



Welcome to [E-XFL.COM](https://www.e-xfl.com)

What is "[Embedded - Microcontrollers](#)"?

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

Applications of "[Embedded - Microcontrollers](#)"

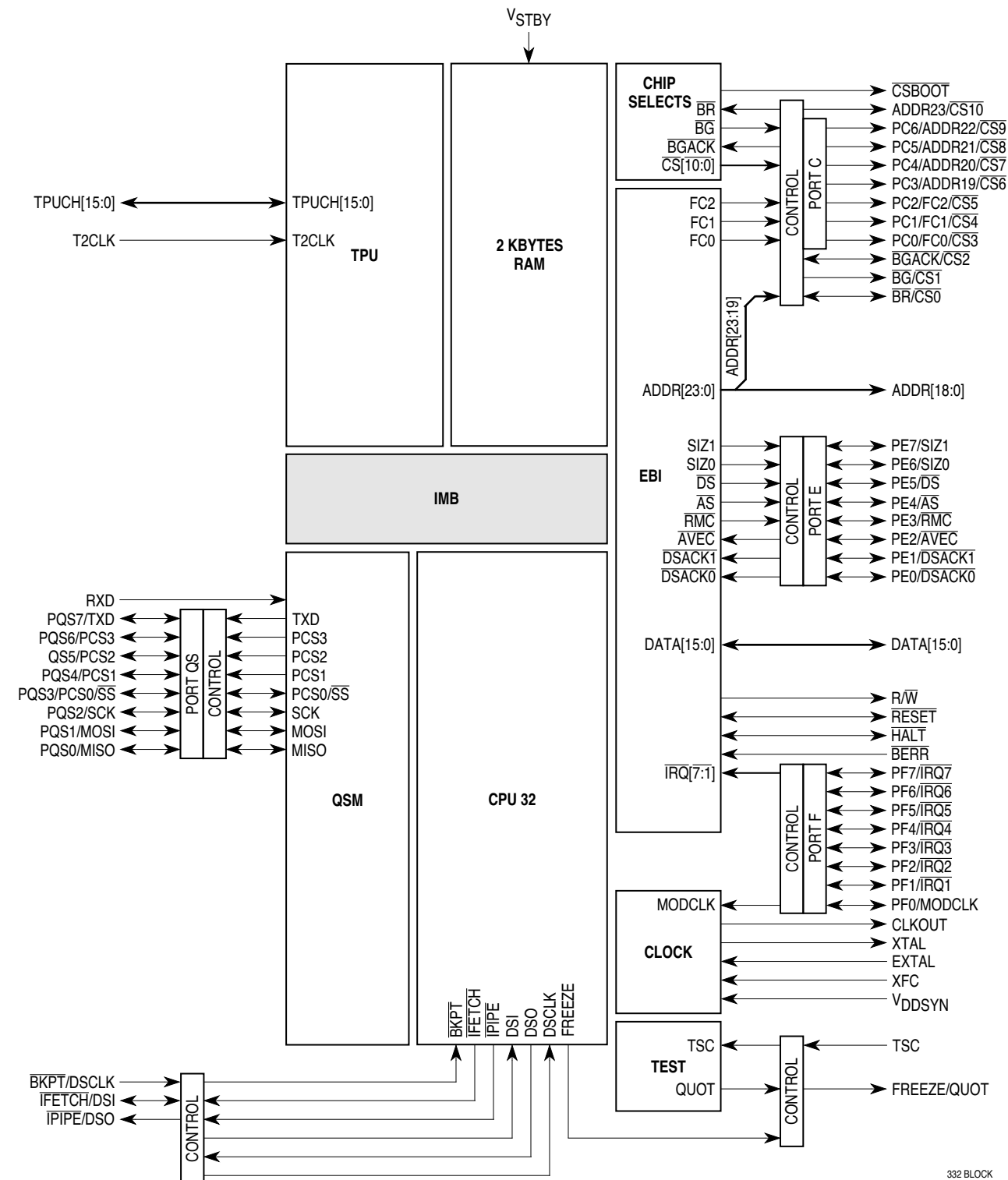
Details

Product Status	Not For New Designs
Core Processor	CPU32
Core Size	32-Bit Single-Core
Speed	25MHz
Connectivity	EBI/EMI, SCI, SPI, UART/USART
Peripherals	POR, PWM, WDT
Number of I/O	15
Program Memory Size	-
Program Memory Type	ROMless
EEPROM Size	-
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	4.5V ~ 5.5V
Data Converters	-
Oscillator Type	Internal
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/product-detail/nxp-semiconductors/mc68332aceh25

1.1 Features

- Central Processing Unit (CPU32)
 - 32-Bit Architecture
 - Virtual Memory Implementation
 - Table Lookup and Interpolate Instruction
 - Improved Exception Handling for Controller Applications
 - High-Level Language Support
 - Background Debugging Mode
 - Fully Static Operation
- System Integration Module (SIM)
 - External Bus Support
 - Programmable Chip-Select Outputs
 - System Protection Logic
 - Watchdog Timer, Clock Monitor, and Bus Monitor
 - Two 8-Bit Dual Function Input/Output Ports
 - One 7-Bit Dual Function Output Port
 - Phase-Locked Loop (PLL) Clock System
- Time Processor Unit (TPU)
 - Dedicated Microengine Operating Independently of CPU32
 - 16 Independent, Programmable Channels and Pins
 - Any Channel can Perform any Time Function
 - Two Timer Count Registers with Programmable Prescalers
 - Selectable Channel Priority Levels
- Queued Serial Module (QSM)
 - Enhanced Serial Communication Interface
 - Queued Serial Peripheral Interface
 - One 8-Bit Dual Function Port
- Static RAM Module with TPU Emulation Capability (TPURAM)
 - 2-Kbytes of Static RAM
 - May be Used as Normal RAM or TPU Microcode Emulation RAM

1.2 Block Diagram



332 BLOCK

Figure 1 MCU Block Diagram

3 System Integration Module

The MCU system integration module (SIM) consists of five functional blocks that control system start-up, initialization, configuration, and external bus.

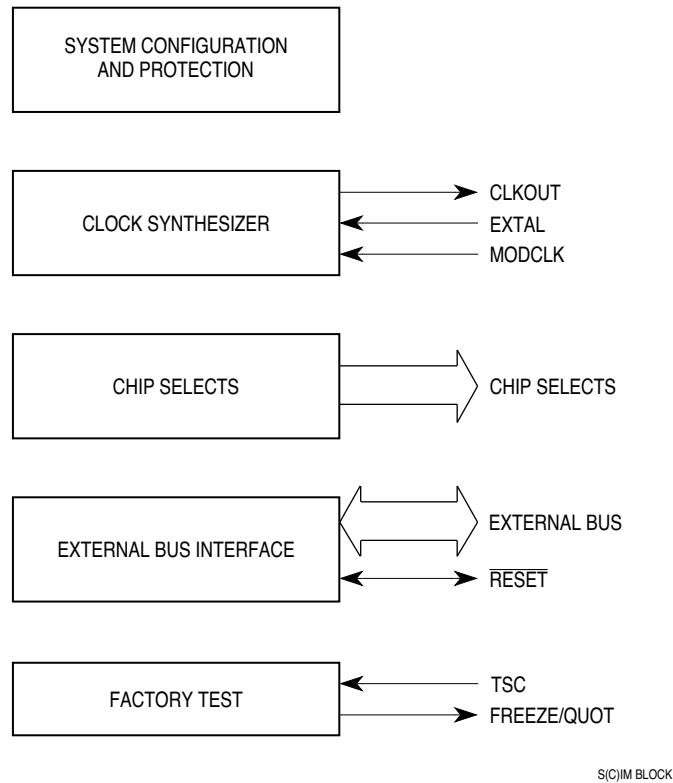


Figure 5 SIM Block Diagram

3.1 Overview

The system configuration and protection block controls MCU configuration and operating mode. The block also provides bus and software watchdog monitors.

The system clock generates clock signals used by the SIM, other IMB modules, and external devices. In addition, a periodic interrupt generator supports execution of time-critical control routines.

The external bus interface handles the transfer of information between IMB modules and external address space.

The chip-select block provides eleven general-purpose chip-select signals and a boot ROM chip select signal. Both general-purpose and boot ROM chip-select signals have associated base address registers and option registers.

The system test block incorporates hardware necessary for testing the MCU. It is used to perform factory tests, and its use in normal applications is not supported.

The SIM control register address map occupies 128 bytes. Unused registers within the 128-byte address space return zeros when read. The "Access" column in the SIM address map below indicates which registers are accessible only at the supervisor privilege level and which can be assigned to either the supervisor or user privilege level, according to the value of the SUPV bit in the SIMCR.

Table 7 SIM Address Map

Access	Address	15	8	7	0
S	\$YFFA00	SIM CONFIGURATION (SIMCR)			
S	\$YFFA02	FACTORY TEST (SIMTR)			
S	\$YFFA04	CLOCK SYNTHESIZER CONTROL (SYNCR)			
S	\$YFFA06	NOT USED		RESET STATUS REGISTER (RSR)	
S	\$YFFA08	MODULE TEST E (SIMTRE)			
S	\$YFFA0A	NOT USED		NOT USED	
S	\$YFFA0C	NOT USED		NOT USED	
S	\$YFFA0E	NOT USED		NOT USED	
S/U	\$YFFA10	NOT USED		PORT E DATA (PORTE0)	
S/U	\$YFFA12	NOT USED		PORT E DATA (PORTE1)	
S/U	\$YFFA14	NOT USED		PORT E DATA DIRECTION (DDRE)	
S	\$YFFA16	NOT USED		PORT E PIN ASSIGNMENT (PEPAR)	
S/U	\$YFFA18	NOT USED		PORT F DATA (PORTF0)	
S/U	\$YFFA1A	NOT USED		PORT F DATA (PORTF1)	
S/U	\$YFFA1C	NOT USED		PORT F DATA DIRECTION (DDRF)	
S	\$YFFA1E	NOT USED		PORT F PIN ASSIGNMENT (PFPAR)	
S	\$YFFA20	NOT USED		SYSTEM PROTECTION CONTROL (SYPCR)	
S	\$YFFA22	PERIODIC INTERRUPT CONTROL (PICR)			
S	\$YFFA24	PERIODIC INTERRUPT TIMING (PITR)			
S	\$YFFA26	NOT USED		SOFTWARE SERVICE (SWSR)	
S	\$YFFA28	NOT USED		NOT USED	
S	\$YFFA2A	NOT USED		NOT USED	
S	\$YFFA2C	NOT USED		NOT USED	
S	\$YFFA2E	NOT USED		NOT USED	
S	\$YFFA30	TEST MODULE MASTER SHIFT A (TSTMSRA)			
S	\$YFFA32	TEST MODULE MASTER SHIFT B (TSTMSRB)			
S	\$YFFA34	TEST MODULE SHIFT COUNT (TSTSC)			
S	\$YFFA36	TEST MODULE REPETITION COUNTER (TSTRC)			
S	\$YFFA38	TEST MODULE CONTROL (CREG)			
S/U	\$YFFA3A	TEST MODULE DISTRIBUTED REGISTER (DREG)			
	\$YFFA3C	NOT USED		NOT USED	
	\$YFFA3E	NOT USED		NOT USED	
S/U	\$YFFA40	NOT USED		PORT C DATA (PORTC)	
	\$YFFA42	NOT USED		NOT USED	
S	\$YFFA44	CHIP-SELECT PIN ASSIGNMENT (CSPAR0)			
S	\$YFFA46	CHIP-SELECT PIN ASSIGNMENT (CSPAR1)			
S	\$YFFA48	CHIP-SELECT BASE BOOT (CSBARBT)			
S	\$YFFA4A	CHIP-SELECT OPTION BOOT (CSORBT)			
S	\$YFFA4C	CHIP-SELECT BASE 0 (CSBAR0)			
S	\$YFFA4E	CHIP-SELECT OPTION 0 (CSOR0)			
S	\$YFFA50	CHIP-SELECT BASE 1 (CSBAR1)			
S	\$YFFA52	CHIP-SELECT OPTION 1 (CSOR1)			
S	\$YFFA54	CHIP-SELECT BASE 2 (CSBAR2)			

Table 7 SIM Address Map (Continued)

Access	Address	15	8 7	0
S	\$YFFA56	CHIP-SELECT OPTION 2 (CSOR2)		
S	\$YFFA58	CHIP-SELECT BASE 3 (CSBAR3)		
S	\$YFFA5A	CHIP-SELECT OPTION 3 (CSOR3)		
S	\$YFFA5C	CHIP-SELECT BASE 4 (CSBAR4)		
S	\$YFFA5E	CHIP-SELECT OPTION 4 (CSOR4)		
S	\$YFFA60	CHIP-SELECT BASE 5 (CSBAR5)		
S	\$YFFA62	CHIP-SELECT OPTION 5 (CSOR5)		
S	\$YFFA64	CHIP-SELECT BASE 6 (CSBAR6)		
S	\$YFFA66	CHIP-SELECT OPTION 6 (CSOR6)		
S	\$YFFA68	CHIP-SELECT BASE 7 (CSBAR7)		
S	\$YFFA6A	CHIP-SELECT OPTION 7 (CSOR7)		
S	\$YFFA6C	CHIP-SELECT BASE 8 (CSBAR8)		
S	\$YFFA6E	CHIP-SELECT OPTION 8 (CSOR8)		
S	\$YFFA70	CHIP-SELECT BASE 9 (CSBAR9)		
S	\$YFFA72	CHIP-SELECT OPTION 9 (CSOR9)		
S	\$YFFA74	CHIP-SELECT BASE 10 (CSBAR10)		
S	\$YFFA76	CHIP-SELECT OPTION 10 (CSOR10)		
	\$YFFA78	NOT USED		NOT USED
	\$YFFA7A	NOT USED		NOT USED
	\$YFFA7C	NOT USED		NOT USED
	\$YFFA7E	NOT USED		NOT USED

Y = M111, where M is the logic state of the module mapping (MM) bit in the SIMCR.

3.2 System Configuration and Protection

This functional block provides configuration control for the entire MCU. It also performs interrupt arbitration, bus monitoring, and system test functions. MCU system protection includes a bus monitor, a HALT monitor, a spurious interrupt monitor, and a software watchdog timer. These functions have been made integral to the microcontroller to reduce the number of external components in a complete control system.

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	8	7	6	5	4	3	2	1	0
NOT USED								SWE	BMT
RESET:								1	0
								MODCLK	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

3.3.3 Clock Control

The clock control circuits determine system clock frequency and clock operation under special circumstances, such as following loss of synthesizer reference or during low-power operation. Clock source is determined by the logic state of the MODCLK pin during reset.

SYNCR — Clock Synthesizer Control Register

\$YFFA04

15	14	13		8	7	6	5	4	3	2	1	0
W	X	Y			EDIV	0	0	SLIMP	SLOCK	RSTEN	STSIM	STEXT

RESET:

0 0 1 1 1 1 1 1 0 0 0 U U 0 0 0

When the on-chip clock synthesizer is used, system clock frequency is controlled by the bits in the upper byte of SYNCR. Bits in the lower byte show status of or control operation of internal and external clocks. The SYNCR can be read or written only when the CPU is operating at the supervisor privilege level.

W — Frequency Control (VCO)

This bit controls a prescaler tap in the synthesizer feedback loop. Setting the bit increases the VCO speed by a factor of four. VCO relock delay is required.

X — Frequency Control Bit (Prescale)

This bit controls a divide by two prescaler that is not in the synthesizer feedback loop. Setting the bit doubles clock speed without changing the VCO speed. There is no VCO relock delay.

Y[5:0] — Frequency Control (Counter)

The Y field controls the modulus down counter in the synthesizer feedback loop, causing it to divide by a value of Y + 1. Values range from 0 to 63. VCO relock delay is required.

EDIV — E Clock Divide Rate

0 = ECLK frequency is system clock divided by 8.

1 = ECLK frequency is system clock divided by 16.

ECLK is an external M6800 bus clock available on pin ADDR23. Refer to **3.5 Chip Selects** for more information.

SLIMP — Limp Mode Flag

0 = External crystal is VCO reference.

1 = Loss of crystal reference.

When the on-chip synthesizer is used, loss of reference frequency causes SLIMP to be set. The VCO continues to run using the base control voltage. Maximum limp frequency is maximum specified system clock frequency. X-bit state affects limp frequency.

SLOCK — Synthesizer Lock Flag

0 = VCO is enabled, but has not locked.

1 = VCO has locked on the desired frequency (or system clock is external).

The MCU maintains reset state until the synthesizer locks, but SLOCK does not indicate synthesizer lock status until after the user writes to SYNCR.

RSTEN — Reset Enable

0 = Loss of crystal causes the MCU to operate in limp mode.

1 = Loss of crystal causes system reset.

STSIM — Stop Mode SIM Clock

0 = When LPSTOP is executed, the SIM clock is driven from the crystal oscillator and the VCO is turned off to conserve power.

1 = When LPSTOP is executed, the SIM clock is driven from the VCO.

STEXT — Stop Mode External Clock

0 = When LPSTOP is executed, the CLKOUT signal is held negated to conserve power.

1 = When LPSTOP is executed, the CLKOUT signal is driven from the SIM clock, as determined by the state of the STSIM bit.

CSPAR1 —Chip Select Pin Assignment Register 1

\$YFFA46

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	CSPA1[4]		CSPA1[3]		CSPA1[2]		CSPA1[1]		CSPA1[0]	
RESET:															
0	0	0	0	0	0	DATA7	1	DATA [7:6]	1	DATA [7:5]	1	DATA [7:4]	1	DATA [7:3]	1

CSPAR1 contains five 2-bit fields that determine the functions of corresponding chip-select pins. CSPAR1[15:10] are not used. These bits always read zero; writes have no effect.

Table 14 CSPAR1 Pin Assignments

CSPAR0 Field	Chip Select Signal	Alternate Signal	Discrete Output
CSPA1[4]	$\overline{CS10}$	ADDR23	ECLK
CSPA1[3]	$\overline{CS9}$	ADDR22	PC6
CSPA1[2]	$\overline{CS8}$	ADDR21	PC5
CSPA1[1]	$\overline{CS7}$	ADDR20	PC4
CSPA1[0]	$\overline{CS6}$	ADDR19	PC3

At reset, either the alternate function (01) or chip-select function (11) can be encoded. DATA pins are driven to logic level one by a weak interval pull-up during reset. Encoding is for chip-select function unless a data line is held low during reset. Note that bus loading can overcome the weak pull-up and hold pins low during reset. The following table shows the hierarchical selection method that determines the reset functions of pins controlled by CSPAR1.

Table 15 Reset Pin Function of $\overline{CS}[10:6]$

Data Bus Pins at Reset					Chip-Select/Address Bus Pin Function				
DATA7	DATA6	DATA5	DATA4	DATA3	$\overline{CS10}/$ ADDR23	$\overline{CS9}/$ ADDR22	$\overline{CS8}/$ ADDR21	$\overline{CS7}/$ ADDR20	$\overline{CS6}/$ ADDR19
1	1	1	1	1	$\overline{CS10}$	$\overline{CS9}$	$\overline{CS8}$	$\overline{CS7}$	$\overline{CS6}$
1	1	1	1	0	$\overline{CS10}$	$\overline{CS9}$	$\overline{CS8}$	$\overline{CS7}$	ADDR19
1	1	1	0	X	$\overline{CS10}$	$\overline{CS9}$	$\overline{CS8}$	ADDR20	ADDR19
1	1	0	X	X	$\overline{CS10}$	$\overline{CS9}$	ADDR21	ADDR20	ADDR19
1	0	X	X	X	$\overline{CS10}$	ADDR22	ADDR21	ADDR20	ADDR19
0	X	X	X	X	ADDR23	ADDR22	ADDR21	ADDR20	ADDR19

A pin programmed as a discrete output drives an external signal to the value specified in the port C pin data register (PORTC), with the following exceptions:

1. No discrete output function is available on pins \overline{BR} , \overline{BG} , or \overline{BGACK} .
2. ADDR23 provides E-clock output rather than a discrete output signal.

When a pin is programmed for discrete output or alternate function, internal chip-select logic still functions and can be used to generate \overline{DSACK} or \overline{AVEC} internally on an address match.

Port size is determined when a pin is assigned as a chip select. When a pin is assigned to an 8-bit port, the chip select is asserted at all addresses within the block range. If a pin is assigned to a 16-bit port, the upper/lower byte field of the option register selects the byte with which the chip select is associated.

PPFAR — Port F Pin Assignment Register

\$YFFA1F

15	8	7	6	5	4	3	2	1	0
NOT USED		PFPA7	PFPA6	PFPA5	PFPA4	PFPA3	PFPA2	PFPA1	PFPA0

RESET:

DATA9 DATA9 DATA9 DATA9 DATA9 DATA9 DATA9 DATA9

The bits in this register control the function of each port F pin. Any bit cleared to zero defines the corresponding pin to be an I/O pin. Any bit set to one defines the corresponding pin to be an interrupt request signal or MODCLK. The MODCLK signal has no function after reset.

Table 17 Port F Pin Assignments

PFFAR Field	Port F Signal	Alternate Signal
PFFA7	PF7	$\overline{\text{IRQ7}}$
PFFA6	PF6	$\overline{\text{IRQ6}}$
PFFA5	PF5	$\overline{\text{IRQ5}}$
PFFA4	PF4	$\overline{\text{IRQ4}}$
PFFA3	PF3	$\overline{\text{IRQ3}}$
PFFA2	PF2	$\overline{\text{IRQ2}}$
PFFA1	PF1	$\overline{\text{IRQ1}}$
PFFA0	PF0	MODCLK

Data bus pin 9 controls the state of this register following reset. If DATA9 is set to one during reset, the register is set to \$FF, which defines all port F pins as interrupt request inputs. If DATA9 is cleared to zero during reset, this register is set to \$00, defining all port F pins as I/O pins.

3.7 Resets

Reset procedures handle system initialization and recovery from catastrophic failure. The MCU performs resets with a combination of hardware and software. The system integration module determines whether a reset is valid, asserts control signals, performs basic system configuration based on hardware mode-select inputs, then passes control to the CPU.

Reset occurs when an active low logic level on the $\overline{\text{RESET}}$ pin is clocked into the SIM. Resets are gated by the CLKOUT signal. Asynchronous resets are assumed to be catastrophic. An asynchronous reset can occur on any clock edge. Synchronous resets are timed to occur at the end of bus cycles. If there is no clock when $\overline{\text{RESET}}$ is asserted, reset does not occur until the clock starts. Resets are clocked in order to allow completion of write cycles in progress at the time $\overline{\text{RESET}}$ is asserted.

Reset is the highest-priority CPU32 exception. Any processing in progress is aborted by the reset exception, and cannot be restarted. Only essential tasks are performed during reset exception processing. Other initialization tasks must be accomplished by the exception handler routine.

3.7.1 SIM Reset Mode Selection

The logic states of certain data bus pins during reset determine SIM operating configuration. In addition, the state of the MODCLK pin determines system clock source and the state of the BKPT pin determines what happens during subsequent breakpoint assertions. The following table is a summary of reset mode selection options.

Table 18 Reset Mode Selection

Mode Select Pin	Default Function (Pin Left High)	Alternate Function (Pin Pulled Low)
-----------------	-------------------------------------	--

Table 18 Reset Mode Selection

DATA0	CSBOOT 16-Bit	CSBOOT 8-Bit
DATA1	CS0 CS1 CS2	BR BG BGACK
DATA2	CS3 CS4 CS5	FC0 FC1 FC2
DATA3 DATA4 DATA5 DATA6 DATA7	CS6 CS[7:6] CS[8:6] CS[9:6] CS[10:6]	ADDR19 ADDR[20:19] ADDR[21:19] ADDR[22:19] ADDR[23:19]
DATA8	DSACK0, DSACK1, AVEC, DS, AS, SIZ[1:0]	PORTE
DATA9	IRQ[7:1] MODCLK	PORTF
DATA11	Test Mode Disabled	Test Mode Enabled
MODCLK	VCO = System Clock	EXTAL = System Clock
BKPT	Background Mode Disabled	Background Mode Enabled

3.7.2 Functions of Pins for Other Modules During Reset

Generally, pins associated with modules other than the SIM default to port functions, and input/output ports are set to input state. This is accomplished by disabling pin functions in the appropriate control registers, and by clearing the appropriate port data direction registers. Refer to individual module sections in this manual for more information. The following table is a summary of module pin function out of reset.

Table 19 Module Pin Functions

Module	Pin Mnemonic	Function
CPU32	DSI/IFETCH	DSI/IFETCH
	DSO/IPIPE	DSO/IPIPE
	BKPT/DSCLK	BKPT/DSCLK
GPT	PGP7/IC4/OC5	Discrete Input
	PGP[6:3]/OC[4:1]	Discrete Input
	PGP[2:0]/IC[3:1]	Discrete Input
	PAI	Discrete Input
	PCLK	Discrete Input
	PWMA, PWMB	Discrete Output
QSM	PQS7/TXD	Discrete Input
	PQS[6:4]/PCS[3:1]	Discrete Input
	PQS3/PCS0/SS	Discrete Input
	PQS2/SCK	Discrete Input
	PQS1/MOSI	Discrete Input
	PQS0/MISO	Discrete Input
	RXD	RXD

4.6 Instruction Set Summary

Table 20 Instruction Set Summary

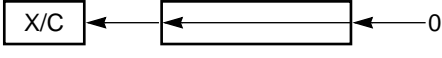
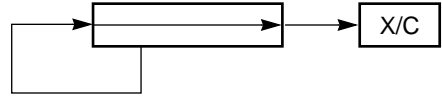
Instruction	Syntax	Operand Size	Operation
ABCD	Dn, Dn – (An), – (An)	8 8	Source ₁₀ + Destination ₁₀ + X ⇒ Destination
ADD	Dn, <ea> <ea>, Dn	8, 16, 32 8, 16, 32	Source + Destination ⇒ Destination
ADDA	<ea>, An	16, 32	Source + Destination ⇒ Destination
ADDI	#<data>, <ea>	8, 16, 32	Immediate data + Destination ⇒ Destination
ADDQ	# <data>, <ea>	8, 16, 32	Immediate data + Destination ⇒ Destination
ADDX	Dn, Dn – (An), – (An)	8, 16, 32 8, 16, 32	Source + Destination + X ⇒ Destination
AND	<ea>, Dn Dn, <ea>	8, 16, 32 8, 16, 32	Source • Destination ⇒ Destination
ANDI	# <data>, <ea>	8, 16, 32	Data • Destination ⇒ Destination
ANDI to CCR	# <data>, CCR	8	Source • CCR ⇒ CCR
ANDI to SR1 ¹	# <data>, SR	16	Source • SR ⇒ SR
ASL	Dn, Dn # <data>, Dn i	8, 16, 32 8, 16, 32 16	
ASR	Dn, Dn # <data>, Dn i	8, 16, 32 8, 16, 32 16	
Bcc	label	8, 16, 32	If condition true, then PC + d ⇒ PC
BCHG	Dn, <ea> # <data>, <ea>	8, 32 8, 32	(bit number) of destination ⇒ Z ⇒ bit of destination
BCLR	Dn, <ea> # <data>, <ea>	8, 32 8, 32	(bit number) of destination 0 ⇒ bit of destination
BGND	none	none	If background mode enabled, then enter background mode, else format/vector ⇒ – (SSP); PC ⇒ – (SSP); SR ⇒ – (SSP); (vector) ⇒ PC
BKPT	# <data>	none	If breakpoint cycle acknowledged, then execute returned operation word, else trap as illegal instruction
BRA	label	8, 16, 32	PC + d ⇒ PC
BSET	Dn, <ea> # <data>, <ea>	8, 32 8, 32	(bit number) of destination ⇒ Z; 1 ⇒ bit of destination
BSR	label	8, 16, 32	SP – 4 ⇒ SP; PC ⇒ (SP); PC + d ⇒ PC
BTST	Dn, <ea> # <data>, <ea>	8, 32 8, 32	(bit number) of destination ⇒ Z
CHK	<ea>, Dn	16, 32	If Dn < 0 or Dn > (ea), then CHK exception
CHK2	<ea>, Rn	8, 16, 32	If Rn < lower bound or Rn > upper bound, then CHK exception
CLR	i	8, 16, 32	0 ⇒ Destination
CMP	<ea>, Dn	8, 16, 32	(Destination – Source), CCR shows results
CMPA	<ea>, An	16, 32	(Destination – Source), CCR shows results
CMPI	# <data>, <ea>	8, 16, 32	(Destination – Data), CCR shows results
CMPM	(An) +, (An) +	8, 16, 32	(Destination – Source), CCR shows results
CMP2	<ea>, Rn	8, 16, 32	Lower bound ≤ Rn ≤ Upper bound, CCR shows result

Table 20 Instruction Set Summary(Continued)

Instruction	Syntax	Operand Size	Operation
SWAP	Dn	16	
TAS	í	8	Destination Tested Condition Codes bit 7 of Destination
TBLS/TBLU	<ea>, Dn Dym : Dyn, Dn	8, 16, 32	$\text{Dyn} - \text{Dym} \Rightarrow \text{Temp}$ $(\text{Temp} * \text{Dn} [7 : 0]) \Rightarrow \text{Temp}$ $(\text{Dym} * 256) + \text{Temp} \Rightarrow \text{Dn}$
TBLSN/TBLUN	<ea>, Dn Dym : Dyn, Dn	8, 16, 32	$\text{Dyn} - \text{Dym} \Rightarrow \text{Temp}$ $(\text{Temp} * \text{Dn} [7 : 0]) / 256 \Rightarrow \text{Temp}$ $\text{Dym} + \text{Temp} \Rightarrow \text{Dn}$
TRAP	#<data>	none	$\text{SSP} - 2 \Rightarrow \text{SSP}$; format/vector offset $\Rightarrow (\text{SSP})$; $\text{SSP} - 4 \Rightarrow \text{SSP}$; PC $\Rightarrow (\text{SSP})$; SR $\Rightarrow (\text{SSP})$; vector address $\Rightarrow \text{PC}$
TRAPcc	none #<data>	none 16, 32	If cc true, then TRAP exception
TRAPV	none	none	If V set, then overflow TRAP exception
TST	í	8, 16, 32	Source – 0, to set condition codes
UNLK	An	32	$\text{An} \Rightarrow \text{SP}$; $(\text{SP}) \Rightarrow \text{An}$, $\text{SP} + 4 \Rightarrow \text{SP}$

1. Privileged instruction.

5.1.2 Input Capture/Input Transition Counter (ITC)

Any channel of the TPU can capture the value of a specified TCR upon the occurrence of each transition or specified number of transitions, and then generate an interrupt request to notify the CPU. A channel can perform input captures continually, or a channel can detect a single transition or specified number of transitions, then cease channel activity until reinitialization. After each transition or specified number of transitions, the channel can generate a link to a sequential block of up to eight channels. The user specifies a starting channel of the block and the number of channels within the block. The generation of links depends on the mode of operation. In addition, after each transition or specified number of transitions, one byte of the parameter RAM (at an address specified by channel parameter) can be incremented and used as a flag to notify another channel of a transition.

5.1.3 Output Compare (OC)

The output compare function generates a rising edge, falling edge, or a toggle of the previous edge in one of three ways:

1. Immediately upon CPU initiation, thereby generating a pulse with a length equal to a programmable delay time.
2. At a programmable delay time from a user-specified time.
3. Continuously. Upon receiving a link from a channel, OC references, without CPU interaction, a specifiable period and calculates an offset:

$$\text{Offset} = \text{Period} * \text{Ratio}$$

where Ratio is a parameter supplied by the user.

This algorithm generates a 50% duty-cycle continuous square wave with each high/low time equal to the calculated OFFSET. Due to offset calculation, there is an initial link time before continuous pulse generation begins.

5.1.4 Pulse-Width Modulation (PWM)

The TPU can generate a pulse-width modulation waveform with any duty cycle from zero to 100% (within the resolution and latency capability of the TPU). To define the PWM, the CPU provides one parameter that indicates the period and another parameter that indicates the high time. Updates to one or both of these parameters can direct the waveform change to take effect immediately, or coherently beginning at the next low-to-high transition of the pin.

5.1.5 Synchronized Pulse-Width Modulation (SPWM)

The TPU generates a PWM waveform in which the CPU can change the period and/or high time at any time. When synchronized to a time function on a second channel, the synchronized PWM low-to-high transitions have a time relationship to transitions on the second channel.

5.1.6 Period Measurement with Additional Transition Detect (PMA)

This function and the following function are used primarily in toothed-wheel speed-sensing applications, such as monitoring rotational speed of an engine. The period measurement with additional transition detect function allows for a special-purpose 23-bit period measurement. It can detect the occurrence of an additional transition (caused by an extra tooth on the sensed wheel) indicated by a period measurement that is less than a programmable ratio of the previous period measurement.

Once detected, this condition can be counted and compared to a programmable number of additional transitions detected before TCR2 is reset to \$FFFF. Alternatively, a byte at an address specified by a channel parameter can be read and used as a flag. A nonzero value of the flag indicates that TCR2 is to be reset to \$FFFF once the next additional transition is detected.

HSQR0 — Host Sequence Register 0

\$YFFE14

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CH 15	CH 14	CH 13	CH 12	CH 11	CH 10	CH 9	CH 8								
RESET:															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

HSQR1 — Host Sequence Register 1

\$YFFE16

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CH 7	CH 6	CH 5	CH 4	CH 3	CH 2	CH 1	CH 0								
RESET:															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

CH[15:0] — Encoded Host Sequence

The host sequence field selects the mode of operation for the time function selected on a given channel. The meaning of the host sequence bits depends on the time function specified.

HSRR0 — Host Service Request Register 0

\$YFFE18

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CH 15	CH 14	CH 13	CH 12	CH 11	CH 10	CH 9	CH 8								
RESET:															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

HSRR1 — Host Service Request Register 1

\$YFFE1A

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CH 7	CH 6	CH 5	CH 4	CH 3	CH 2	CH 1	CH 0								
RESET:															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

CH[15:0] — Encoded Type of Host Service

The host service request field selects the type of host service request for the time function selected on a given channel. The meaning of the host service request bits depends on the time function specified. A host service request field cleared to %00 signals the host that service is completed by the microengine on that channel. The host can request service on a channel by writing the corresponding host service request field to one of three nonzero states. The CPU should monitor the host service request register until the TPU clears the service request to %00 before the CPU changes any parameters or issues a new service request to the channel.

CPR0 — Channel Priority Register 0

\$YFFE1C

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CH 15	CH 14	CH 13	CH 12	CH 11	CH 10	CH 9	CH 8								
RESET:															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

CPR1 — Channel Priority Register 1

\$YFFE1E

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CH 7	CH 6	CH 5	CH 4	CH 3	CH 2	CH 1	CH 0								
RESET:															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

CH[15:0] — Encoded One of Three Channel Priority Levels

Pin Names	Mnemonics	Mode	Function
Master In Slave Out	MISO	Master Slave	Serial Data Input to QSPI Serial Data Output from QSPI
Master Out Slave In	MOSI	Master Slave	Serial Data Output from QSPI Serial Data Input to QSPI
Serial Clock	SCK	Master Slave	Clock Output from QSPI Clock Input to QSPI
Peripheral Chip Selects	PCS[3:1]	Master	Select Peripherals
Peripheral Chip Select Slave Select	PCS0 SS	Master Master Slave	Selects Peripheral Causes Mode Fault Initiates Serial Transfer

6.5.2 QSPI Registers

The programmer's model for the QSPI submodule consists of the QSM global and pin control registers, four QSPI control registers, one status register, and the 80-byte QSPI RAM.

The CPU can read and write to registers and RAM. The four control registers must be initialized before the QSPI is enabled to ensure defined operation. SPCR1 should be written last because it contains QSPI enable bit SPE. Asserting this bit starts the QSPI. The QSPI control registers are reset to a defined state and can then be changed by the CPU. Reset values are shown below each register.

Refer to the following memory map of the QSPI.

Address	Name	Usage
\$YFFC18	SPCR0	QSPI Control Register 0
\$YFFC1A	SPCR1	QSPI Control Register 1
\$YFFC1C	SPCR2	QSPI Control Register 2
\$YFFC1E	SPCR3	QSPI Control Register 3
\$YFFC1F	SPSR	QSPI Status Register
\$YFFD00	RAM	QSPI Receive Data (16 Words)
\$YFFD20	RAM	QSPI Transmit Data (16 Words)
\$YFFD40	RAM	QSPI Command Control (8 Words)

Writing a different value into any control register except SPCR2 while the QSPI is enabled disrupts operation. SPCR2 is buffered to prevent disruption of the current serial transfer. After completion of the current serial transfer, the new SPCR2 values become effective.

Writing the same value into any control register except SPCR2 while the QSPI is enabled has no effect on QSPI operation. Rewriting NEWQP in SPCR2 causes execution to restart at the designated location.

SPCR0 — QSPI Control Register 0

\$YFFC18

15	14	13		10	9	8	7								0
MSTR	WOMQ			BITS		CPOL	CPHA								SPBR

RESET:

0 0 0 0 0 0 0 1 0 0 0 0 0 0 1 0 0

SPCR0 contains parameters for configuring the QSPI before it is enabled. The CPU can read and write this register. The QSM has read-only access.

MSTR — Master/Slave Mode Select

0 = QSPI is a slave device and only responds to externally generated serial data.

1 = QSPI is system master and can initiate transmission to external SPI devices.

MSTR configures the QSPI for either master or slave mode operation. This bit is cleared on reset and may only be written by the CPU.

WOMQ — Wired-OR Mode for QSPI Pins

0 = Outputs have normal MOS drivers.

1 = Pins designated for output by DDRQS have open-drain drivers.

WOMQ allows the wired-OR function to be used on QSPI pins, regardless of whether they are used as general-purpose outputs or as QSPI outputs. WOMQ affects the QSPI pins regardless of whether the QSPI is enabled or disabled.

BITS — Bits Per Transfer

In master mode, when BITSE in a command is set, the BITS field determines the number of data bits transferred. When BITSE is cleared, eight bits are transferred. Reserved values default to eight bits. BITSE is not used in slave mode.

The following table shows the number of bits per transfer.

BITS	Bits per Transfer
0000	16
0001	Reserved
0010	Reserved
0011	Reserved
0100	Reserved
0101	Reserved
0110	Reserved
0111	Reserved
1000	8
1001	9
1010	10
1011	11
1100	12
1101	13
1110	14
1111	15

CPOL — Clock Polarity

0 = The inactive state value of SCK is logic level zero.

1 = The inactive state value of SCK is logic level one.

CPOL is used to determine the inactive state value of the serial clock (SCK). It is used with CPHA to produce a desired clock/data relationship between master and slave devices.

CPHA — Clock Phase

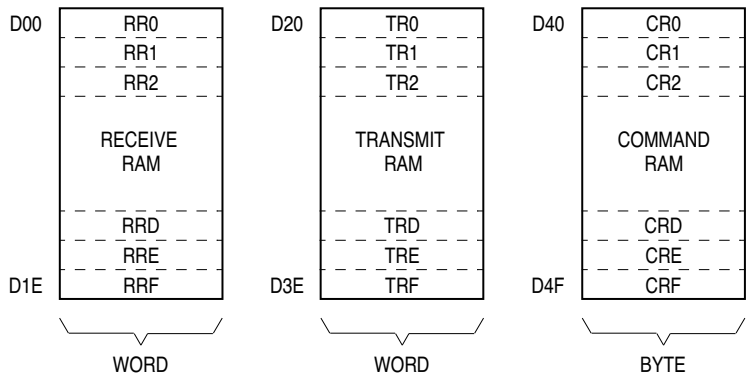
0 = Data is captured on the leading edge of SCK and changed on the following edge of SCK.

1 = Data is changed on the leading edge of SCK and captured on the following edge of SCK.

CPHA determines which edge of SCK causes data to change and which edge causes data to be captured. CPHA is used with CPOL to produce a desired clock/data relationship between master and slave devices. CPHA is set at reset.

SPBR — Serial Clock Baud Rate

The QSPI uses a modulus counter to derive SCK baud rate from the MCU system clock. Baud rate is selected by writing a value from 2 to 255 into the SPBR field. The following equation determines the



QSPI RAM MAP

Figure 15 QSPI RAM

Once the CPU has set up the queue of QSPI commands and enabled the QSPI, the QSPI can operate independently of the CPU. The QSPI executes all of the commands in its queue, sets a flag indicating that it is finished, and then either interrupts the CPU or waits for CPU intervention. It is possible to execute a queue of commands repeatedly without CPU intervention.

RR[0:F] — Receive Data RAM **\$YFFD00**

Data received by the QSPI is stored in this segment. The CPU reads this segment to retrieve data from the QSPI. Data stored in receive RAM is right-justified. Unused bits in a receive queue entry are set to zero by the QSPI upon completion of the individual queue entry. The CPU can access the data using byte, word, or long-word addressing.

The CPTQP value in SPSR shows which queue entries have been executed. The CPU uses this information to determine which locations in receive RAM contain valid data before reading them.

TR[0:F] — Transmit Data RAM **\$YFFD20**

Data that is to be transmitted by the QSPI is stored in this segment. The CPU usually writes one word of data into this segment for each queue command to be executed.

Information to be transmitted must be written to transmit data RAM in a right-justified format. The QSPI cannot modify information in the transmit data RAM. The QSPI copies the information to its data serializer for transmission. Information remains in transmit RAM until overwritten.

CR[0:F] — Command RAM **\$YFFD40**

7	6	5	4	3	2	1	0
CONT	BITSE	DT	DSCK	PCS3	PCS2	PCS1	PCS0*
—	—	—	—	—	—	—	—
CONT	BITSE	DT	DSCK	PCS3	PCS2	PCS1	PCS0*

COMMAND CONTROL

PERIPHERAL CHIP SELECT

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

Command RAM is used by the QSPI when in master mode. The CPU writes one byte of control information to this segment for each QSPI command to be executed. The QSPI cannot modify information in command RAM.

Command RAM consists of 16 bytes. Each byte is divided into two fields. The peripheral chip-select field enables peripherals for transfer. The command control field provides transfer options.

A maximum of 16 commands can be in the queue. Queue execution by the QSPI proceeds from the address in NEWQP through the address in ENDQP. (Both of these fields are in SPCR2.)

CONT — Continue

- 0 = Control of chip selects returned to PORTQS after transfer is complete.
- 1 = Peripheral chip selects remain asserted after transfer is complete.

BITSE — Bits per Transfer Enable

- 0 = 8 bits
- 1 = Number of bits set in BITS field of SPCR0

DT — Delay after Transfer

The QSPI provides a variable delay at the end of serial transfer to facilitate the interface with peripherals that have a latency requirement. The delay between transfers is determined by the SPCR1 DTL field.

DSCK — PCS to SCK Delay

- 0 = PCS valid to SCK transition is one-half SCK.
- 1 = SPCR1 DSCKL field specifies delay from PCS valid to SCK.

PCS[3:0] — Peripheral Chip Select

Use peripheral chip-select bits to select an external device for serial data transfer. More than one peripheral chip select can be activated at a time, and more than one peripheral chip can be connected to each PCS pin, provided that proper fanout is observed.

\overline{SS} — Slave Mode Select

Initiates slave mode serial transfer. If \overline{SS} is taken low when the QSPI is in master mode, a mode fault will be generated.

6.5.4 Operating Modes

The QSPI operates in either master or slave mode. Master mode is used when the MCU originates data transfers. Slave mode is used when an external device initiates serial transfers to the MCU through the QSPI. Switching between the modes is controlled by MSTR in SPCR0. Before entering either mode, appropriate QSM and QSPI registers must be properly initialized.

In master mode, the QSPI executes a queue of commands defined by control bits in each command RAM queue entry. Chip-select pins are activated, data is transmitted from transmit RAM and received into receive RAM.

In slave mode, operation proceeds in response to \overline{SS} pin activation by an external bus master. Operation is similar to master mode, but no peripheral chip selects are generated, and the number of bits transferred is controlled in a different manner. When the QSPI is selected, it automatically executes the next queue transfer to exchange data with the external device correctly.

Although the QSPI inherently supports multimaster operation, no special arbitration mechanism is provided. A mode fault flag (MODF) indicates a request for SPI master arbitration. System software must provide arbitration. Note that unlike previous SPI systems, MSTR is not cleared by a mode fault being set, nor are the QSPI pin output drivers disabled. The QSPI and associated output drivers must be disabled by clearing SPE in SPCR1.

7 Standby RAM with TPU Emulation RAM

The TPURAM module contains a 2-Kbyte array of fast (two bus cycle) static RAM, which is especially useful for system stacks and variable storage. Alternately, it can be used by the TPU as emulation RAM for new timer algorithms.

7.1 Overview

The TPURAM can be mapped to any 4-Kbyte boundary in the address map, but must not overlap the module control registers. (Overlap makes the registers inaccessible.) Data can be read or written in bytes, word, or long words. TPURAM responds to both program and data space accesses. Data can be read or written in bytes, words, or long words. The TPURAM is powered by V_{DD} in normal operation. During power-down, the TPURAM contents are maintained by power on standby voltage pin V_{STBY} . Power switching between sources is automatic.

Access to the TPURAM array is controlled by the RASP field in TRAMMCR. This field can be encoded so that TPURAM responds to both program and data space accesses. This allows code to be executed from TPURAM, and permits the use of program counter relative addressing mode for operand fetches from the array.

An address map of the TPURAM control registers follows. All TPURAM control registers are located in supervisor data space.

Table 28 TPURAM Control Register Address Map

Access	Address	15	8	7	0
S	\$YFFB00	TPURAM MODULE CONFIGURATION REGISTER (TRAMMCR)			
S	\$YFFB02	TPURAM TEST REGISTER (TRAMTST)			
S	\$YFFB04	TPURAM BASE ADDRESS REGISTER (TRAMBAR)			
	\$YFFB06– \$YFFB3F	NOT USED			

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

7.2 TPURAM Register Block

There are three TPURAM control registers: the RAM module configuration register (TRAMMCR), the RAM test register (TRAMTST), and the RAM array base address registers (TRAMBAR).

There is an 8-byte minimum register block size for the module. Unimplemented register addresses are read as zeros, and writes have no effect.

7.3 TPURAM Registers

TRAMMCR —TPURAM Module Configuration Register

\$YFFB00

15	14	13	12	11	10	9	8	7	0
STOP	0	0	0	0	0	0	RASP	NOT USED	

RESET:

0 0 0 0 0 0 0 1

TSTOP —Stop Control

0 = RAM array operates normally.

1 = RAM array enters low-power stop mode.

This bit controls whether the RAM 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.



Freescale Semiconductor, Inc.

Home Page:

www.freescale.com

email:

support@freescale.com

USA/Europe or Locations Not Listed:

Freescale Semiconductor
Technical Information Center, CH370
1300 N. Alma School Road
Chandler, Arizona 85224
(800) 521-6274
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
+81 2666 8080
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
(800) 441-2447
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>.

Information in this document is provided solely to enable system and software implementers to use Freescale Semiconductor products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits or integrated circuits based on the information in this document. Freescale Semiconductor reserves the right to make changes without further notice to any products herein. Freescale Semiconductor makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Freescale Semiconductor assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. "Typical" parameters which may be provided in Freescale Semiconductor data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including "Typicals" must be validated for each customer application by customer's technical experts. Freescale Semiconductor does not convey any license under its patent rights nor the rights of others. Freescale Semiconductor products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the Freescale Semiconductor product could create a situation where personal injury or death may occur. Should Buyer purchase or use Freescale Semiconductor products for any such unintended or unauthorized application, Buyer shall indemnify and hold Freescale Semiconductor and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that Freescale Semiconductor was negligent regarding the design or manufacture of the part.

