Welcome to E-XFL.COM

Embedded - Microcontrollers - Application Specific: Tailored Solutions for Precision and Performance

Embedded - Microcontrollers - Application Specific represents a category of microcontrollers designed with unique features and capabilities tailored to specific application needs. Unlike general-purpose microcontrollers, application-specific microcontrollers are optimized for particular tasks, offering enhanced performance, efficiency, and functionality to meet the demands of specialized applications.

What Are Embedded - Microcontrollers - Application Specific?

Application-specific microcontrollers are engineered to

Details	
Product Status	Obsolete
Applications	Network Processor
Core Processor	ARM7®
Program Memory Type	External Program Memory
Controller Series	NET+40
RAM Size	External
Interface	EBI/EMI, Ethernet, DMA, HDLC, IEEE1284/ENI, SPI, UART
Number of I/O	24
Voltage - Supply	3V ~ 3.6V
Operating Temperature	-40°C ~ 85°C
Mounting Type	Surface Mount
Package / Case	208-BFQFP
Supplier Device Package	208-PQFP (28x28)
Purchase URL	https://www.e-xfl.com/product-detail/digi-international/net-40-qitro-4

external peripheral devices. The data bus is bi-directional. When the NET+ARM chip is the system bus master, the data bus is an output during write cycles and input during read cycles. When an external bus master is the system bus master, the data bus is an input during write cycles and an output during read cycles.

The D0 bit is set to 1 to indicate that Byte 0 is dirty and inconsistent with external memory. Byte 3 will be updated in external memory when this cache entry is flushed to make room for a new entry. The dirty bit can only be set when the CCR is configured to operate in copy-back mode. Byte 0 refers to the most significant address byte of a 32-bit word when operating in Big Endian mode, the least significant byte when operating in Little Endian mode.

5.3 Ethernet Controller Configuration

The Ethernet controller has a block of configuration space that is mapped into the DMA module configuration space as defined in Table 2-1: BBus Address Decoding.

Address	Register
FF80 0000	Ethernet General Control Register
FF80 0004	Ethernet General Status Register
FF80 0008	Ethernet FIFO Data Register
FF80 000C	
FF80 0010	Ethernet Transmit Status Register
FF80 0014	Ethernet Receive Status Register
FF80 0400	MAC Configuration Register
FF80 0404	MAC Test Register
FF80 0408	PCS Configuration Register
FF80 040C	PCS Test Register
FF80 0410	STL Configuration Register
FF80 0414	STL Test Register
FF80 0440	Transmit Control Registers
FF80 0464	
FF80 0480	Receive Control Registers
FF80 0488	
FF80 04C0	Link Fail Counter
FF80 0500	10Mbit Jabber Counter
FF80 0504	10Mbit Loss of Carrier Counter
FF80 0540	MII Command Register
FF80 0544	MII Address Register
FF80 0548	MII Write Data Register
FF80 054C	MII Read Data Register
FF80 0550	MII Indicators Register
FF80 0580	CRC Error Counter

Table 5-1: Ethernet Configuration Registers

DADR

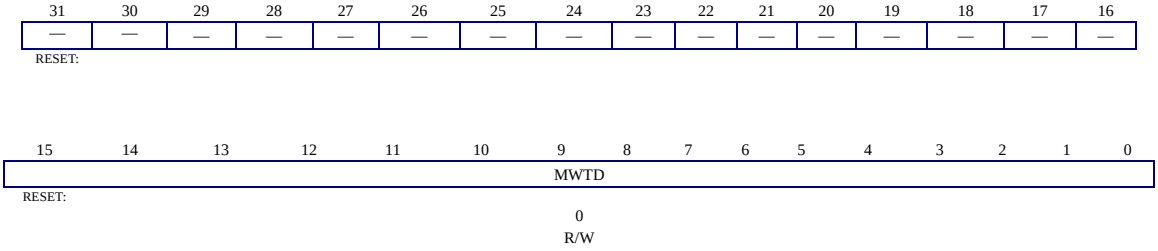
Device Address: This field represents the 5-bit PHY Device Address field for management cycles. Up to 31 different PHY devices can be addressed; address 0 is reserved. The default value for this field is 0.

RADR

Register Address: This field represents the 5-bit PHY Register Address field for management cycles. Up to 32 registers within a single PHY Device can be addressed. The default value for this field is 0.

5.3.15.3 MII Write Data Register

Address = FF80 044C

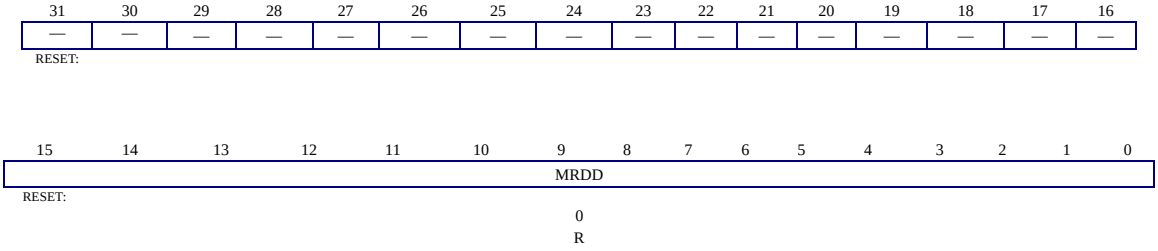


MWTD

Write Data: When this register is written an MII Management write cycle is performed using the 16-bit data the pre-configured PHY Device and PHY Register addresses defined in the PHY Address register. The write operation is completed when the BUSY bit in the MII Indicators Register returns to 0.

5.3.15.4 MII Read Data Register

Address = FF80 054C

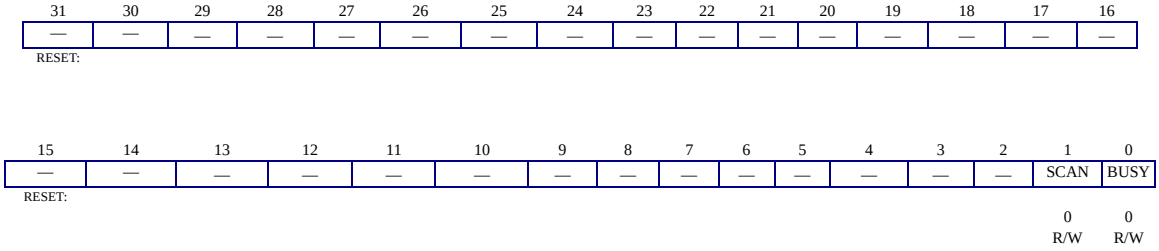


MRDD

Read Data: Following an MII Management read cycle, the read data is obtained by reading from this register. An MII Management read cycle is executed by loading the PHY Address Register then setting the READ bit in the MII Command Register to 1. Read data is available after the BUSY bit in the MII Indicators register returns to 0.

5.3.15.5 MII Indicators Register

Address = FF80 0550



SCAN

Scanning: A 1 in this bit position indicates continuous MII Management Scanning read operations are in progress.

BUSY

Busy: A 1 in this bit position indicates the MII Management module is currently performing an MII Management Read or Write Cycle. This bit will return to 0 when the operation is complete.

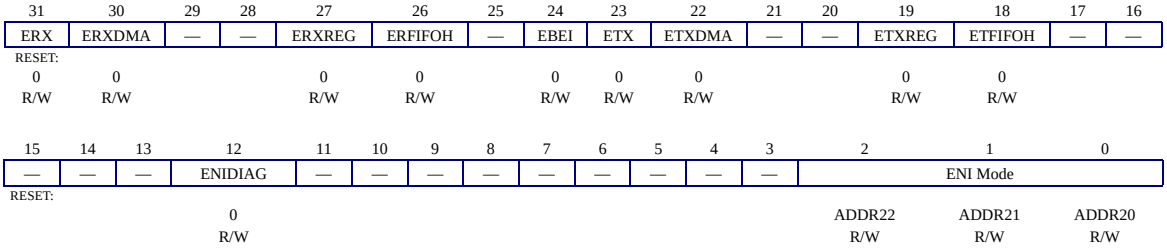
TXMCR

The TXMCR counter is incremented each time a packet transmission is aborted because it received too many collisions. The maximum number of allowable collisions is defined by the RETRY field in the Collision Window / Collision Retry Register. The counter is automatically cleared when read provided the AUTOZ bit is set in the STL Configuration Register.

6.5.2 General Control Register

The general control register is a 32-bit register that contains control bits which affect the entire ENI controller module.

Address = FFA0 0000



ERX **Enable Receive FIFO**

0 = Disables inbound data flow and resets the FIFO

1 = Enables inbound data flow

The ERX bit must be set to active high to allow data to be received through the FIFO interface. The ERX bit can be used to reset the receive side FIFO. In general, the ERX bit should be set once on device open.

ERXDMA **Enable Receive FIFO DMA**

0 = Disables inbound DMA data request (use to stall receiver)

1 = Enables inbound DMA data request

The ERXDMA bit must be set to active high to allow the receive FIFO to issue receive data move requests to the DMA controller. DMA Channel 3 is used for the ENI receive FIFO. This bit can be cleared to temporarily stall receive side FIFO DMA. This bit should not be set when operating the Ethernet receiver in interrupt service mode. In general, the ERXDMA bit should be set once on device open.

ERXREG **Enable Receive FIFO Data Ready Interrupt**

The ERXREG bit must be set to active high to generate an interrupt when data is available in the receive FIFO. The ERXREG bit should only be set when operating the receive FIFO in interrupt service mode instead of DMA mode. In general, DMA mode is more desirable than interrupt service mode. The receive FIFO interrupt is routed to the GEN Module Interrupt Controller via the ENI Port 1 Interrupt (refer to Section *6.5.9 ENI Module Interrupts*).

ERFIFOH **Enable Receive FIFO Half Full Interrupt**

The ERFIFOH bit must be set to active high to generate an interrupt when the receive FIFO is at least half full (16 bytes). The ERFIFOH bit should only be set when operating the receive FIFO in interrupt service mode instead of DMA mode. In general, DMA mode is more desirable than

(refer to Table 6-1 *IEEE 1284 Data Bus Assignments*).

MSTB* **1284 Manual Strobe Configuration**

The MSTB* bit controls the state of the STROBE* signal in the outside control latch when the MAN bit is set to 1. The MSTB* bit is used to manually manipulate the state of the STROBE* signal. This feature is primarily used for 1284 Negotiation. Refer to Section *6.2.4 IEEE Negotiation* for more details.

AUTOFD* **1284 AUTOFD* Setting**

The AUTOFD* bit controls the state of the AUTOFD* signal in the external control latch. The value written into the AUTOFD* field will appear on the AUTOFD* signal in the external control latch (refer to Table 6-1 *IEEE 1284 Data Bus Assignments*).

INIT* **1284 INIT* Setting**

The INIT* bit controls the state of the INIT* signal in the external control latch. The value written into the INIT* field will appear on the INIT* signal in the external control latch (refer to Table 6-1 *IEEE 1284 Data Bus Assignments*).

HSELECT* **1284 HSELECT* Setting**

The HSELECT* bit controls the state of the HSELECT* signal in the external control latch. The value written into the HSELECT* field will appear on the HSELECT* signal in the external control latch (refer to Table 6-1 *IEEE 1284 Data Bus Assignments*).

INTEF **1284 Port FIFO Empty Interrupt Request Enable**

The INTEF bit must be set active high to generate an interrupt when the 1284 data FIFO is empty. Setting the INTEF bit to 1 will cause an interrupt to be generated when the FIFOE status bit is 1. The INTEF bit should only be set when operating the 1284 Port in interrupt service mode instead of DMA mode. In general, DMA mode is more desirable than interrupt mode.

EPP **1284 EPP Mode of Operation**

The EPP bit must be set to 1 to allow the 1284 port to operate in the EPP mode of operation. Refer to Section *6.2.10 IEEE 1284 EPP Mode* for more details on how to operate the 1284 port in EPP mode.

IBRIE **1284 Inbound Buffer Ready Interrupt Enable**

The IBRIE bit must be set to active high to generate an interrupt when the 1284 Port Inbound Buffer is available for reading. Setting the IBRIE bit to 1 will cause an interrupt to be generated when the IBR status bit is 1. The IBRIE bit should only be set when operating the 1284 Port in interrupt service mode instead of DMA mode. In general, DMA mode is more desirable than interrupt mode.

IBR **1284 Inbound Buffer Ready**

The IBR bit will go active high when the inbound buffer is ready to be read by the processor. The

3. If RRDY is True, then:
   a. Read the Data FIFO.
   b. Use the RXFDB field to pick out valid bytes.

To facilitate an interrupt when either the RRDY or RBC status bits are active, the processor must set one or both of the corresponding interrupt enables found in control register A (D11 and D09). The appropriate interrupt enable (SER 1 RX or SER 2 RX) bit in the GEN module interrupt enable register must also be set.

The receive FIFO can also be emptied using the DMA controller (DMA channels 7 and 9). When using DMA, the processor need not interface with any of the serial port registers for data flow, instead, the processor must interface with the DMA channel registers and the DMA buffer descriptor block attached to DMA channels 7 and 9. To facilitate the use of transmit DMA, the ERXDMA bit in serial channel control register A must be set active high. When ERXDMA is set active high, the serial receiver interrupts should be disabled.

7.2.3.3 SPI Master Mode

SPI master mode controls the flow of data between memory in the master SPI interface and an external SPI slave peripheral. The SPI master determines the numbers of bytes for transfer. The SPI master port simultaneously transmits and receives the same number of bytes. A single clock signal controls the transfer of information. For the SPI master interface, the clock signal is an output. Transfer of information is also qualified with an enable signal. The SPI master controls the SPI enable signal. The SPI enable signal must be active low for data transfer to occur, regardless of the SPI clock signal. The SPI enable function allows for multiple slaves to be individually addressed in a multi-drop configuration.

Configuration

The GEN module must be properly configured to allow the SPI interface signals from the SER module to interface with the PORTA/C GEN module GPIO pins.

SPI Channel A
PORTA7 - Configure as Special Function Output (SPI TXD)
PORTA3 - Configure as Special Function Input (SPI RXD)
PORTA4 - Configure as Special Function Output (SPI Enable)
PORTC7 - Configure as Special Function Output (SPI Clock)

.

Configuration

The GEN module must be properly configured to allow the SPI interface signals from the SER module to interface with the PORTA/C GEN module GPIO pins.

SPI Channel A
PORTA7 - Configure as Special Function Output (SPI TXD)
PORTA3 - Configure as Special Function Input (SPI RXD)
PORTA4 - Configure as Special Function Input (SPI Clock)
PORTC7 - Configure as Special Function Input (SPI Enable)

SPI Channel B
PORTB7 - Configure as Special Function Output (SPI TXD)
PORTB3 - Configure as Special Function Input (SPI RXD)
PORTB4 - Configure as Special Function Input (SPI Clock)
PORTC5 - Configure as Special Function Input (SPI Enable)

The SER module must be properly configured to operate in either master or slave mode. For slave mode operation, the MODE field in control register B must be set to a value of “11” before the CE field in control register A is set to 1.

The following is the suggested ordering of configuration for SPI slave mode of operation.

1. Reset the serial port by writing a 0 to control register A.
2. Configure the bit-rate register:

   EBIT- 1 for enable
   TMODE- 1 for 1X mode
   RXSRC- 1 for external
   TXSRC- 1 for external
   RXEXT- 0 for disable
   TXEXT- 0 for disable
   CLKMUX- N/A (external timing source)
   TXCINV- 0 for normal
   RXCINV - 0 for normal
   N - N/A (external timing source)
3. Configure the buffer GAP timer:

RBRK

The receive break condition (RBRK) field indicates a UART receive break condition has been detected. When set, the RBRK field remains set until acknowledged. The RBRK bit is acknowledged by writing a 1 to the same bit position in status register A. The RBRK status condition can be programmed to generate an interrupt by setting the IE bit in control register A.

RFE

The receive framing error (RFE) field indicates a receive framing error condition has been detected. When set, the RFE field remains set until acknowledged. The RFE bit is acknowledged by writing a 1 to the same bit position in status register A. The RFE status condition can be programmed to generate an interrupt by setting the IE bit in control register A.

RPE

The receive parity error (RPE) field indicates a receive parity error condition has been detected. When set, the RPE field remains set until acknowledged. The RPE bit is acknowledged by writing a 1 to the same bit position in status register A. The RPE status condition can be programmed to generate an interrupt by setting the IE bit in control register A.

ROVER

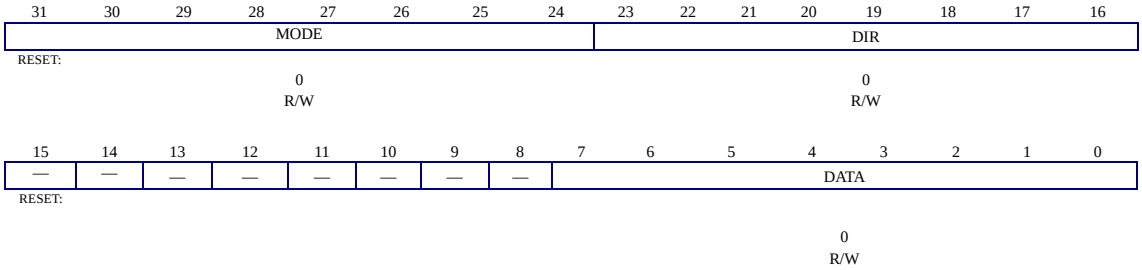
The receive overrun (ROVER) field indicates a receive overrun error condition has been detected. An overrun condition indicates the FIFO was full while data needed to be written by the receiver. When set, the ROVER field remains set until acknowledged. The ROVER bit is acknowledged by writing a 1 to the same bit position in status register A. The ROVER status condition can be programmed to generate an interrupt by setting the IE bit in control register A.

RRDY

The RRDY field indicates data is available to be read from the FIFO data register. Before reading the FIFO data register, the RXFDB field in status register A must be read to determine how many active bytes are available during the next read of the FIFO data register. The RRDY field is typically only used in interrupt driven applications, the RRDY field is not used for DMA operation. The RRDY status condition can be programmed to generate an interrupt by setting the IE bit in control register A.

The RRDY bit is never active while the RBC bit is active. The RBC bit must be acknowledged to activate the RRDY bit. When the receiver is configured to operate in DMA mode, the interlock between RBC and RRDY is handled automatically in hardware.

Address = FFB0 0020



MODE

The 8-bit MODE field is used to individually configure each of the PORTA pins to operate in either GPIO mode or Special Function mode. Setting the MODE bit to 0 selects GPIO mode while a 1 selects Special Function Mode.

Each bit in the MODE field corresponds to one of the NET+ARM PORTA bits. D31 controls PORTA7, D30 controls PORTA6 … D24 controls PORTA0.

DIR

The 8-bit DIR field is used to individually configure each of the PORTA pins to operate in either Input Mode or Output Mode. Setting the DIR bit to 0 selects Input mode while a 1 selects Output Mode.

Each bit in the DIR field corresponds to one of the NET+ARM PORTA bits. D23 controls PORTA7, D22 controls PORTA6 … D16 controls PORTA0.

DATA

The 8-bit DATA field is used when a PORTA bit is configured to operate in GPIO mode. Reading the DATA field provides the current state of the NET+ARM GPIO signal (regardless of its configuration mode). Writing the DATA field defines the current state of the NET+ARM GPIO signal when the signal is defined to operate in GPIO output mode. Writing a DATA bit when configured in GPIO input mode or Special Function Mode has no effect.

Each bit in the DATA field corresponds to one of the NET+ARM PORTA bits. D07 controls PORTA7, D06 controls PORTA6 … D00 controls PORTA0.

8.2.11 Interrupt Enable Register - Set

The Interrupt Enable - Set register is a 32-bit write-only register. This register sets specific bits without affecting the other bits. Writing a one to a bit position in this register sets the respective bit in the Interrupt Enable Register. A zero in any bit position has no affect.

Address = FFB0 0034

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DMA1	DMA2	DMA3	DMA4	DMA5	DMA6	DMA7	DMA	DMA9	DMA10	ENI PORT 1	ENI PORT 2	ENI PORT 3	ENI PORT 4	ENET RX	ENET TX
RESET:															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SER 1 RX	SER 1 TX	SER2 RX	SER 2 TX	—	—	—	—	—	Watch Dog	Timer 1	Timer 2	PortC PC3	PortC PC2	PortC PC1	PortC PC0
RESET:															
0	0	0	0						0	0	0	0	0	0	0
R/W	R/W	R/W	R/W						R/W	R/W	R/W	R/W	R/W	R/W	R/W

8.2.12 Interrupt Enable Register - Clear

The Interrupt Enable - Clear register is a 32-bit write-only register. This register clears specific bits without affecting the other bits. Writing a 1 to a bit position in this register clears the respective bit in the Interrupt Enable Register. A zero in any bit position has no affect.

Address = FFB0 0038

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DMA1	DMA2	DMA3	DMA4	DMA5	DMA6	DMA7	DMA	DMA9	DMA10	ENI PORT 1	ENI PORT 2	ENI PORT 3	ENI PORT 4	ENET RX	ENET TX
RESET:															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SER 1 RX	SER 1 TX	SER2 RX	SER 2 TX	—	—	—	—	—	Watch Dog	Timer 1	Timer 2	PortC PC3	PortC PC2	PortC PC1	PortC PC0
RESET:															
0	0	0	0						0	0	0	0	0	0	0
R/W	R/W	R/W	R/W						R/W	R/W	R/W	R/W	R/W	R/W	R/W

Chapter 9
BUS Controller Module

The bus controller is responsible for moving data between the BBus and the external system bus. The bus controller can support dynamic bus sizing for any logical addresses selected by the memory controller. The bus controller packs 8-bit bytes and 16-bit words into the proper location on the 32-bit BBus.

Figure 9-1 provides a simple block diagram of the bus controller module. The BUS module works in close concert with the MEM module to support the dynamic bus sizing feature. The BUS module is responsible for performing the byte/word data packing when accessing less than a 32-bit data operand.

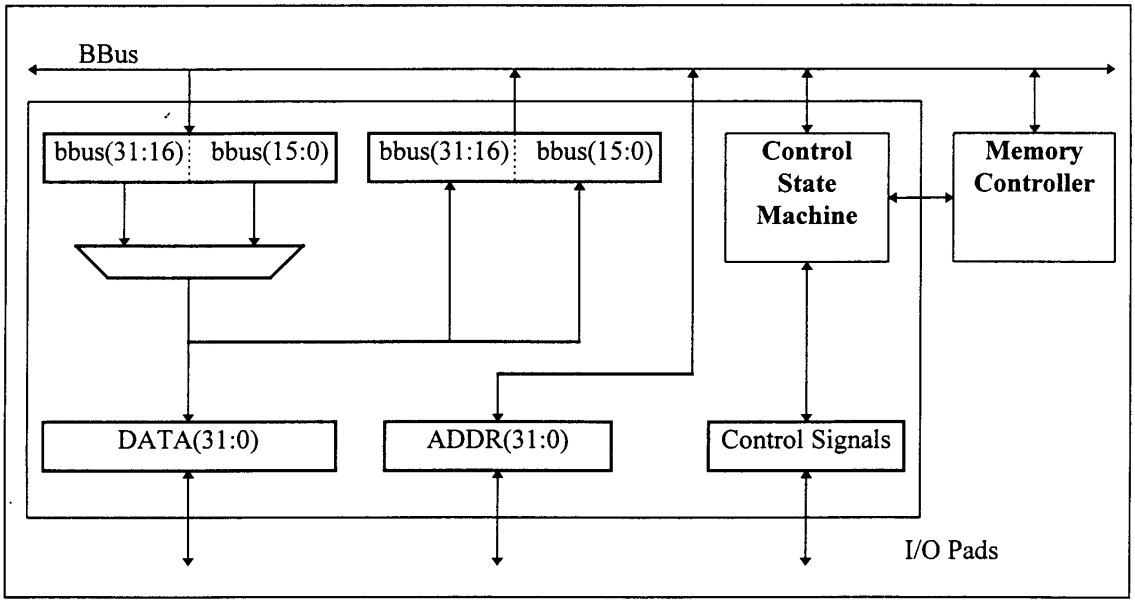


Figure 9-1: Bus Controller Module

When the NET+ARM chip interfaces with an external bus master or CPU, then the bus controller is responsible for providing arbitration of the system bus. The bus controller can be configured to allow the NET+ARM chip to be the default system master (other devices must request ownership from the NET+ARM chip). Alternately,

9.9.6 Completion

The external bus master owns the NET+ARM chip system bus until it relinquishes the bus by driving the BUSY* signal inactive high. The external bus master is responsible for maintaining fairness. When the external bus master does relinquish the bus, it must tri-state the following signals: BUSY*, TS*, ADDR, RW*, and BE3:0. Before tri-stating the BUSY* signal, the external bus master must drive BUSY* to a solid high first. This is required to ensure a good rise-time on the BUSY* signal.

9.9.7 Throttling

The external bus master needs to be concerned about bus utilization and fairness. The external bus master should not maintain ownership of the NET+ARM chip system memory bus for long extended periods of time. Doing so can cause bandwidth problems for the internal DMA controller. In general, an external bus master should maintain ownership of the system bus for something on the order of 4-8 memory cycles only.

10.5 NET+ARM External DRAM Address Multiplexing

The NET+ARM chip can be configured to use DRAM components using an external DRAM address multiplexer.

An external address multiplexer is required when designing a system with an external bus master. When interfacing with an external bus master, the NET+ARM chip memory controller cannot multiplex the address signals out the A13:A0 address pins since the NET+ARM chip would require those pins to be both an output (for the multiplexed signal) and an input (address inputs from the external bus master) at the same time, which is impossible.

An external address multiplexer may also be required when the selected SDRAM component cannot interface with the NET+ARM chip internal multiplexer. This happens as DRAM densities grow and the DRAM requirements for RAS/CAS addressing change.

Even though an external address multiplexer is used, the NET+ARM chip memory controller can be used to control the basic DRAM signal protocol. The NET+ARM chip can be configured to output the DRAM address multiplexer select signal out the PORTC3 pin. This is accomplished by setting the AMUX or the AMUX2 bit in the memory module configuration register (MMCR) or the DMUXS bit in the chip select base address register.

The AMUX bit is set to indicate that the internal address multiplexer must be disabled. When AMUX is set, the NET+ARM chip drives the address bus using standard addressing without any multiplexing. When AMUX is set, the internal address multiplexer is disabled and the multiplexer indicator is driven out the PORTC3 pin.

The AMUX2 control bit allows the internal multiplexer to operate normally and forces the PORTC3 signal to be driven. Using the AMUX2 setting allows the internal bus masters to use the internal address multiplexer and the external bus masters to perform their own address multiplexing by watching the state of the PORTC3 pin. The AMUX2 feature removes the need for an external physical address multiplexer (for example, F157 chips) provided the external bus master can perform the address multiplexing itself.

The DMUXS bit is set to indicate that the internal address multiplexer must be disabled when the specific chip select is activated. When DMUXS is set, the NET+ARM chip drives the address bus using standard addressing without any multiplexing, but only for the specific chip select. When DMUXS is set, the internal address multiplexer is disabled and the multiplexer indicator is driven out the PORTC3 pin.

Chapter 11
SYS Module

The SYS module provides the NET+ARM chip with its system clock and system reset resources.

11.1 System Clock Generation

The system clock is provided by an external crystal, an internal crystal oscillator, and an internal phase locked loop. The crystal oscillator circuit generates an internal CMOS transition clock signal. The clock signal from the crystal oscillator is initially divided by 5 and provided as an input to the phase locked loop (PLL).

The PLL is configured to operate in a multiplier mode. The output of the PLL is fed back through a four-bit programmable divider. This allows the PLL output to operate anywhere from 1 to 16 times the output of the prescaler, or 1/5th to 3X that of the input crystal frequency. The configuration for the PLL multiplication factor is provided by registers in the GEN module.

After multiplication, the system clock is divided down by the same value loaded in the PLL divider. This produces a clock signal that is always 1/5th the frequency of the crystal input regardless of what rate the PLL is programmed for. This clock is used as a reference timing pulse in both the GEN, and SER modules. Using a 18.432 MHz crystal as an example, the reference pulse generates a timing reference of 3.6864 MHz (quite useful for serial bit-rate generation and timers). The 1/5th reference timing pulse remains constant no matter what rate the PLL is programmed for system clock generation.

The PLL can be disabled and bypassed by asserting the PLLTST* input active low. In this mode, the system clock is provided by the XTAL1 input. Refer to Section *12.2 PLL* for more details on PLL testing.

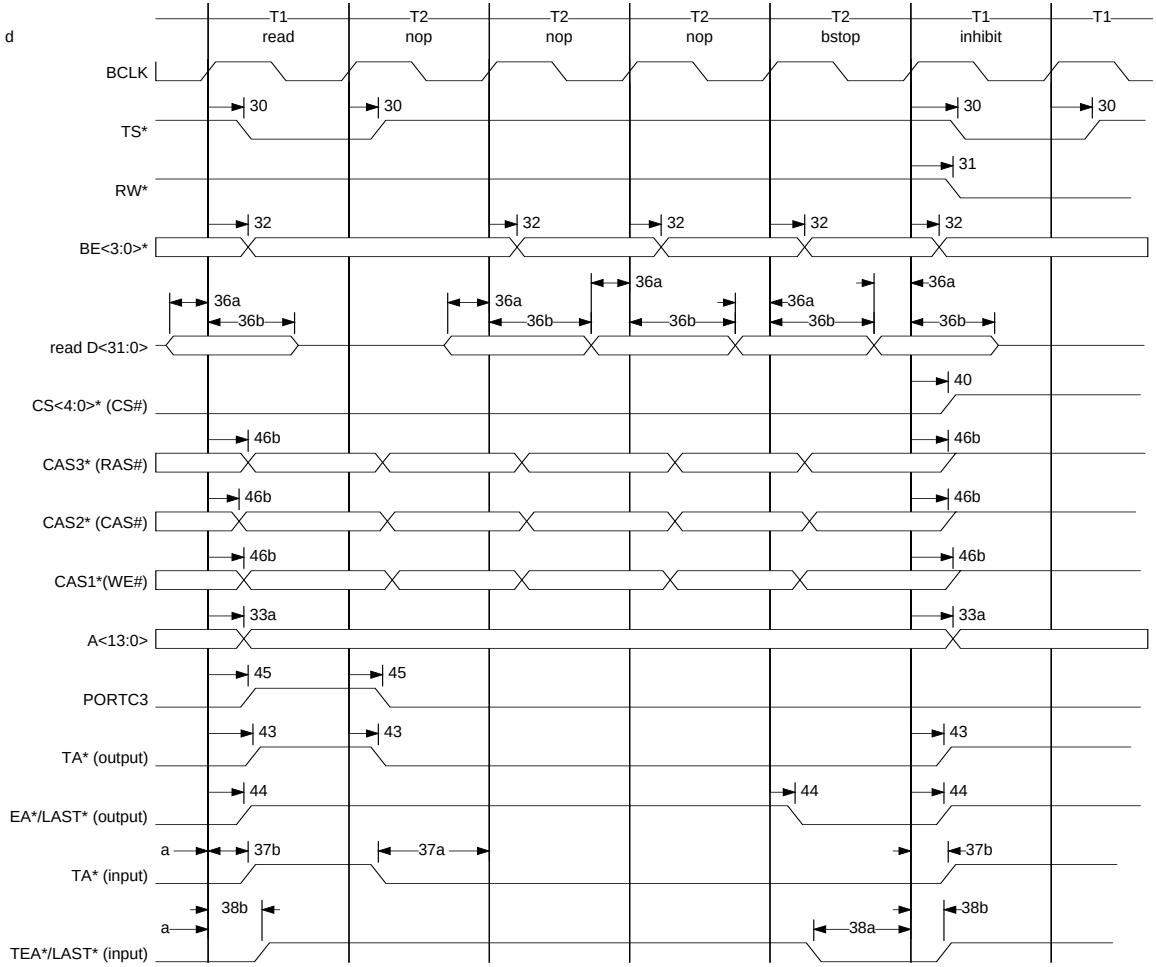


Figure 13-11: SDRAM Burst Read

Refer to Tables 13-9 and 13-10 for values.

Num	Characteristic	Min	Max	Unit
115	DATA valid to STROBE* low (setup)	STROBE		ns
117	STROBE* low to BUSY (input) high	STROBE		ns
118	BUSY high to STROBE* high (hold)	3 * T_{sys}		ns
120	STROBE* high to DATA change (hold)	STROBE		ns
121	BUSY low to STROBE* (low)	3 * T_{sys}		ns

Table 13-19: 1284 SLOW Forward ECP Mode Timing (FAST = 0)

Num	Characteristic	Min	Max	Unit
115	DATA valid to STROBE* low (setup)	3 * T_{sys}		ns
117	STROBE* low to BUSY (input) high	0		ns
118	BUSY high to STROBE* high (hold)	3 * T_{sys}		ns
120	STROBE* high to DATA change (hold)	3 * T_{sys}		ns
121	BUSY low to STROBE* (low)	3 * T_{sys}		ns

Table 13-20: 1284 FAST Forward ECP Mode Timing (FAST = 1)

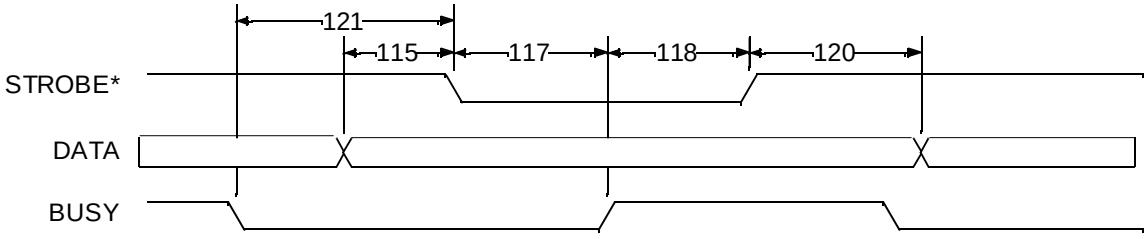


Figure 13-31: 1284 Forward ECP Mode Timing

Num	Characteristic	Min	Max	Unit
130	SPI Clock Low to SPI Enable Low	$-T_{SYS}$	T_{SYS}	ns
131	SPI Clock low to TXD valid	$-T_{SYS}$	T_{SYS}	ns
132	RXD Input Valid to SPI Clock High (setup)	20		ns
133	SPI Clock High to RXD Input Change (hold)	0		ns
134	SPI Clock High to SPI Enable High	½ Bit-Time		ns

Table 13-21: SPI Master Timing

T_{SYS} refers to the period of the NET+ARM chip system clock. If the NET+ARM chip is operating at 33Mhz, then T_{SYS} is equal to 30ns. ½ bit-time is a function of how fast the SPI port is configured to operate at. If the SPI port is configured to operate at its maximum speed of 4Mbps, then ½ bit-time would be equal to 120ns.

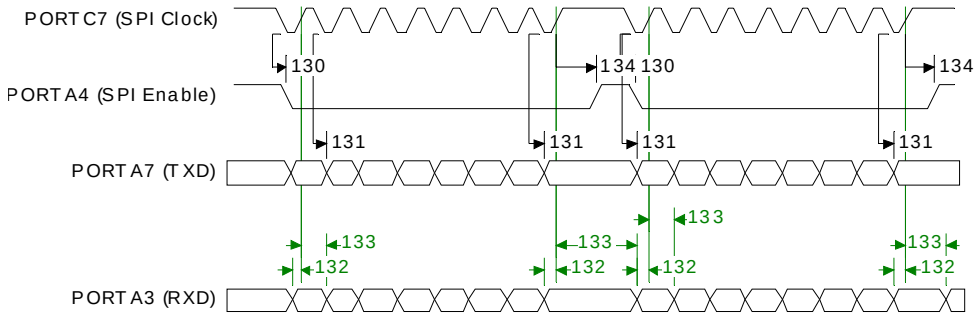


Figure 13-32: SPI Master Mode Waveform

Refer to Table 13-21 for values.