 x 10 Mbit Ethernet, 1 x 200 Kbit HDLC
9-16	1 x 10 Mbit Ethernet, 1 x 200 Kbit HDLC
17-24	1 x 10 Mbit Ethernet

D.1.2 Serial Interface

- Serial Multiplexed (Full Duplex) Input/Output 2048, 1544 or 1536 Kbps PCM - High-ways
- Compatible with T1/DS1 24 Channel and CEPT/E1 32 Channel PCM Highway, ISDN Basic Rate, ISDN Primary Rate, User Defined
- Sub Channeling On Each Time-Slot
- Allows Independent Transmit and Receive Routing, Frame Syncs, Clocking
- Concatenation of Any, Not Necessarily Consecutive, Time Slots to Channels Independently for Receive/Transmit
- Supports H0, H11, H12 ISDN - Channels
- Allows Dynamic Allocation of Channels
- Up to 3 Additional HDLC 64 Kbps Channels at 25 Mhz System Clock when in QMC Mode
- Ethernet Support at 33 Mhz System Clock on One SCC when Operating in QMC Mode

D.1.3 System Interface

- QUICC Powerful System Support Features: CPU32+, Slave Mode, 040 Companion, Memory Controller, 2xIDMA, SIM60 Watchdogs, Bus Monitors, Timers, Low Power Modes, Breakpoint Logic, Interrupt Controller
- On-Chip Bus Arbitration for Serial DMAs with No Performance Penalty
- Efficient Bus Usage (No Bus Usage For Non-Active Channel, and for Active Channels that Have Nothing to Transmit)
- Efficient Control of the Interrupts to the CPU
- Supports External Buffer Descriptors Table
- Using On-Chip Enlarged Dual-Port RAM for Parameter Storage

D.2 QUICC ARCHITECTURE OVERVIEW

The QUICC is 32-bit controller that is an extension of other members of the Motorola M68300 family. Like other members of the M68300 family, the QUICC incorporates the inter-module bus (IMB). (The MC68302 is an exception, having an M68000 bus on chip.) The IMB provides a common interface for all modules of the M68300 family, which allows Motorola to develop new devices more quickly by using the library of existing modules. Although the IMB definition always included an option for an on-chip 32-bit bus, the QUICC is the first device to implement this option.

The QUICC consists of three modules: the CPU32+ core, the SIM60, and the CPM. Each module uses the 32-bit IMB. The MC68360 QUICC block diagram is shown in Figure D-1.

Ethernet Controller 7-235
Ethernet Memory Map 7-246
Ethernet on QUICC 7-236
Ethernet Programming Model 7-250
Ethernet Receive Buffer Descriptor 7-258
Ethernet Transmit Buffer Descriptor 7-261
EVT 9-16
Exception
    Bus Exception 4-41
Exception Handler 5-34, 5-39, 5-46, 5-47, 5-49, 5-51, 5-52, 5-53, 5-54, 5-55
Exception Priorities 5-38
Exception Processing 5-34, 5-39, 5-46
Exception Processing 5-36, 5-38, 5-39, 5-41, 5-43, 5-47, 5-52, 5-57, 5-82
Exception Vectors 5-36, 5-43, 5-59
EXTAL 2-11
External Burst Requests 7-40
External Bus Interface Control 6-21
External Bus Masters 9-18
External Components 6-14
External Cycle Steal 7-42
External Exception 5-36
External Master Support 6-63
External Master Wait State 6-67
External Sync 7-115
External Synchronization Signals 7-223

F

Fast Termination 4-21
Fast-Termination Option 7-50
Fault Address Register 5-63
Fault Correction 5-63
Fault Types 5-51, 5-52
FC 2-5
FC3–FC0 4-3
FCR 7-31
Fetch Effective Address Timing Table 5-90
FIFO 7-3, 7-179, 7-209
FIFO Latency 7-114
FIFO Length 7-114
FIFO Width 7-114
Fill Memory Block 5-76
First-Time User 7-238
Flag Status 7-186
Flash EPROM 9-5, 9-41
F-Line Instructions 5-44
FM0 7-116, 7-118, 7-136, 7-197
FM1 7-116, 7-118, 7-136
FORCE_BGND 5-67
Format Error Exception 5-43, 5-48
Four-Word Stack Frame 5-56
Fractional Stop Bits 7-153
Frame Sync Delay 7-81
Frames Threshold 7-174
FREEZE 5-62, 5-66, 6-31, 7-60
Freeze 2-12
Freeze Support 6-11
Frequency Multiplication 6-14
Function Codes 2-5, 4-3

G

GADDR 7-249
GCI 7-80, 7-96, 7-268
    SMC 7-305
GCI Controller 7-305
GCIDCL 7-97
General System Clock 6-16
General-Purpose Chip-Select Overview 6-56
General-purpose Timers 7-17
Glitch 7-185
Glitch Detect 7-112, 7-139
Global (Boot) Chip-select 6-58
Global Chip Select 6-25
Global Memory Register 6-64
Glueless System 1-8
GMR 6-56, 6-64, 9-11, 9-49
GND 2-13
GNDCLK 6-19
GNDS 2-13
GNDSYN 2-13, 6-19
GO 5-63, 5-70, 5-77
GRACEFUL STOP TRANSMIT 7-120, 7-148, 7-175, 7-205, 7-226, 7-250
Grant Support 7-86
Grouped 7-380
GSMR 7-111
GSMR programming 7-196, 7-199

H

HADDR 7-173
HALT 2-10, 4-20, 4-41, 4-46, 7-44, 7-51, 7-