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	CPU32
Core Size	32-Bit Single-Core
Speed	16MHz
Connectivity	EBI/EMI, UART/USART
Peripherals	-
Number of I/O	47
Program Memory Size	-
Program Memory Type	ROMless
EEPROM Size	-
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 5.5V
Data Converters	A/D 8x8/10b
Oscillator Type	External
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	132-BQFP Bumpered
Supplier Device Package	132-PQFP (24.13x24.13)
Purchase URL	https://www.e-xfl.com/pro/item?MUrl=&PartUrl=mc68334gceh16

TABLE OF CONTENTS

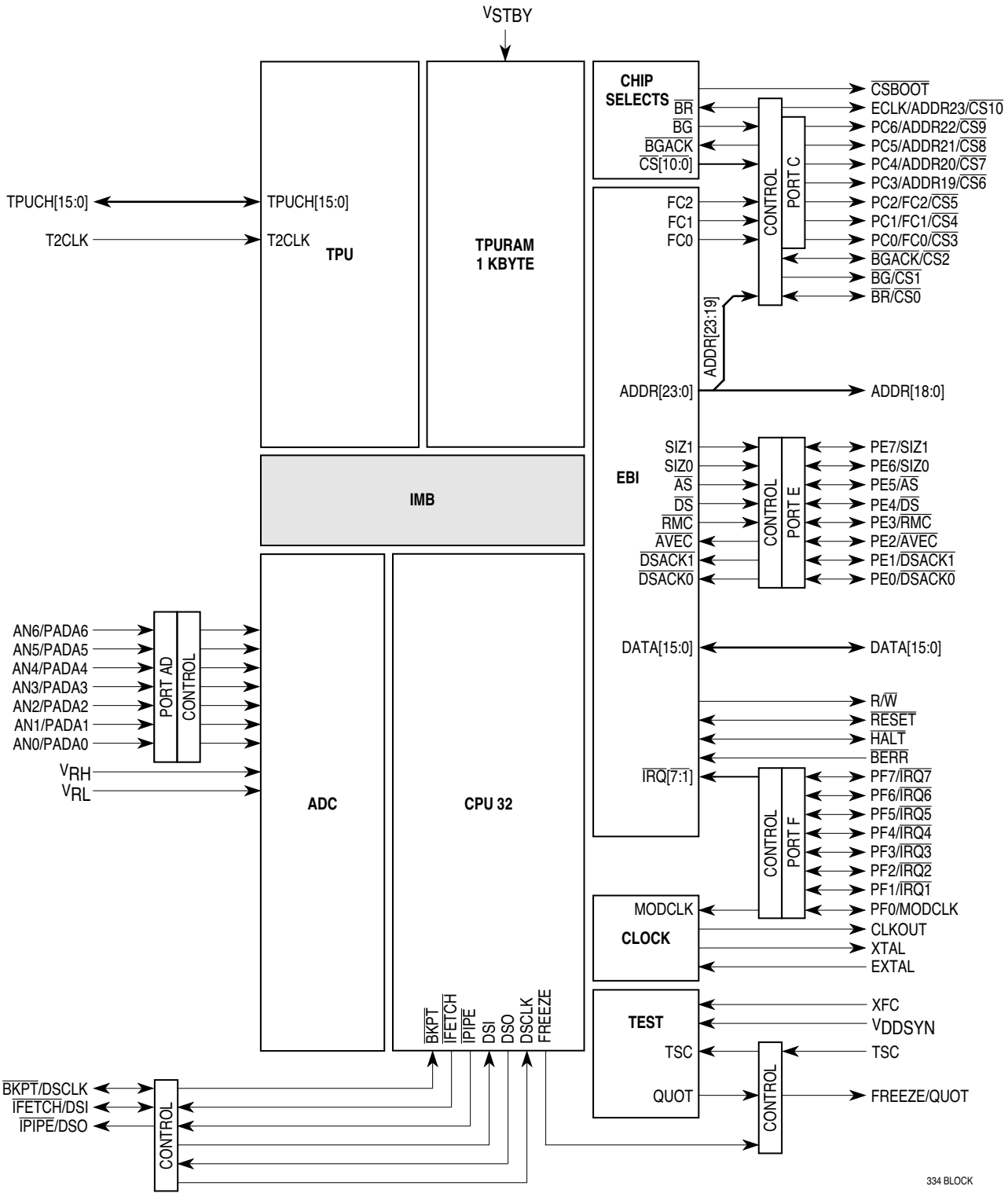


Figure 1 MC68334 Block Diagram

2.5 Signal Function

Table 6 MCU Signal Function

Mnemonic	Signal Name	Function
ADDR[23:0]	Address Bus	24-bit address bus used by CPU32
AN[6:0]	ADC Analog Input	Inputs to ADC multiplexer
$\overline{AS}$	Address Strobe	Indicates that a valid address is on the address bus
$\overline{AVEC}$	Autovector	Requests an automatic vector during interrupt acknowledge
$\overline{BERR}$	Bus Error	Indicates that a bus error has occurred
$\overline{BG}$	Bus Grant	Indicates that the MCU has relinquished the bus
$\overline{BGACK}$	Bus Grant Acknowledge	Indicates that an external device has assumed bus mastership
$\overline{BKPT}$	Breakpoint	Signals a hardware breakpoint to the CPU
$\overline{BR}$	Bus Request	Indicates that an external device requires bus mastership
CLKOUT	System Clockout	System clock output
$\overline{CS}$[10:0]	Chip Selects	Select external devices at programmed addresses
$\overline{CSBOOT}$	Boot Chip Select	Chip select for external boot start-up ROM
DATA[15:0]	Data Bus	16-bit data bus
$\overline{DS}$	Data Strobe	During a read cycle, indicates when it is possible for an external device to place data on the data bus. During a write cycle, indicates that valid data is on the data bus.
$\overline{DSACK}$[1:0]	Data and Size Acknowledge	Provide asynchronous data transfers and dynamic bus sizing
DSI, DSO, DSCLK	Development Serial In, Out, Clock	Serial I/O and clock for background debugging mode
ECLK	E-Clock	External M6800 bus clock output
EXTAL, XTAL	Crystal Oscillator	Connections for clock synthesizer circuit reference; a crystal or an external oscillator can be used
FC[2:0]	Function Codes	Identify processor state and current address space
FREEZE	Freeze	Indicates that the CPU has entered background mode
$\overline{HALT}$	Halt	Suspend external bus activity
$\overline{IFETCH}$	Instruction Fetch	Identifies bus cycles in which operand is loaded into pipeline
$\overline{IPIPE}$	Instruction Pipeline	Indicates instruction pipeline activity
$\overline{IRQ}$[7:1]	Interrupt Request Level	Provide prioritized interrupts to the CPU
MODCLK	Clock Mode Select	Selects the source and type of system clock
PADA[6:0]	Port ADA	ADC digital input port signals
PC[6:0]	Port C	SIM digital output port signals
PE[7:0]	Port E	SIM digital I/O port signals
PF[7:0]	Port F	SIM digital I/O port signals
QUOT	Quotient Out	Provides the quotient bit of the polynomial divider
$\overline{RESET}$	Reset	System reset
$\overline{RMC}$	Read-Modify-Write Cycle	Indicates an indivisible read-modify-write instruction
R/$\overline{W}$	Read/Write	Indicates the direction of data transfer on the bus
SIZ[1:0]	Size	Indicates the number of bytes to be transferred during a bus cycle
TPUCH[15:0]	TPU I/O Channels	Bidirectional TPU channels
TSC	Three-State Control	Places all output drivers in a high-impedance state
T2CLK	TCR2 Clock	TPU clock input
$V_{RH, VRL}$	ADC Reference Voltage	Provide precise reference for A/D conversion
XFC	External Filter Capacitor	Connection for external phase-locked loop filter capacitor

3.2.2 System Protection Control Register

The system protection control register controls system monitor functions, software watchdog clock prescaling, and bus monitor timing. This register can be written only once following power-on or reset, but can be read at any time.

SYPCR — System Protection Control Register **$YFFA21**

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NOT USED								SWE	SWP	SWT		HME	BME	BMT	
RESET:								1	$\overline{\text{MODCLK}}$	0	0	0	0	0	0

SWE — Software Watchdog Enable
- 0 = Software watchdog disabled
- 1 = Software watchdog enabled

SWP — Software Watchdog Prescale

This bit controls the value of the software watchdog prescaler.
- 0 = Software watchdog clock not prescaled
- 1 = Software watchdog clock prescaled by 512

SWT[1:0] — Software Watchdog Timing

This field selects the divide ratio used to establish software watchdog time-out period. The following table gives the ratio for each combination of SWP and SWT bits.

SWP	SWT	Ratio
0	00	2^9
0	01	2^{11}
0	10	2^{13}
0	11	2^{15}
1	00	2^{18}
1	01	2^{20}
1	10	2^{22}
1	11	2^{24}

HME — Halt Monitor Enable
- 0 = Disable halt monitor function
- 1 = Enable halt monitor function

BME — Bus Monitor External Enable
- 0 = Disable bus monitor function for an internal to external bus cycle.
- 1 = Enable bus monitor function for an internal to external bus cycle.

BMT[1:0] — Bus Monitor Timing

This field selects a bus monitor time-out period as shown in the following table.

BMT	Bus Monitor Time-out Period
00	64 System Clocks
01	32 System Clocks
10	16 System Clocks
11	8 System Clocks

3.3.4 Low-Power Operation

Low-power operation is initiated by the CPU32. To reduce power consumption selectively, the CPU can set the STOP bits in each module configuration register. To minimize overall microcontroller power consumption, the CPU can execute the LPSTOP instruction, which causes the SIM to turn off the system clock.

When individual module STOP bits are set, clock signals inside each module are turned off, but module registers are still accessible.

When the CPU executes LPSTOP, a special CPU space bus cycle writes a copy of the current interrupt mask into the clock control logic. The SIM brings the MCU out of low-power operation when either an interrupt of higher priority than the stored mask or a reset occurs.

During a low-power stop, unless the system clock signal is supplied by an external source and that source is removed, the SIM clock control logic and the SIM clock signal (SIMCLK) continue to operate. The periodic interrupt timer and input logic for the $\overline{\text{RESET}}$ and $\overline{\text{IRQ}}$ pins are clocked by SIMCLK. The SIM can also continue to generate the CLKOUT signal while in low-power mode.

The stop mode system integration module clock (STSIM) and stop mode external clock (STEXT) bits in SYNCR determine clock operation during low-power stop. The table below summarizes the effects of STSIM and STEXT. MODCLK value is the logic level on the MODCLK pin during the last reset before LPSTOP execution. Any clock in the off state is held low. If the synthesizer VCO is turned off during LPSTOP, there is a PLL relock delay after the VCO is turned back on.

Table 8 Clock Control

Mode	Pins		SYNCR Bits		Clock Status		
LPSTOP	**MODCLK**	**EXTAL**	**STSIM**	**STEXT**	**SIMCLK**	**CLKOUT**	**ECLK**
No	0	External Clock	X	X	External Clock	External Clock	External Clock
Yes	0	External Clock	0	0	External Clock	Off	Off
Yes	0	External Clock	0	1	External Clock	External Clock	External Clock
Yes	0	External Clock	1	0	External Clock	Off	Off
Yes	0	External Clock	1	1	External Clock	External Clock	External Clock
No	1	Crystal or Reference	X	X	VCO	VCO	VCO
Yes	1	Crystal or Reference	0	0	Crystal or Reference	Off	Off
Yes	1	Crystal or Reference	0	1	Crystal or Reference	Crystal/ Reference	Off
Yes	1	Crystal or Reference	1	0	VCO	Off	Off
Yes	1	Crystal or Reference	1	1	VCO	VCO	VCO

3.3.5 Loss of Reference Signal

The state of the reset enable (RSTEN) bit in SYNCR determines what happens when clock logic detects a reference failure.

When RSTEN is cleared (default state out of reset), the clock synthesizer is forced into an operating condition referred to as limp mode. Limp mode frequency varies from device to device, but maxi-

Table 13 Chip Select Allocation

Pin	Chip Select	Discrete Outputs
$\overline{\text{CSBOOT}}$	$\overline{\text{CSBOOT}}$	—
$\overline{\text{BR}}$	$\overline{\text{CS0}}$	—
$\overline{\text{BG}}$	$\overline{\text{CS1}}$	—
$\overline{\text{BGACK}}$	$\overline{\text{CS2}}$	—
FC0	$\overline{\text{CS3}}$	PC0
FC1	$\overline{\text{CS4}}$	PC1
FC2	$\overline{\text{CS5}}$	PC2
ADDR19	$\overline{\text{CS6}}$	PC3
ADDR20	$\overline{\text{CS7}}$	PC4
ADDR21	$\overline{\text{CS8}}$	PC5
ADDR22	$\overline{\text{CS9}}$	PC6
ADDR23	$\overline{\text{CS10}}$	ECLK

3.5.1 Chip-Select Registers

Each chip-select pin can have one or more functions. Chip-select pin assignment registers (CSPAR[0:1]) determine functions of the pins. Pin assignment registers also determine port size (8- or 16-bit) for dynamic bus allocation. A pin data register (PORTC) latches data for chip-select pins that are used for discrete output.

Blocks of addresses are assigned to each chip-select function. Block sizes of two Kbytes to one Mbyte can be selected by writing values to the appropriate base address register (CSBAR[0:10], CSBARBT). Address blocks for separate chip-select functions can overlap.

Chip select option registers (CSOR[0:10], CSORBT) determine timing of and conditions for assertion of chip-select signals. Eight parameters, including operating mode, access size, synchronization, and wait state insertion can be specified.

Initialization software usually resides in a peripheral memory device controlled by the chip-select circuits. A set of special chip-select functions and registers (CSORBT, CSBARBT) is provided to support bootstrap operation.

3.5.2 Pin Assignment Registers

The pin assignment registers contain twelve 2-bit fields ($\overline{\text{CS}}$[10:0] and $\overline{\text{CSBOOT}}$) that determine functions of the chip-select pins. Each pin has two or three possible functions, as shown below.

Table 14 Chip-Select Pin Functions

Assignment Register	16-Bit Chip Select	8-Bit Chip Select	Alternate Function	Discrete Output
CSPAR0	$\overline{\text{CSBOOT}}$	$\overline{\text{CSBOOT}}$	$\overline{\text{CSBOOT}}$	—
	$\overline{\text{CS0}}$	$\overline{\text{CS0}}$	$\overline{\text{BR}}$	—
	$\overline{\text{CS1}}$	$\overline{\text{CS1}}$	$\overline{\text{BG}}$	—
	$\overline{\text{CS2}}$	$\overline{\text{CS2}}$	$\overline{\text{BGACK}}$	—
	$\overline{\text{CS3}}$	$\overline{\text{CS3}}$	FC0	PC0
	$\overline{\text{CS4}}$	$\overline{\text{CS4}}$	FC1	PC1
	$\overline{\text{CS5}}$	$\overline{\text{CS5}}$	FC2	PC2
CSPAR1	$\overline{\text{CS6}}$	$\overline{\text{CS6}}$	ADDR19	PC3
	$\overline{\text{CS7}}$	$\overline{\text{CS7}}$	ADDR20	PC4
	$\overline{\text{CS8}}$	$\overline{\text{CS8}}$	ADDR21	PC5
	$\overline{\text{CS9}}$	$\overline{\text{CS9}}$	ADDR22	PC6
	$\overline{\text{CS10}}$	$\overline{\text{CS10}}$	ADDR23	ECLK

DSACK	Description
0000	No Wait States
0001	1 Wait State
0010	2 Wait States
0011	3 Wait States
0100	4 Wait States
0101	5 Wait States
0110	6 Wait States
0111	7 Wait States
1000	8 Wait States
1001	9 Wait States
1010	10 Wait States
1011	11 Wait States
1100	12 Wait States
1101	13 Wait States
1110	Fast Termination
1111	External DSACK

SPACE — Address Space

Use this option field to select an address space for the chip-select logic. The CPU32 normally operates in supervisor or user space, but interrupt acknowledge cycles must take place in CPU space.

Space	Address Space
00	CPU Space
01	User Space
10	Supervisor Space
11	Supervisor/User Space

IPL — Interrupt Priority Level

If the space field is set for CPU space (00), chip-select logic can be used for interrupt acknowledge. During an interrupt acknowledge cycle, the priority level on address lines ADDR[3:1] is compared to the value in the IPL field. If the values are the same, a chip select is asserted, provided that other option register conditions are met. The following table shows IPL field encoding.

IPL	Description
000	Any Level
001	IPL1
010	IPL2
011	IPL3
100	IPL4
101	IPL5
110	IPL6
111	IPL7

This field only affects the response of chip selects and does not affect interrupt recognition by the CPU. Any level means that chip select is asserted regardless of the level of the interrupt acknowledge cycle.

AVEC — Autovector Enable

0 = External interrupt vector enabled

1 = Autovector enabled

This field selects one of two methods of acquiring an interrupt vector number during an external interrupt acknowledge cycle.

When an external device asserts $\overline{\text{RESET}}$ for the proper period, reset control logic clocks the signal into an internal latch. The control logic drives the $\overline{\text{RESET}}$ pin low for an additional 512 CLKOUT cycles after it detects that the $\overline{\text{RESET}}$ signal is no longer being externally driven, to guarantee this length of reset to the entire system.

If an internal source asserts a reset signal, the reset control logic asserts $\overline{\text{RESET}}$ for a minimum of 512 cycles. If the reset signal is still asserted at the end of 512 cycles, the control logic continues to assert $\overline{\text{RESET}}$ until the internal reset signal is negated.

After 512 cycles have elapsed, the reset input pin goes to an inactive, high-impedance state for 10 cycles. At the end of this 10-cycle period, the reset input is tested. When the input is at logic level one, reset exception processing begins. If, however, the reset input is at logic level zero, the reset control logic drives the pin low for another 512 cycles. At the end of this period, the pin again goes to high-impedance state for ten cycles, then it is tested again. The process repeats until $\overline{\text{RESET}}$ is released.

3.7.4 Power-On Reset

When the SIM clock synthesizer is used to generate the system clock, power-on reset involves special circumstances related to application of system and clock synthesizer power. Regardless of clock source, voltage must be applied to clock synthesizer power input pin V_{DDSYN} in order for the MCU to operate. The following discussion assumes that V_{DDSYN} is applied before and during reset. This minimizes crystal start-up time. When V_{DDSYN} is applied at power-on, start-up time is affected by specific crystal parameters and by oscillator circuit design. V_{DD} ramp-up time also affects pin state during reset.

During power-on reset, an internal circuit in the SIM drives the internal IMB and external reset lines. The circuit releases the internal reset line as V_{DD} ramps up to the minimum specified value, and SIM pins are initialized. As V_{DD} reaches specified minimum value, the clock synthesizer VCO begins operation and clock frequency ramps up to specified limp mode frequency. The external $\overline{\text{RESET}}$ line remains asserted until the clock synthesizer PLL locks and 512 CLKOUT cycles elapse.

The SIM clock synthesizer provides clock signals to the other MCU modules. After the clock is running and the internal reset signal is asserted for four clock cycles, these modules reset. V_{DD} ramp time and VCO frequency ramp time determine how long these four cycles take. Worst case is approximately 15 milliseconds. During this period, module port pins may be in an indeterminate state. While input-only pins can be put in a known state by means of external pull-up resistors, external logic on input/output or output-only pins must condition the lines during this time. Active drivers require high-impedance buffers or isolation resistors to prevent conflict.

3.7.5 Use of Three State Control Pin

Asserting the three-state control (TSC) input causes the MCU to put all output drivers in an inactive, high-impedance state. The signal must remain asserted for ten clock cycles in order for drivers to change state. There are certain constraints on use of TSC during power-on reset:

When the internal clock synthesizer is used (MODCLK held high during reset), synthesizer ramp-up time affects how long the ten cycles take. Worst case is approximately 20 ms from TSC assertion.

When an external clock signal is applied (MODCLK held low during reset), pins go to high-impedance state as soon after TSC assertion as ten clock pulses have been applied to the EXTAL pin.

When TSC assertion takes effect, internal signals are forced to values that can cause inadvertent mode selection. Once the output drivers change state, the MCU must be powered down and restarted before normal operation can resume.

3.8 Interrupts

Interrupt recognition and servicing involve complex interaction between the central processing unit, the system integration module, and a device or module requesting interrupt service.

The CPU32 provides eight levels of interrupt priority. All interrupts with priorities less than seven can be masked by the interrupt priority (IP) field in status register.

There are seven interrupt request signals ($\overline{IRQ}[7:1]$). These signals are used internally on the IMB, and there are corresponding pins for external interrupt service requests. The CPU treats all interrupt requests as though they come from internal modules — external interrupt requests are treated as interrupt service requests from the SIM. Each of the interrupt request signals corresponds to an interrupt priority level. $\overline{IRQ1}$ has the lowest priority and $\overline{IRQ7}$ the highest.

Interrupt recognition is determined by interrupt priority level and interrupt priority mask value. The interrupt priority mask consists of three bits in the CPU32 status register. Binary values %000 to %111 provide eight priority masks. Masks prevent an interrupt request of a priority less than or equal to the mask value from being recognized and processed. $\overline{IRQ7}$, however, is always recognized, even if the mask value is %111.

$\overline{IRQ}[7:1]$ are active-low level-sensitive inputs. The low on the pin must remain asserted until an interrupt acknowledge cycle corresponding to that level is detected.

$\overline{IRQ7}$ is transition-sensitive as well as level-sensitive: a level-7 interrupt is not detected unless a falling edge transition is detected on the $\overline{IRQ7}$ line. This prevents redundant servicing and stack overflow. A nonmaskable interrupt is generated each time $\overline{IRQ7}$ is asserted as well as each time the priority mask changes from%111 to a lower number while $\overline{IRQ7}$ is asserted.

Interrupt requests are sampled on consecutive falling edges of the system clock. Interrupt request input circuitry has hysteresis: to be valid, a request signal must be asserted for at least two consecutive clock periods. Valid requests do not cause immediate exception processing, but are left pending. Pending requests are processed at instruction boundaries or when exception processing of higher-priority exceptions is complete.

The CPU32 does not latch the priority of a pending interrupt request. If an interrupt source of higher priority makes a service request while a lower priority request is pending, the higher priority request is serviced. If an interrupt request with a priority equal to or lower than the current IP mask value is made, the CPU32 does not recognize the occurrence of the request. If simultaneous interrupt requests of different priorities are made, and both have a priority greater than the mask value, the CPU32 recognizes the higher-level request.

3.8.1 Interrupt Acknowledge and Arbitration

When the CPU32 detects one or more interrupt requests of a priority higher than the interrupt priority mask value, it places the interrupt request level on the address bus and initiates a CPU space read cycle. The request level serves two purposes: it is decoded by modules or external devices that have requested interrupt service, to determine whether the current interrupt acknowledge cycle pertains to them, and it is latched into the interrupt priority mask field in the CPU32 status register, to preclude further interrupts of lower priority during interrupt service.

Modules or external devices that have requested interrupt service must decode the interrupt priority mask value placed on the address bus during the interrupt acknowledge cycle and respond if the priority of the service request corresponds to the mask value. However, before modules or external devices respond, interrupt arbitration takes place.

Arbitration is performed by means of serial contention between values stored in individual module interrupt arbitration (IARB) fields. Each module that can make an interrupt service request, including the SIM, has an IARB field in its configuration register. IARB fields can be assigned values from %0000 to

3. The request level is latched from the address bus into the interrupt priority mask field in the status or condition code register.

D. Modules that have requested interrupt service decode the priority value in ADDR[3:1]. If request priority is the same as acknowledged priority, arbitration by IARB contention takes place.

E. After arbitration, the interrupt acknowledge cycle is completed in one of the following ways:
   1. When there is no contention (IARB = %0000), the spurious interrupt monitor asserts $\overline{\text{BERR}}$, and the CPU generates the spurious interrupt vector number.
   2. The dominant interrupt source supplies a vector number and $\overline{\text{DSACK}}$ signals appropriate to the access. The CPU acquires the vector number.
   3. The $\overline{\text{AVEC}}$ signal is asserted (the signal can be asserted by the dominant interrupt source or the pin can be tied low), and the CPU generates an autovector number corresponding to interrupt priority.
   4. The bus monitor asserts $\overline{\text{BERR}}$ and the CPU32 generates the spurious interrupt vector number.

F. The vector number is converted to a vector address.

G. The content of the vector address is loaded into the PC, and the processor transfers control to the exception handler routine.

3.9 Factory Test Block

The test submodule supports scan-based testing of the various MCU modules. It is integrated into the SIM to support production testing.

Test submodule registers are intended for Freescale use. Register names and addresses are provided to indicate that these addresses are occupied.

SIMTR — System Integration Test Register **$YFFA02**

SIMTRE — System Integration Test Register (E Clock) **$YFFA08**

TSTMSRA — Master Shift Register A **$YFFA30**

TSTMSRB — Master Shift Register B **$YFFA32**

TSTSC — Test Module Shift Count **$YFFA34**

TSTRC — Test Module Repetition Count **$YFFA36**

CREG — Test Module Control Register **$YFFA38**

DREG — Test Module Distributed Register **$YFFA3A**

Table 23 Instruction Set Summary

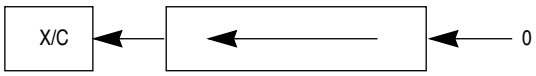
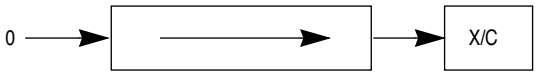
Instruction	Syntax	Operand Size	Operation
DBcc	Dn, ⟨label⟩	16	If condition false, then Dn – 1 ⇒ PC; if Dn ≠ (– 1), then PC + d ⇒ PC
DIVS/DIVU	⟨ea⟩, Dn	32/16 ⇒ 16: 16	Destination / Source ⇒ Destination (signed or unsigned)
DIVSL/DIVUL	⟨ea⟩, Dr: Dq ⟨ea⟩, Dq ⟨ea⟩, Dr : Dq	64/32 ⇒ 32 : 32 32/32 ⇒ 32 32/32 ⇒ 32 : 32	Destination / Source ⇒ Destination (signed or unsigned)
EOR	Dn, ⟨ea⟩	8, 16, 32	Source ⊕ Destination ⇒ Destination
EORI	#⟨data⟩, ⟨ea⟩	8, 16, 32	Data ⊕ Destination ⇒ Destination
EORI to CCR	#⟨data⟩, CCR	8	Source ⊕ CCR ⇒ CCR
EORI to SR[1]	#⟨data⟩, SR	16	Source ⊕ SR ⇒ SR
EXG	Rn, Rn	32	Rn ⇒ Rn
EXT	Dn Dn	8 ⇒ 16 16 ⇒ 32	Sign extended Destination ⇒ Destination
EXTB	Dn	8 ⇒ 32	Sign extended Destination ⇒ Destination
ILLEGAL	none	none	SSP – 2 ⇒ SSP; vector offset ⇒ (SSP); SSP – 4 ⇒ SSP; PC ⇒ (SSP); SSP – 2 ⇒ SSP; SR ⇒ (SSP); illegal instruction vector address ⇒ PC
JMP	⟨ea⟩	none	Destination ⇒ PC
JSR	⟨ea⟩	none	SP – 4 ⇒ SP; PC ⇒ (SP); destination ⇒ PC
LEA	⟨ea⟩, An	32	⟨ea⟩ ⇒ An
LINK	An, #⟨d⟩	16, 32	SP – 4 ⇒ SP, An ⇒ (SP); SP ⇒ An, SP + d ⇒ SP
LPSTOP[1]	#⟨data⟩	none	Data ⇒ SR; interrupt mask ⇒ EBI; STOP
LSL	Dn, Dn #⟨data⟩, Dn ⟨ea⟩	8, 16, 32 8, 16, 32 16	
LSR	Dn, Dn #⟨data⟩, Dn ⟨ea⟩	8, 16, 32 8, 16, 32 16	
MOVE	⟨ea⟩, ⟨ea⟩	8, 16, 32	Source ⇒ Destination
MOVEA	⟨ea⟩, An	16, 32 ⇒ 32	Source ⇒ Destination
MOVEA[1]	USP, An An, USP	32 32	USP ⇒ An An ⇒ USP
MOVE from CCR	CCR, ⟨ea⟩	16	CCR ⇒ Destination
MOVE to CCR	⟨ea⟩, CCR	16	Source ⇒ CCR
MOVE from SR[1]	SR, ⟨ea⟩	16	SR ⇒ Destination
MOVE to SR[1]	⟨εα⟩, SR	16	Source ⇒ SR
MOVE USP[1]	USP, An An, USP	32 32	USP ⇒ An An ⇒ USP
MOVEC[1]	Rc, Rn Rn, Rc	32 32	Rc ⇒ Rn Rn ⇒ Rc
MOVEM	list, ⟨ea⟩ ⟨ea⟩, list	16, 32 16, 32 ⇒ 32	Listed registers ⇒ Destination Source ⇒ Listed registers
MOVEP	Dn, (d_{16}, An) (d_{16}, An), Dn	16, 32	Dn [31:24] ⇒ (An + d); Dn [23:16] ⇒ (An + d + 2); Dn [15:8] ⇒ (An + d + 4); Dn [7:0] ⇒ (An + d + 6) (An + d) ⇒ Dn [31:24]; (An + d + 2) ⇒ Dn [23:16]; (An + d + 4) ⇒ Dn [15:8]; (An + d + 6) ⇒ Dn [7:0]

Table 24 Background Debugging Command Summary

Command	Mnemonic	Description
Read D/A Register	RDREG/RAREG	Read the selected address or data register and return the results through the serial interface.
Write D/A Register	WDREG/WAREG	The data operand is written to the specified address or data register.
Read System Register	RSREG	The specified system control register is read. All registers that can be read in supervisor mode can be read in background mode.
Write System Register	WSREG	The operand data is written into the specified system control register.
Read Memory Location	READ	Read the sized data at the memory location specified by the long-word address. The source function code register (SFC) determines the address space accessed.
Write Memory Location	WRITE	Write the operand data to the memory location specified by the long-word address. The destination function code (DFC) register determines the address space accessed.
Dump Memory Block	DUMP	Used in conjunction with the READ command to dump large blocks of memory. An initial READ is executed to set up the starting address of the block and retrieve the first result. Subsequent operands are retrieved with the DUMP command.
Fill Memory Block	FILL	Used in conjunction with the WRITE command to fill large blocks of memory. Initially, a WRITE is executed to set up the starting address of the block and supply the first operand. The FILL command writes subsequent operands.
Resume Execution	GO	The pipe is flushed and refilled before resuming instruction execution at the current PC.
Patch User Code	CALL	Current program counter is stacked at the location of the current stack pointer. Instruction execution begins at user patch code.
Reset Peripherals	RST	Asserts RESET for 512 clock cycles. The CPU is not reset by this command. Synonymous with the CPU RESET instruction.
No Operation	NOP	NOP performs no operation and can be used as a null command.

Table 25 TPU Address Map

Access	Address	15 8	7 0
S	$YFFE00	TPU MODULE CONFIGURATION REGISTER (TPUMCR)	
S	$YFFE02	TEST CONFIGURATION REGISTER (TCR)	
S	$YFFE04	DEVELOPMENT SUPPORT CONTROL REGISTER (DSCR)	
S	$YFFE06	DEVELOPMENT SUPPORT STATUS REGISTER (DSSR)	
S	$YFFE08	TPU INTERRUPT CONFIGURATION REGISTER (TICR)	
S	$YFFE0A	CHANNEL INTERRUPT ENABLE REGISTER (CIER)	
S	$YFFE0C	CHANNEL FUNCTION SELECTION REGISTER 0 (CFSR0)	
S	$YFFE0E	CHANNEL FUNCTION SELECTION REGISTER 1 (CFSR1)	
S	$YFFE10	CHANNEL FUNCTION SELECTION REGISTER 2 (CFSR2)	
S	$YFFE12	CHANNEL FUNCTION SELECTION REGISTER 3 (CFSR3)	
S/U	$YFFE14	HOST SEQUENCE REGISTER 0 (HSQR0)	
S/U	$YFFE16	HOST SEQUENCE REGISTER 1 (HSQR1)	
S/U	$YFFE18	HOST SERVICE REQUEST REGISTER 0 (HSRR0)	
S/U	$YFFE1A	HOST SERVICE REQUEST REGISTER 1 (HSRR1)	
S	$YFFE1C	CHANNEL PRIORITY REGISTER 0 (CPR0)	
S	$YFFE1E	CHANNEL PRIORITY REGISTER 1 (CPR1)	
S	$YFFE20	CHANNEL INTERRUPT STATUS REGISTER (CISR)	
S	$YFFE22	LINK REGISTER (LR)	
S	$YFFE24	SERVICE GRANT LATCH REGISTER (SGLR)	
S	$YFFE26	DECODED CHANNEL NUMBER REGISTER (DCNR)	

Y = M111, where M represents the logic state of the MM bit in the SIMCR.

5.3 TPU Components

The TPU module consists of two 16-bit time bases, sixteen independent timer channels, a task scheduler, a microengine, and a host interface. In addition, a dual-port parameter RAM is used to pass parameters between the module and the host CPU.

5.3.1 Time Bases

Two 16-bit counters provide reference time bases for all output compare and input capture events. Prescalers for both time bases are controlled by the host CPU via bit fields in the TPU module configuration register (TPUMCR). Timer count registers TCR1 and TCR2 provide access to current counter values. TCR1 and TCR2 can be read/write accessed in microcode, but are not directly available to the host CPU. The TCR1 clock is derived from the system clock. The TCR2 clock can be derived from the system clock or from an external clock input via the T2CLK pin.

5.3.2 Timer Channels

The TPU has 16 independent channels, each connected to an MCU pin. The channels have identical hardware. Each channel consists of an event register and pin control logic. The event register contains a 16-bit capture register, a 16-bit compare/match register, and a 16-bit greater-than-or-equal-to comparator. The direction of each pin, either output or input, is determined by the TPU microengine. Each channel can either use the same time base for match and capture, or can use one time base for match and the other for capture.

5.6.2 Input Capture/Transition Counter (ITC)

Any channel of the TPU can capture the value of a specified TCR or any specified location in parameter RAM upon the occurrence of each transition or specified number of transitions, and then generate an interrupt request to notify the bus master. The times of the most recent two transitions are maintained in parameter RAM. A channel can perform input captures continually, or a channel can detect a single transition or specified number of transitions, ceasing channel activity until reinitialization. After each transition or specified number of transitions, the channel can generate a link to other channels.

5.6.3 Queued Output Match (QOM)

QOM can generate single or multiple output match events from a table of offsets in parameter RAM. Loop modes allow complex pulse trains to be generated once, a specified number of times, or continuously. The function can be triggered by a link from another TPU channel. In addition, the reference time for the sequence of matches can be obtained from another channel. QOM can generate pulse-width modulated waveforms, including waveforms with high times of 0% or 100%. QOM also allows a TPU channel to be used as a discrete output pin.

5.6.4 Programmable Time Accumulator (PTA)

PTA accumulates a 32-bit sum of the total high time, low time, or period of an input signal over a programmable number of periods or pulses. The accumulation can start on a rising or falling edge. After the specified number of periods or pulses, the PTA generates an interrupt request and optionally generates links to other channels.

From 1 to 255 period measurements can be made and summed with the previous measurement(s) before the TPU interrupts the CPU, providing instantaneous or average frequency measurement capability, and the latest complete accumulation (over the programmed number of periods).

5.6.5 Multichannel Pulse-Width Modulation (MCPWM)

MCPWM generates pulse-width modulated outputs with full 0% to 100% duty cycle range independent of other TPU activity. This capability requires two TPU channels plus an external gate for one PWM channel. (A simple one-channel PWM capability is supported by the QOM function.)

Multiple PWMs generated by MCPWM have two types of high time alignment: edge aligned and center aligned. Edge aligned mode uses n + 1 TPU channels for n PWMs; center aligned mode uses 2n + 1 channels. Center aligned mode allows a user defined ‘dead time’ to be specified so that two PWMs can be used to drive an H-bridge without destructive current spikes. This feature is important for motor control applications.

5.6.6 Fast Quadrature Decode (FQD)

FQD is a position feedback function for motor control. It decodes the two signals from a slotted encoder to provide the CPU with a 16-bit free running position counter. FQD incorporates a “speed switch” which disables one of the channels at high speed, allowing faster signals to be decoded. A time stamp is provided on every counter update to allow position interpolation and better velocity determination at low speed or when low resolution encoders are used. The third index channel provided by some encoders is handled by the ITC function.

5.6.7 Universal Asynchronous Receiver/Transmitter (UART)

The UART function uses one or two TPU channels to provide asynchronous communications. Data word length is programmable from 1 to 14 bits. The function supports detection or generation of even, odd, and no parity. Baud rate is freely programmable and can be higher than 100 Kbaud. Eight bidirectional UART channels running in excess of 9600 baud could be implemented on the TPU.

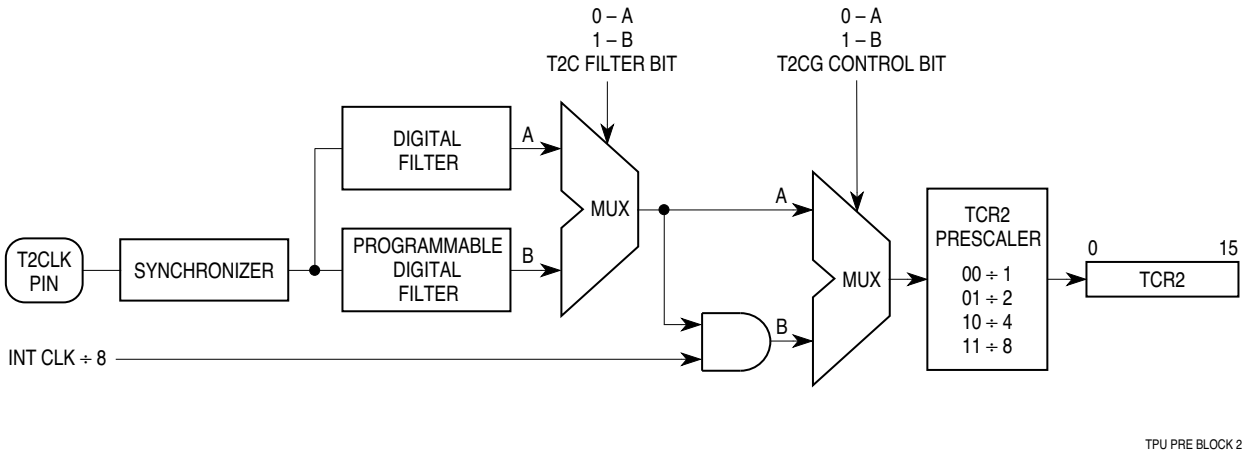


Figure 13 Prescaler Control 2

STF — Stop Flag
0 = TPU operating
1 = TPU stopped (STOP bit has been asserted)

SUPV — Supervisor Data Space
0 = Assignable registers are unrestricted (FC2 is ignored)
1 = Assignable registers are restricted (FC2 is decoded)

PSCK — Prescaler Clock
0 = System clock/32 is input to TCR1 prescaler
1 = System clock/4 is input to TCR1 prescaler

IARB — Interrupt Arbitration Identification Number
The IARB field is used to arbitrate between simultaneous interrupt requests of the same priority. Each module that can generate interrupt requests must be assigned a unique, nonzero IARB field value. Refer to **3.8 Interrupts** for more information.

TICR — TPU Interrupt Configuration Register **$YFFE08**

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NOT USED					CIRL			CIBV				NOT USED			
RESET:					0	0	0	0	0	0	0				

CIRL — Channel Interrupt Request Level
The interrupt request level for all channels is specified by this 3-bit encoded field. Level seven for this field indicates a nonmaskable interrupt; level zero indicates that all channel interrupts are disabled.

CIBV — Channel Interrupt Base Vector
The TPU is assigned 16 unique interrupt vector numbers, one vector number for each channel. The CIBV field specifies the most significant nibble of all 16 TPU channel interrupt vector numbers. The lower nibble of the TPU interrupt vector number is determined by the channel number on which the interrupt occurs.

ADCTL0 — A/D Control Register 0 **$YFF70A**

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NOT USED								RES10	STS		PRS				
RESET:								0	0	0	0	0	0	1	1

ADCTL0 is used to select ADC clock source and to set up prescaling. Writes to it have immediate effect.

RES10 — 10-Bit Resolution

0 = 8-bit resolution

1 = 10-bit resolution

Conversion results are appropriately aligned in result registers to reflect conversion status.

STS[1:0] — Sample Time Select

Total conversion time depends on initial sample time, transfer time, final sample time, and resolution time. Initial sample time is fixed at two clocks. Transfer time is fixed at two clocks. Resolution time is fixed at ten ADC clock cycles for an 8-bit conversion and twelve ADC clock cycles for a 10-bit conversion. Final sample time depends on the STS field, as shown below.

STS[1:0]	Sample Time
00	2 A/D Clock Periods
01	4 A/D Clock Periods
10	8 A/D Clock Periods
11	16 A/D Clock Periods

PRS[4:0] — Prescaler Rate Selection Field

ADC clock is generated from system clock using a modulo counter and a divide-by-two circuit. The binary value of this field is the counter modulus. System clock is divided by the PRS value plus one, then sent to the divide-by-two circuit, as shown in the following table. Maximum ADC clock rate is 2 MHz. Reset value of PRS is a divisor value of eight, resulting in a nominal 2-MHz ADC clock.

PRS[4:0]	Divisor Value
00000	4
00001	4
00010	6
...	...
11101	60
11110	62
11111	64

The following table summarizes the operation of S8CM and CD:CA when MULT is set (multichannel mode). Number of conversions per channel is determined by SCAN. Channel numbers are given in order of conversion.

S8CM	CD	CC	CB	CA	Input	Result Register
0	0	0	X	X	AN0	RSLT0
					AN1	RSLT1
					AN2	RSLT2
					AN3	RSLT3
0	0	1	X	X	AN4	RSLT0
					AN5	RSLT1
					AN6	RSLT2
					AN7*	RSLT3
0	1	0	X	X	Reserved	RSLT0
					Reserved	RSLT1
					Reserved	RSLT2
					Reserved	RSLT3
0	1	1	X	X	V_{RH}	RSLT0
					V_{RL}	RSLT1
					$(V_{RH} - V_{RL}) / 2$	RSLT2
					Test/Reserved	RSLT3
1	0	X	X	X	AN0	RSLT0
					AN1	RSLT1
					AN2	RSLT2
					AN3	RSLT3
					AN4	RSLT4
					AN5	RSLT5
					AN6	RSLT6
					AN7*	RSLT7
1	1	X	X	X	Reserved	RSLT0
					Reserved	RSLT1
					Reserved	RSLT2
					Reserved	RSLT3
					V_{RH}	RSLT4
					V_{RL}	RSLT5
					$(V_{RH} - V_{RL}) / 2$	RSLT6
					Test/Reserved	RSLT7

* Since the ADC in the MCU has only seven external analog inputs, AN7 is connected to V_{SSA}.

ADSTAT — ADC Status Register **$YFF70E**

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SCF	NOT USED				CCTR			CCF							
RESET:															
0					0	0	0	0	0	0	0	0	0	0	0

ADSTAT contains information related to the status of a conversion sequence.

SCF — Sequence Complete Flag

0 = Sequence not complete

1 = Sequence complete

SCF is set at the end of the conversion sequence when SCAN is cleared, and at the end of the first conversion sequence when SCAN is set. SCF is cleared when ADCTL1 is written and a new conversion sequence begins.

CCTR[2:0] — Conversion Counter Field

This field reflects the contents of the conversion counter pointer in either four or eight count conversion sequence. The value corresponds to the number of the next result register to be written, and thus indicates which channel is being converted.

CCF[7:0] — Conversion Complete Field

Each bit in this field corresponds to an A/D result register (CCF7 to RSLT7, etc.). A bit is set when conversion for the corresponding channel is complete, and remains set until the result register is read.

RSLT[7:0] — A/D Result Registers **$YFF710–$YFF73E**

The result registers store data after conversion is complete. Each register can be read from three different addresses in the register block. Data format depends on the address from which the result register is read.

RJURR — Unsigned Right-Justified Format **$YFF710–$YFF71F**

Conversion result is unsigned right-justified data. Bits [9:0] are used for 10-bit resolution, bits [7:0] are used for 8-bit resolution (bits [9:8] are zero). Bits [15:10] always return zero when read.

LJSRR — Signed Left-Justified Format **$YFF720–$YFF72F**

Conversion result is signed left-justified data. Bits [15:6] are used for 10-bit resolution, bits [15:8] are used for 8-bit resolution (bits [7:6] are zero). Although the ADC is unipolar, it is assumed that the zero point is halfway between low and high reference when this format is used. For positive input, bit 15 = 0, for negative input, bit 15 = 1. Bits [5:0] always return zero when read.

LJURR — Unsigned Left-Justified Format **$YFF730–$YFF73F**

Conversion result is unsigned left-justified data. Bits [15:6] are used for 10-bit resolution, bits [15:8] are used for 8-bit resolution (bits [7:6] are zero). Bits [5:0] always return zero when read.

STOP — Stop Control

0 = TPURAM array operates normally.

1 = TPURAM array enters low-power stop mode.

This bit controls whether the TPURAM array is in stop mode or normal operation. Reset state is zero, for normal operation. In stop mode, the array retains its contents, but cannot be read or written by the CPU.

RASP — RAM Array Space Field

0 = TPURAM array is placed in unrestricted space

1 = TPURAM array is placed in supervisor space

TRAMTST — TPURAM Test Register **$YFFB02**

TRAMTST is used for factory testing of the TPURAM module.

TRAMBAR — TPURAM Base Address and Status Register **$YFFB04**

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADDR 23	ADDR 22	ADDR 21	ADDR 20	ADDR 19	ADDR 18	ADDR 17	ADDR 16	ADDR 15	ADDR 14	ADDR 13	ADDR 12	ADDR 11	NOT USED		RAMDS
RESET:															
0	0	0	0	0	0	0	0	0	0	0	0	0			0

ADDR[23:11] — RAM Array Base Address

This field specifies ADDR[23:11] of the TPURAM array base address when the array is enabled. ADDR11 determines the 2-Kbyte boundary that the block of addresses containing the array is mapped to. The array occupies only the upper 1024 bytes of the 2-Kbyte block of addresses. When ADDR11 bit is set, valid array addresses range from $0C00 to%0FFFF; when the ADDR11 bit is cleared, valid array addresses range from $0400 to $07FF.

RAMDS — RAM Array Disable

0 = TPURAM array is enabled

1 = TPURAM array is disabled

The TPURAM array is disabled by internal logic after a master reset. Writing a valid base address to the TPURAM array base address field (bits [15:3]) automatically clears RAMDS, enabling the TPURAM array.

7.4 TPURAM Operation

There are six TPURAM operating modes, as follows:

- The TPURAM module is in normal mode when powered by V_{DD}. The array can be accessed by byte, word, or long word. A byte or aligned word (high-order byte is at an even address) access only takes one bus cycle or two system clocks. A long word or misaligned word access requires two bus cycles.
- Standby mode is intended to preserve TPURAM contents when V_{DD} is removed. TPURAM contents are maintained by V_{STBY}. Circuitry within the TPURAM module switches to the higher of V_{DD} or V_{STBY} with no loss of data. When TPURAM is powered by V_{STBY}, access to the array is not guaranteed.
- Reset mode allows the CPU to complete the current bus cycle before resetting. When a synchronous reset occurs while a byte or word TPURAM access is in progress, the access will be completed. If reset occurs during the first word access of a long-word operation, only the first word access will be completed. If reset occurs during the second word access of a long word operation, the entire access will be completed. Data being read from or written to the RAM may be corrupted by asynchronous reset.

How to Reach Us:

Home Page:
www.freescale.com

E-mail:
support@freescale.com

USA/Europe or Locations Not Listed:
Freescale Semiconductor
Technical Information Center, CH370
1300 N. Alma School Road
Chandler, Arizona 85224
+1-800-521-6274 or +1-480-768-2130
support@freescale.com

Europe, Middle East, and Africa:
Freescale Halbleiter Deutschland GmbH
Technical Information Center
Schatzbogen 7
81829 Muenchen, Germany
+44 1296 380 456 (English)
+46 8 52200080 (English)
+49 89 92103 559 (German)
+33 1 69 35 48 48 (French)
support@freescale.com

Japan:
Freescale Semiconductor Japan Ltd.
Headquarters
ARCO Tower 15F
1-8-1, Shimo-Meguro, Meguro-ku,
Tokyo 153-0064
Japan
0120 191014 or +81 3 5437 9125
support.japan@freescale.com

Asia/Pacific:
Freescale Semiconductor Hong Kong Ltd.
Technical Information Center
2 Dai King Street
Tai Po Industrial Estate
Tai Po, N.T., Hong Kong
+800 2666 8080
support.asia@freescale.com

For Literature Requests Only:
Freescale Semiconductor Literature Distribution Center
P.O. Box 5405
Denver, Colorado 80217
1-800-441-2447 or 303-675-2140
Fax: 303-675-2150
LDCForFreescaleSemiconductor@hibbertgroup.com

RoHS-compliant and/or Pb- free versions of Freescale products have the functionality and electrical characteristics of their non-RoHS-compliant and/or non-Pb- free counterparts. For further information, see http://www.freescale.com or contact your Freescale sales representative.

For information on Freescale.s Environmental Products program, go to http://www.freescale.com/epp.

freescale™
semiconductor