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	Not For New Designs
Core Processor	CPU32
Core Size	32-Bit Single-Core
Speed	20MHz
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/mc68332aceh20

Freescale Semiconductor, Inc.

Table 1 Ordering Information

| Package Type	TPU Type	Temperature	Frequency (MHz)	Package Order Quantity	Order Number
132-Pin PQFP	Motion Control	−40 to +85 °C	16 MHz	2 pc tray	SPAKMC332GCFC16
				36 pc tray	MC68332GCFC16
			20 MHz	2 pc tray	SPAKMC332GCFC20
				36 pc tray	MC68332GCFC20
		−40 to +105 °C	16 MHz	2 pc tray	SPAKMC332GVFC16
				36 pc tray	MC68332GVFC16
			20 MHz	2 pc tray	SPAKMC332GVFC20
				36 pc tray	MC68332GVFC20
		−40 to +125 °C	16 MHz	2 pc tray	SPAKMC332GMFC16
				36 pc tray	MC68332GMFC16
			20 MHz	2 pc tray	SPAKMC332GMFC20
				36 pc tray	MC68332GMFC20
	Standard	−40 to +85 °C	16 MHz	2 pc tray	SPAKMC332CFC16
				36 pc tray	MC68332CFC16
			20 MHz	2 pc tray	SPAKMC332CFC20
				36 pc tray	MC68332CFC20
		−40 to +105 °C	16 MHz	2 pc tray	SPAKMC332VFC16
				36 pc tray	MC68332VFC16
			20 MHz	2 pc tray	SPAKMC332VFC20
				36 pc tray	MC68332VFC20
		−40 to +125 °C	16 MHz	2 pc tray	SPAKMC332MFC16
				36 pc tray	MC68332MFC16
			20 MHz	2 pc tray	SPAKMC332MFC20
				36 pc tray	MC68332MFC20
	Std w/enhanced PPWA	−40 to +85 °C	16 MHz	2 pc tray	SPAKMC332ACFC16
				36 pc tray	MC68332ACFC16
			20 MHz	2 pc tray	SPAKMC332ACFC20
				36 pc tray	MC68332ACFC20
		−40 to +105 °C	16 MHz	2 pc tray	SPAKMC332AVFC16
				36 pc tray	MC68332AVFC16
			20 MHz	2 pc tray	SPAKMC332AVFC20
				36 pc tray	MC68332AVFC20
		−40 to +125 °C	16 MHz	2 pc tray	SPAKMC332AMFC16
				36 pc tray	MC68332AMFC16
			20 MHz	2 pc tray	SPAKMC332AMFC20
				36 pc tray	MC68332AMFC20

1.3 Pin Assignments
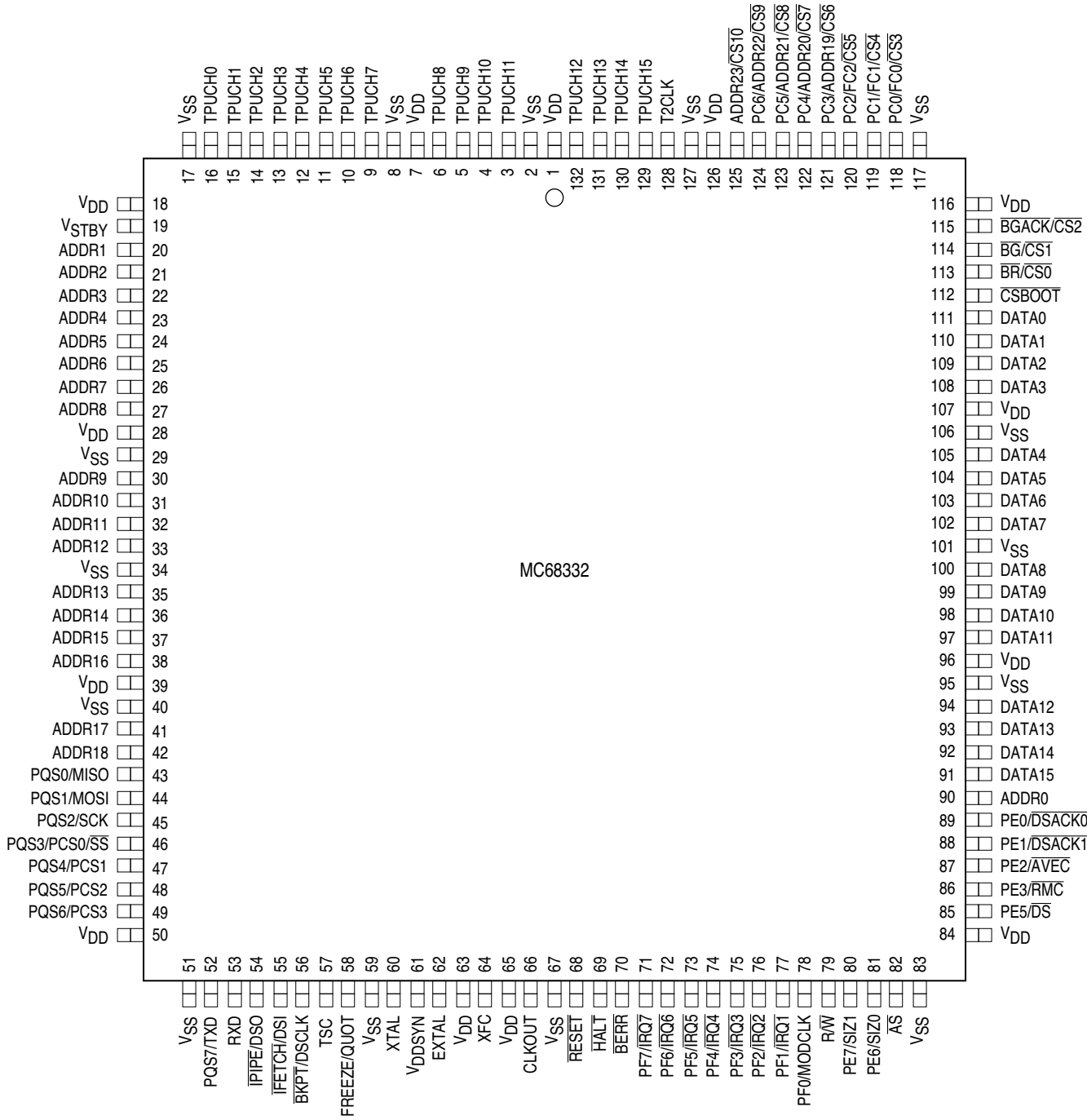
Figure 2 MC68332 132-Pin QFP Pin Assignments

332 132-PIN QFP

1.4 Address Map

The following figure is a map of the MCU internal addresses. The RAM array is positioned by the base address registers in the associated RAM control block. Unimplemented blocks are mapped externally.
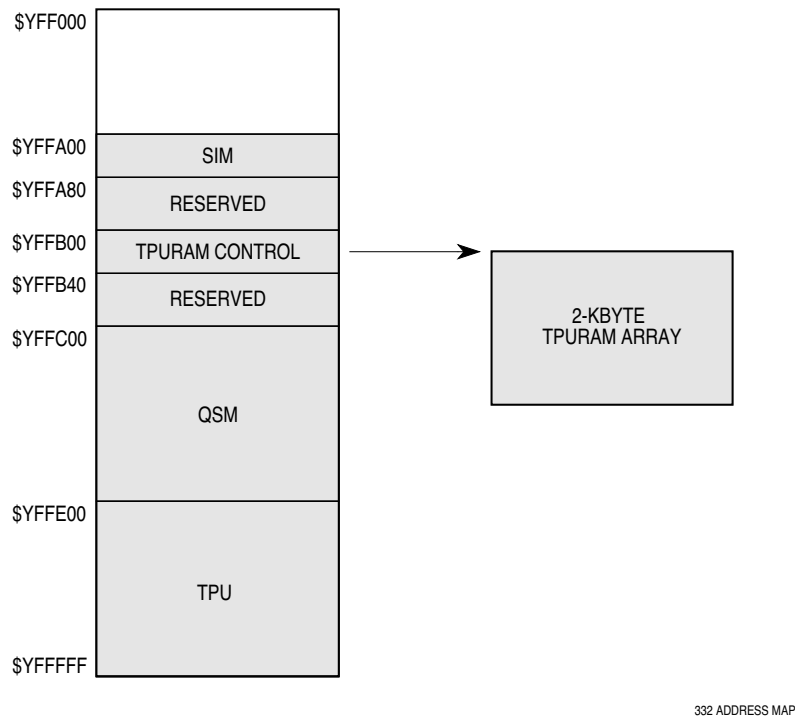

Figure 4 MCU Address Map

1.5 Intermodule Bus

The intermodule bus (IMB) is a standardized bus developed to facilitate both design and operation of modular microcontrollers. It contains circuitry to support exception processing, address space partitioning, multiple interrupt levels, and vectored interrupts. The standardized modules in the MCU communicate with one another and with external components through the IMB. The IMB in the MCU uses 24 address and 16 data lines.

Table 2 MCU Pin Characteristic (Continued)

Pin Mnemonic	Output Driver	Input Synchronized	Input Hysteresis	Discrete I/O	Port Designation
T2CLK	—	Y	Y	—	—
TPUCH[15:0]	A	Y	Y	—	—
TSC	—	Y	Y	—	—
TXD	Bo	Y	Y	I/O	PQS7
XFC[2]	—	—	—	Special	—
XTAL[2]	—	—	—	Special	—

NOTES:
1. DATA[15:0] are synchronized during reset only. MODCLK is synchronized only when used as an input port pin.
2. EXTAL, XFC, and XTAL are clock reference connections.

2.2 MCU Power Connections

Table 3 MCU Power Connections

V_{STBY}	Standby RAM Power/Clock Synthesizer Power
V_{DDSYN}	Clock Synthesizer Power
V_{SSE}/V_{DDE}	External Periphery Power (Source and Drain)
V_{SSI}/V_{DDI}	Internal Module Power (Source and Drain)

2.3 MCU Driver Types

Table 4 MCU Driver Types

Type	I/O	Description
A	O	Output-only signals that are always driven; no external pull-up required
Aw	O	Type A output with weak P-channel pull-up during reset
B	O	Three-state output that includes circuitry to pull up output before high impedance is established, to ensure rapid rise time. An external holding resistor is required to maintain logic level while the pin is in the high-impedance state.
Bo	O	Type B output that can be operated in an open-drain mode

2.4 Signal Characteristics

Table 5 MCU Signal Characteristics

Signal Name	MCU Module	Signal Type	Active State
ADDR[23:0]	SIM	Bus	—
$\overline{AS}$	SIM	Output	0
$\overline{AVEC}$	SIM	Input	0
$\overline{BERR}$	SIM	Input	0
$\overline{BG}$	SIM	Output	0
$\overline{BGACK}$	SIM	Input	0
$\overline{BKPT}$	CPU32	Input	0
$\overline{BR}$	SIM	Input	0
CLKOUT	SIM	Output	—
$\overline{CS}$[10:0]	SIM	Output	0
$\overline{CSBOOT}$	SIM	Output	0
DATA[15:0]	SIM	Bus	—
$\overline{DS}$	SIM	Output	0
$\overline{DSACK}$[1:0]	SIM	Input	0
DSCLK	CPU32	Input	Serial Clock
DSI	CPU32	Input	(Serial Data)
DSO	CPU32	Output	(Serial Data)
EXTAL	SIM	Input	—
FC[2:0]	SIM	Output	—
FREEZE	SIM	Output	1
$\overline{HALT}$	SIM	Input/Output	0
$\overline{IFETCH}$	CPU32	Output	—
$\overline{IPIPE}$	CPU32	Output	—
$\overline{IRQ}$[7:1]	SIM	Input	0
MISO	QSM	Input/Output	—
MODCLK	SIM	Input	—
MOSI	QSM	Input/Output	—
PC[6:0]	SIM	Output	(Port)
PCS[3:0]	QSM	Input/Output	—
PE[7:0]	SIM	Input/Output	(Port)
PF[7:0]	SIM	Input/Output	(Port)
PQS[7:0]	QSM	Input/Output	(Port)
QUOT	SIM	Output	—
$\overline{RESET}$	SIM	Input/Output	0
$\overline{RMC}$	SIM	Output	0
R/$\overline{W}$	SIM	Output	1/0
RXD	QSM	Input	—
SCK	QSM	Input/Output	—
SIZ[1:0]	SIM	Output	—
$\overline{SS}$	QSM	Input	0
T2CLK	TPU	Input	—
TPUCH[15:0]	TPU	Input/Output	1

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.
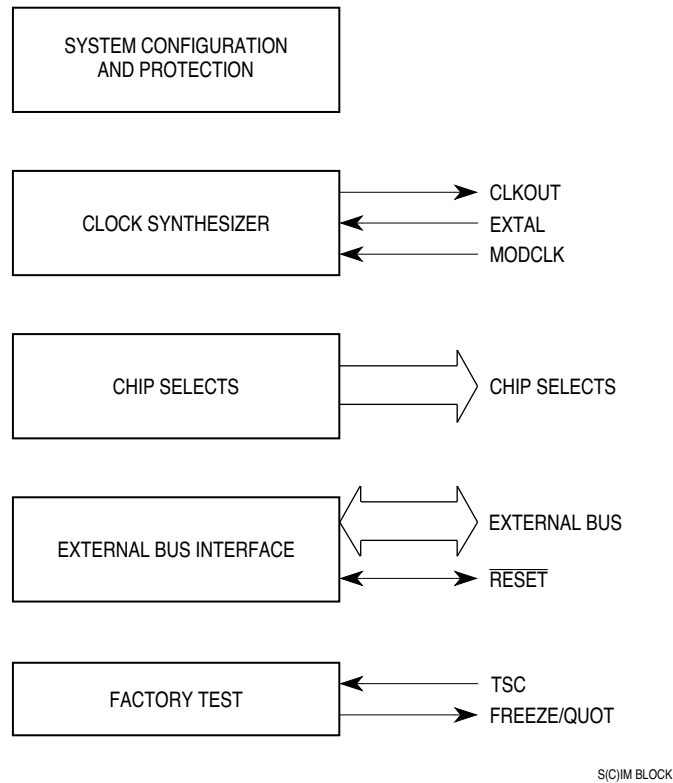
```
┌─────────────────────────┐
│  SYSTEM CONFIGURATION   │
│     AND PROTECTION      │
└─────────────────────────┘

┌─────────────────────────┐
│                         │ ──────►  CLKOUT
│    CLOCK SYNTHESIZER    │ ◄──────  EXTAL
│                         │ ◄──────  MODCLK
└─────────────────────────┘

┌─────────────────────────┐
│                         │
│      CHIP SELECTS       │ ══════►  CHIP SELECTS
│                         │
└─────────────────────────┘

┌─────────────────────────┐
│                         │ ◄═════►  EXTERNAL BUS
│  EXTERNAL BUS INTERFACE │
│                         │ ◄──────► RESET
└─────────────────────────┘

┌─────────────────────────┐
│                         │ ◄──────  TSC
│      FACTORY TEST       │ ──────►  FREEZE/QUOT
│                         │
└─────────────────────────┘
```

S(C)IM BLOCK

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.
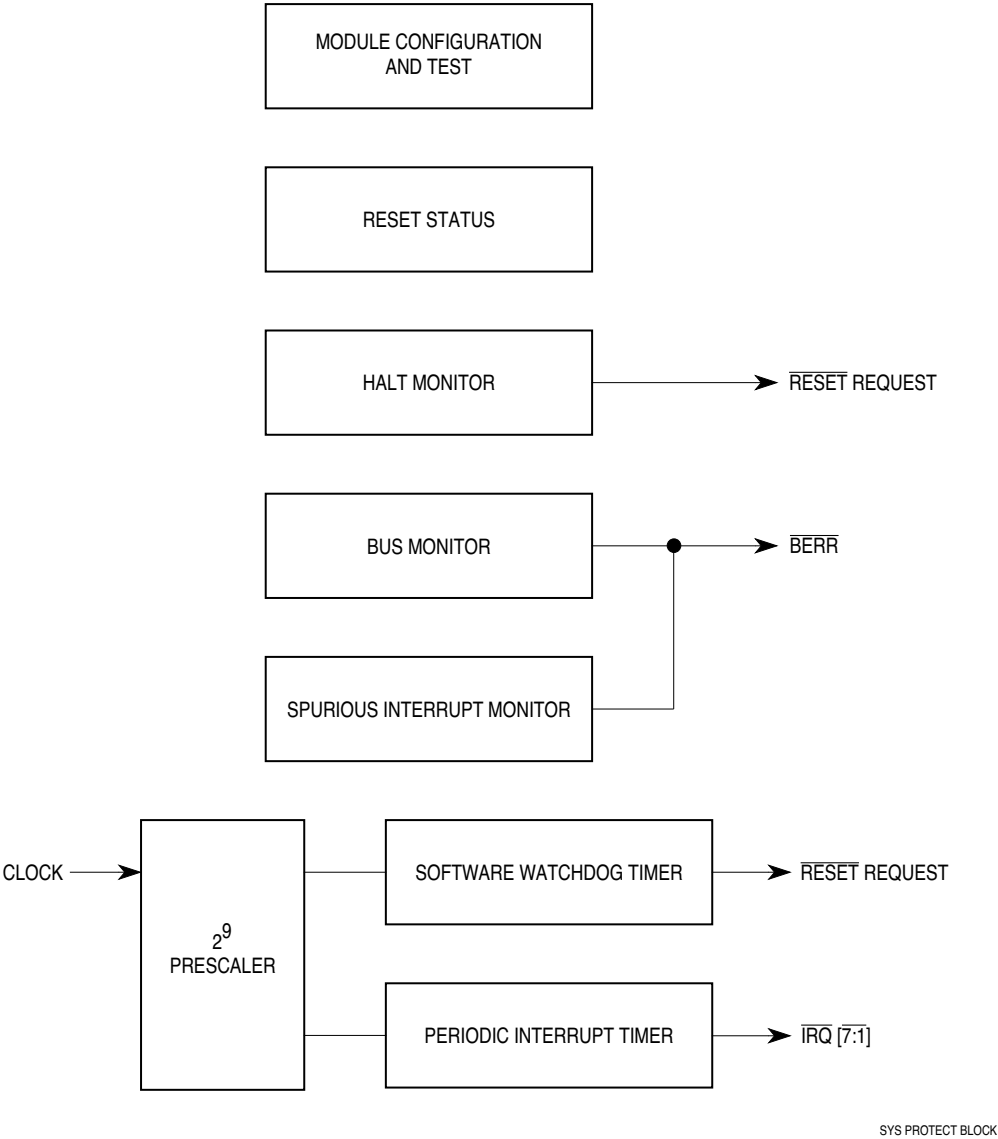
Figure 6 System Configuration and Protection Block

3.2.1 System Configuration

The SIM controls MCU configuration during normal operation and during internal testing.

SIMCR —SIM Configuration Register   **$YFFA00**

15	14	13	12	11	10	9	8	7	6	5	4	3			0
EXOFF	FRZSW	FRZBM	0	SLVEN	0	SHEN		SUPV	MM	0	0	IARB			

RESET:

| 0 | 0 | 0 | 0 | DATA11 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 |

The SIM configuration register controls system configuration. It can be read or written at any time, except for the module mapping (MM) bit, which can be written only once.

3.4 External Bus Interface

The external bus interface (EBI) transfers information between the internal MCU bus and external devices. The external bus has 24 address lines and 16 data lines.

The EBI provides dynamic sizing between 8-bit and 16-bit data accesses. It supports byte, word, and long-word transfers. Ports are accessed through the use of asynchronous cycles controlled by the data transfer (SIZ1 and SIZ0) and data size acknowledge pins ($\overline{DSACK1}$ and $\overline{DSACK0}$). Multiple bus cycles may be required for a transfer to or from an 8-bit port.

Port width is the maximum number of bits accepted or provided during a bus transfer. External devices must follow the handshake protocol described below. Control signals indicate the beginning of the cycle, the address space, the size of the transfer, and the type of cycle. The selected device controls the length of the cycle. Strobe signals, one for the address bus and another for the data bus, indicate the validity of an address and provide timing information for data. The EBI operates in an asynchronous mode for any port width.

To add flexibility and minimize the necessity for external logic, MCU chip-select logic can be synchronized with EBI transfers. Chip-select logic can also provide internally-generated bus control signals for these accesses. Refer to **3.5 Chip Selects** for more information.

3.4.1 Bus Control Signals

The CPU initiates a bus cycle by driving the address, size, function code, and read/write outputs. At the beginning of the cycle, size signals SIZ0 and SIZ1 are driven along with the function code signals. The size signals indicate the number of bytes remaining to be transferred during an operand cycle. They are valid while the address strobe ($\overline{AS}$) is asserted. The following table shows SIZ0 and SIZ1 encoding. The read/write (R/$\overline{W}$) signal determines the direction of the transfer during a bus cycle. This signal changes state, when required, at the beginning of a bus cycle, and is valid while $\overline{AS}$ is asserted. R/$\overline{W}$ only changes state when a write cycle is preceded by a read cycle or vice versa. The signal can remain low for two consecutive write cycles.

Table 8 Size Signal Encoding

SIZ1	SIZ0	Transfer Size
0	1	Byte
1	0	Word
1	1	Three Byte
0	0	Long Word

3.4.2 Function Codes

The CPU32 automatically generates function code signals FC[2:0]. The function codes can be considered address extensions that automatically select one of eight address spaces to which an address applies. These spaces are designated as either user or supervisor, and program or data spaces. Address space 7 is designated CPU space. CPU space is used for control information not normally associated with read or write bus cycles. Function codes are valid while $\overline{AS}$ is asserted.

PEPAR — Port E Pin Assignment Register   **$YFFA17**

15		8	7	6	5	4	3	2	1	0
	NOT USED		PEPA7	PEPA6	PEPA5	PEPA4	PEPA3	PEPA2	PEPA1	PEPA0

RESET:

| | | | DATA8 | DATA8 | DATA8 | DATA8 | DATA8 | DATA8 | DATA8 | DATA8 |

The bits in this register control the function of each port E pin. Any bit set to one configures the corresponding pin as a bus control signal, with the function shown in the following table. Any bit cleared to zero defines the corresponding pin to be an I/O pin, controlled by PORTE and DDRE.

Data bus bit 8 controls the state of this register following reset. If DATA8 is set to one during reset, the register is set to $FF, which defines all port E pins as bus control signals. If DATA8 is cleared to zero during reset, this register is set to $00, configuring all port E pins as I/O pins.

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 a bus control signal.

Table 16 Port E Pin Assignments

PEPAR Bit	Port E Signal	Bus Control Signal
PEPA7	PE7	SIZ1
PEPA6	PE6	SIZ0
PEPA5	PE5	$\overline{AS}$
PEPA4	PE4	$\overline{DS}$
PEPA3	PE3	$\overline{RMC}$
PEPA2	PE2	$\overline{AVEC}$
PEPA1	PE1	$\overline{DSACK1}$
PEPA0	PE0	$\overline{DSACK0}$

PORTF0, PORTF1 — Port F Data Register   **$YFFA19, $YFFA1B**

15		8	7	6	5	4	3	2	1	0
	NOT USED		PF7	PF6	PF5	PF4	PF3	PF2	PF1	PF0

RESET:

| | | | U | U | U | U | U | U | U | U |

The write to the port F data register is stored in the internal data latch, and if any port F pin is configured as an output, the value stored for that bit is driven onto the pin. A read of the port F data register returns the value at the pin only if the pin is configured as a discrete input. Otherwise, the value read is the value stored in the register.

The port F data register is a single register that can be accessed in two locations. When accessed at $YFFA19, the register is referred to as PORTF0; when accessed at $YFFA1B, the register is referred to as PORTF1. The register can be read or written at any time. It is unaffected by reset.

DDRF — Port F Data Direction Register   **$YFFA1D**

15		8	7	6	5	4	3	2	1	0
	NOT USED		DDF7	DDF6	DDF5	DDF4	DDF3	DDF2	DDF1	DDF0

RESET:

| | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

The bits in this register control the direction of the pin drivers when the pins are configured for I/O. Any bit in this register set to one configures the corresponding pin as an output. Any bit in this register cleared to zero configures the corresponding pin as an input.

PFPAR — 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

PFPAR Field	Port F Signal	Alternate Signal
PFPA7	PF7	$\overline{IRQ7}$
PFPA6	PF6	$\overline{IRQ6}$
PFPA5	PF5	$\overline{IRQ5}$
PFPA4	PF4	$\overline{IRQ4}$
PFPA3	PF3	$\overline{IRQ3}$
PFPA2	PF2	$\overline{IRQ2}$
PFPA1	PF1	$\overline{IRQ1}$
PFPA0	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 $\overline{RESET}$ 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 $\overline{RESET}$ 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 $\overline{RESET}$ 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 $\overline{BKPT}$ 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)

4.3 Status Register

The status register contains the condition codes that reflect the results of a previous operation and can be used for conditional instruction execution in a program. The lower byte containing the condition codes is the only portion of the register available at the user privilege level; it is referenced as the condition code register (CCR) in user programs. At the supervisor privilege level, software can access the full status register, including the interrupt priority mask and additional control bits.

SR —Status Register

15	14	13	12	11	10		8	7	6	5	4	3	2	1	0
T1	T0	S	0	0		IP		0	0	0	X	N	Z	V	C

RESET:

| 0 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | U | U | U | U | U |

System Byte
- T[1:0] —Trace Enable
- S —Supervisor/User State
- Bits [12:11] —Unimplemented
- IP[2:0] —Interrupt Priority Mask

User Byte (Condition Code Register)
- Bits [7:5] — Unimplemented
- X — Extend
- N — Negative
- Z — Zero
- V — Overflow
- C — Carry

4.4 Data Types

Six basic data types are supported:

- Bits
- Packed Binary Coded Decimal Digits
- Byte Integers (8 bits)
- Word Integers (16 bits)
- Long-Word Integers (32 bits)
- Quad-Word Integers (64 bits)

4.5 Addressing Modes

Addressing in the CPU32 is register-oriented. Most instructions allow the results of the specified operation to be placed either in a register or directly in memory. This flexibility eliminates the need for extra instructions to store register contents in memory. The CPU32 supports seven basic addressing modes:

- Register direct
- Register indirect
- Register indirect with index
- Program counter indirect with displacement
- Program counter indirect with index
- Absolute
- Immediate

Included in the register indirect addressing modes are the capabilities to post-increment, predecrement, and offset. The program counter relative mode also has index and offset capabilities. In addition to these addressing modes, many instructions implicitly specify the use of the status register, stack pointer, or program counter.

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	Dyn – Dym $\Rightarrow$ Temp (Temp $*$ Dn [7 : 0]) $\Rightarrow$ Temp (Dym $*$ 256) + Temp $\Rightarrow$ Dn
TBLSN/TBLUN	<ea>, Dn Dym : Dyn, Dn	8, 16, 32	Dyn – Dym $\Rightarrow$ Temp (Temp $*$ Dn [7 : 0]) / 256 $\Rightarrow$ Temp Dym + Temp $\Rightarrow$ Dn
TRAP	#<data>	none	SSP – 2 $\Rightarrow$ SSP; format/vector offset $\Rightarrow$ (SSP); SSP – 4 $\Rightarrow$ SSP; PC $\Rightarrow$ (SSP); SR $\Rightarrow$ (SSP); vector address $\Rightarrow$ 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	An $\Rightarrow$ SP; (SP) $\Rightarrow$ An, SP + 4 $\Rightarrow$ SP

1. Privileged instruction.

5 Time Processor Unit

The time processor unit (TPU) provides optimum performance in controlling time-related activity. The TPU contains a dedicated execution unit, a tri-level prioritized scheduler, data storage RAM, dual-time bases, and microcode ROM. The TPU controls 16 independent, orthogonal channels, each with an associated I/O pin, and is capable of performing any microcoded time function. Each channel contains dedicated hardware that allows input or output events to occur simultaneously on all channels.
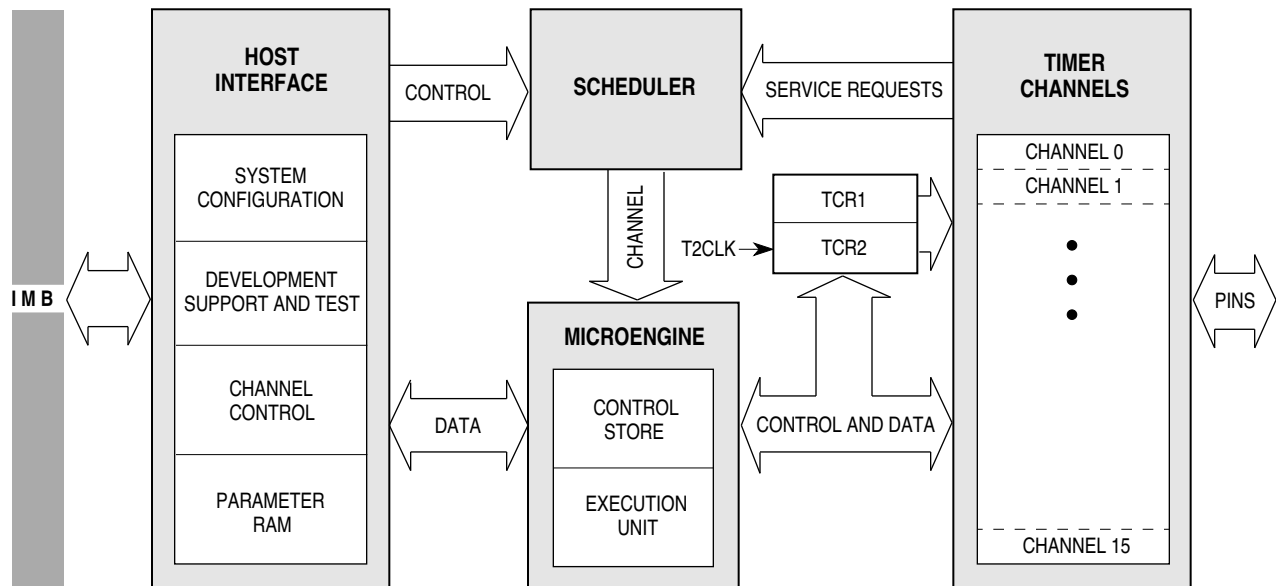
Figure 12 TPU Block Diagram

5.1 MC68332 and MC68332A Time Functions

The following paragraphs describe factory-programmed time functions implemented in standard and enhanced standard TPU microcode ROM. A complete description of the functions is beyond the scope of this summary. Refer to *Using the TPU Function Library and TPU Emulation Mode* (TPUPN00/D) as well as other TPU programming notes for more information about specific functions.

5.1.1 Discrete Input/Output (DIO)

When a pin is used as a discrete input, a parameter indicates the current input level and the previous 15 levels of a pin. Bit 15, the most significant bit of the parameter, indicates the most recent state. Bit 14 indicates the next most recent state, and so on. The programmer can choose one of the three following conditions to update the parameter: 1) when a transition occurs, 2) when the CPU makes a request, or 3) when a rate specified in another parameter is matched. When a pin is used as a discrete output, it is set high or low only upon request by the CPU.

5.2.3 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.

5.2.4 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).

5.2.5 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.

5.2.6 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 ICTC function.

5.2.7 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.

5.2.8 Brushless Motor Commutation (COMM)

This function generates the phase commutation signals for a variety of brushless motors, including three-phase brushless direct current. It derives the commutation state directly from the position decoded in FQD, thus eliminating the need for hall effect sensors.

The state sequence is implemented as a user-configurable state machine, thus providing a flexible approach with other general applications. A CPU offset parameter is provided to allow all the switching angles to be advanced or retarded on the fly by the CPU. This feature is useful for torque maintenance at high speeds.

TCR1P — Timer Count Register 1 Prescaler Control

TCR1 is clocked from the output of a prescaler. The prescaler's input is the internal TPU system clock divided by either 4 or 32, depending on the value of the PSCK bit. The prescaler divides this input by 1, 2, 4, or 8. Channels using TCR1 have the capability to resolve down to the TPU system clock divided by 4.
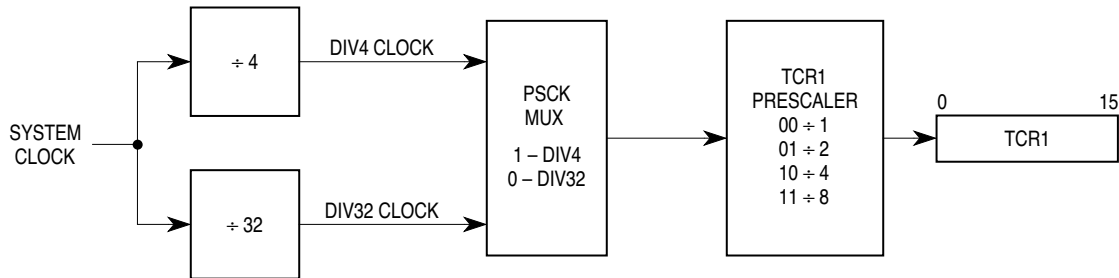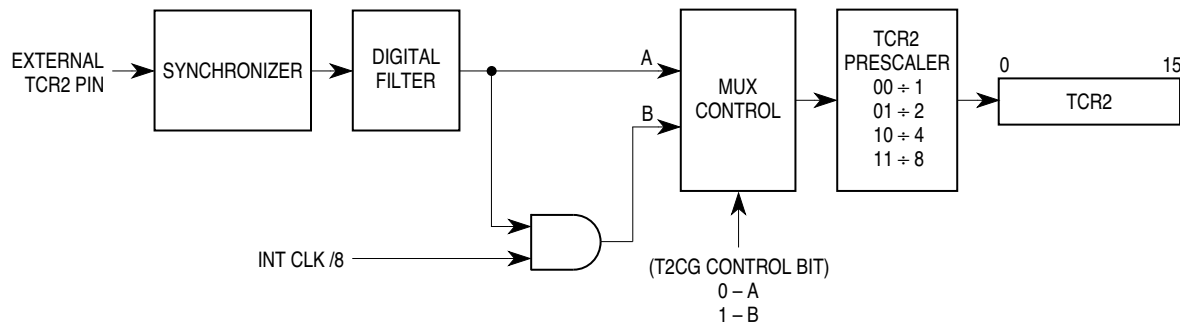

PRESCALER CTL BLOCK 1

TCR1 Prescaler	Divide By	PSCK = 0		PSCK = 1	
		Number of Clocks	Rate at 16 MHz	Number of Clocks	Rate at 16 MHz
00	1	32	2 ms	4	250 ns
01	2	64	4 ms	8	500 ns
10	4	128	8 ms	16	1 ms
11	8	256	16 ms	32	2 ms

TCR2P — Timer Count Register 2 Prescaler Control

TCR2 is clocked from the output of a prescaler. If T2CG = 0, the input to the TCR2 prescaler is the external TCR2 clock source. If T2CG = 1, the input is the TPU system clock divided by eight. The TCR2P field specifies the value of the prescaler: 1, 2, 4, or 8. Channels using TCR2 have the capability to resolve down to the TPU system clock divided by 8. The following table is a summary of prescaler output.

PRESCALER CTL BLOCK 2

TCR2 Prescaler	Divide By	Internal Clock Divided By	External Clock Divided By
00	1	8	1
01	2	16	2
10	4	32	4
11	8	64	8

6 Queued Serial Module

The QSM contains two serial interfaces, the queued serial peripheral interface (QSPI) and the serial communication interface (SCI).
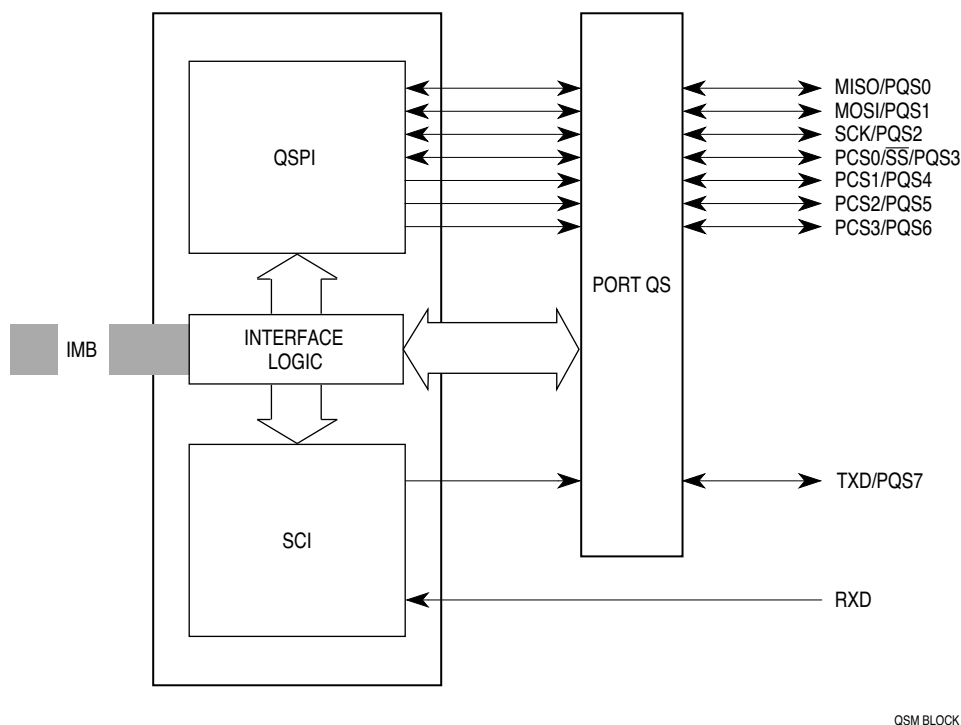

Figure 13 QSM Block Diagram

6.1 Overview

The QSPI provides easy peripheral expansion or interprocessor communication through a full-duplex, synchronous, three-line bus: data in, data out, and a serial clock. Four programmable peripheral chip-select pins provide addressability for up to 16 peripheral devices. A self-contained RAM queue allows up to 16 serial transfers of 8 to 16 bits each, or transmission of a 256-bit data stream without CPU intervention. A special wraparound mode supports continuous sampling of a serial peripheral, with automatic QSPI RAM updating, which makes the interface to A/D converters more efficient.

The SCI provides a standard nonreturn to zero (NRZ) mark/space format. It operates in either full- or half-duplex mode. There are separate transmitter and receiver enable bits and dual data buffers. A modulus-type baud rate generator provides rates from 64 to 524 kbaud with a 16.78-MHz system clock, or 110 to 655 kbaud with a 20.97-MHz system clock. Word length of either 8 or 9 bits is software selectable. Optional parity generation and detection provide either even or odd parity check capability. Advanced error detection circuitry catches glitches of up to 1/16 of a bit time in duration. Wakeup functions allow the CPU to run uninterrupted until meaningful data is available.

An address map of the QSM is shown below.

6.2 Address Map

The "Access" column in the QSM 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 QSMCR.

Table 24 QSM Address Map

Access	Address	15 8	7 0
S	$YFFC00	QSM MODULE CONFIGURATION (QSMCR)	
S	$YFFC02	QSM TEST (QTEST)	
S	$YFFC04	QSM INTERRUPT LEVEL (QILR)	QSM INTERRUPT VECTOR (QIVR)
S/U	$YFFC06	NOT USED	
S/U	$YFFC08	SCI CONTROL 0 (SCCR0)	
S/U	$YFFC0A	SCI CONTROL 1 (SCCR1)	
S/U	$YFFC0C	SCI STATUS (SCSR)	
S/U	$YFFC0E	SCI DATA (SCDR)	
S/U	$YFFC10	NOT USED	
S/U	$YFFC12	NOT USED	
S/U	$YFFC14	NOT USED	PQS DATA (PORTQS)
S/U	$YFFC16	PQS PIN ASSIGNMENT (PQSPAR)	PQS DATA DIRECTION (DDRQS)
S/U	$YFFC18	SPI CONTROL 0 (SPCR0)	
S/U	$YFFC1A	SPI CONTROL 1 (SPCR1)	
S/U	$YFFC1C	SPI CONTROL 2 (SPCR2)	
S/U	$YFFC1E	SPI CONTROL 3 (SPCR3)	SPI STATUS (SPSR)
S/U	$YFFC20–$YFFCFF	NOT USED	
S/U	$YFFD00–$YFFD1F	RECEIVE RAM (RR[0:F])	
S/U	$YFFD20–$YFFD3F	TRANSMIT RAM (TR[0:F])	
S/U	$YFFD40–$YFFD4F	COMMAND RAM (CR[0:F])	

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

Table 25 QSPAR Pin Assignments

PQSPAR Field	PQSPAR Bit	Pin Function
PQSPA0	0	PQS0
	1	MISO
PQSPA1	0	PQS1
	1	MOSI
PQSPA2	0	PQS2[1]
	1	SCK
PQSPA3	0	PQS3
	1	PCS0/$\overline{SS}$
PQSPA4	0	PQS4
	1	PCS1
PQSPA5	0	PQS5
	1	PCS2
PQSPA6	0	PQS6
	1	PCS3
PQSPA7	0	PQS7[2]
	1	TXD

NOTES:

1. PQS2 is a digital I/O pin unless the SPI is enabled (SPE in SPCR1 set), in which case it becomes SPI serial clock SCK.

2. PQS7 is a digital I/O pin unless the SCI transmitter is enabled (TE in SCCR1 = 1), in which case it becomes SCI serial output TXD.

DDRQS determines whether pins are inputs or outputs. Clearing a bit makes the corresponding pin an input; setting a bit makes the pin an output. DDRQS affects both QSPI function and I/O function.

Table 26 Effect of DDRQS on QSM Pin Function

QSM Pin	Mode	DDRQS Bit	Bit State	Pin Function
MISO	Master	DDQ0	0	Serial Data Input to QSPI
			1	Disables Data Input
	Slave		0	Disables Data Output
			1	Serial Data Output from QSPI
MOSI	Master	DDQ1	0	Disables Data Output
			1	Serial Data Output from QSPI
	Slave		0	Serial Data Input to QSPI
			1	Disables Data Input
SCK[1]	Master	DDQ2	0	Disables Clock Output
			1	Clock Output from QSPI
	Slave		0	Clock Input to QSPI
			1	Disables Clock Input
PCS0/$\overline{SS}$	Master	DDQ3	0	Assertion Causes Mode Fault
			1	Chip-Select Output
	Slave		0	QSPI Slave Select Input
			1	Disables Select Input
PCS[3:1]	Master	DDQ[4:6]	0	Disables Chip-Select Output
			1	Chip-Select Output
	Slave		0	Inactive
			1	Inactive
TXD[2]	Transmit	DDQ7	X	Serial Data Output from SCI
RXD	Receive	None	NA	Serial Data Input to SCI

NOTES:
1. PQS2 is a digital I/O pin unless the SPI is enabled (SPE in SPCR1 set), in which case it becomes SPI serial clock SCK.
2. PQS7 is a digital I/O pin unless the SCI transmitter is enabled (TE in SCCR1 = 1), in which case it becomes SCI serial output TXD.

DDRQS determines the direction of the TXD pin only when the SCI transmitter is disabled. When the SCI transmitter is enabled, the TXD pin is an output.

Freescale Semiconductor, Inc.

8 Summary of Changes

This is a partial revision. Most of the publication remains the same, but the following changes were made to improve it. Typographical errors that do not affect content are not annotated. This document has also been reformatted for use on the web.

Pages 2-3 New Ordering Information included.

Page 6 New block diagram drawn.

Page 7 New 132-pin assignment diagram drawn.

Page 8 New 144-pin assignment diagram drawn.

Page 9 New address map drawn.

Pages 10-14 Added Signal Description section.

Pages 15-47 Expanded and revised SIM section. Made all register diagrams and bit mnemonics consistent. Incorporated new information concerning the system clock, resets, interrupts, and chip-selects circuits.

Page 48-56 Expanded and revised CPU section. Made all register diagrams and bit mnemonics consistent. Revised instruction set summary information.

Page 57-70 Expanded and revised TPU section. Made all register diagrams and bit mnemonics consistent. Revised time functions information to include both MC68332A and MC68332G microcode ROM applications.

Page 71-92 Expanded and revised QSM section. Made all register diagrams and bit mnemonics consistent. Added information concerning SPI and SCI operation.

Page 93-95 Revised Standby RAM with TPU Emulation RAM section. Made all register diagrams and bit mnemonics consistent.