Welcome to E-XFL.COM

Understanding Embedded - Microprocessors

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

Applications of Embedded - Microprocessors

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

Details	
Product Status	Obsolete
Core Processor	CPU32
Number of Cores/Bus Width	1 Core, 32-Bit
Speed	25MHz
Co-Processors/DSP	-
RAM Controllers	DRAM
Graphics Acceleration	No
Display & Interface Controllers	-
Ethernet	-
SATA	-
USB	-
Voltage - I/O	5.0V
Operating Temperature	0°C ~ 70°C (TA)
Security Features	-
Package / Case	144-BCQFP
Supplier Device Package	144-CQFP (26.77x26.77)
Purchase URL	https://www.e-xfl.com/pro/item?MUrl=&PartUrl=mc68340fe25e

Freescale Semiconductor, Inc.

TABLE OF CONTENTS (Continued)

Section 6
DMA Controller Module

Freescale Semiconductor, Inc.

For More Information On This Product,
Go to: www.freescale.com

Freescale Semiconductor, Inc.

TABLE OF CONTENTS (Continued)

Freescale Semiconductor, Inc.

For More Information On This Product,
Go to: www.freescale.com

To use an external clock source (see Figure 4-6), the operating clock frequency can be driven directly into the EXTAL pin (the XTAL pin must be left floating for this case). This approach results in a system clock and CLKOUT that are the same as the input signal frequency, but not tightly coupled to it. To enable this mode, MODCK must be held low during reset, and V_{CCSYN} held at 0 V while the chip is in operation.

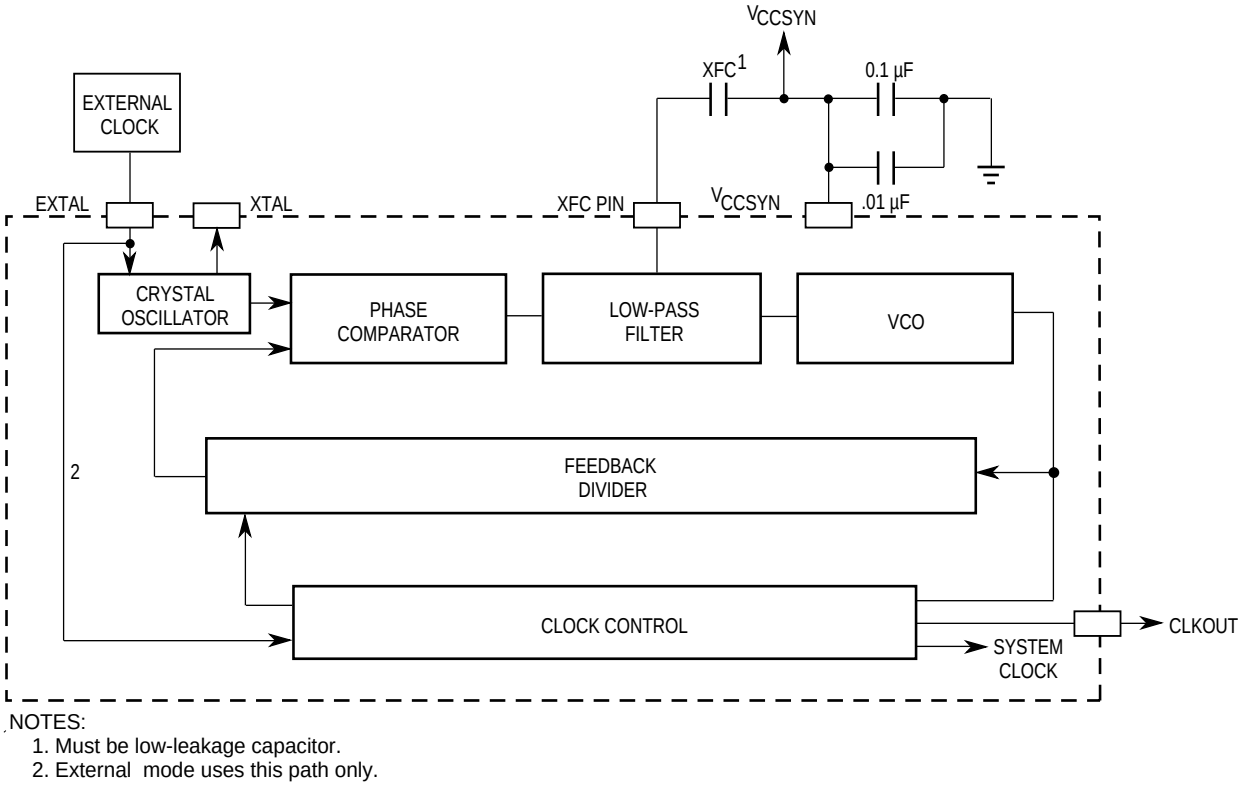


Figure 4-6. Clock Block Diagram for External Oscillator Operation

Alternatively, an external clock signal can be directly driven into EXTAL (with XTAL left floating) using the on-chip PLL. This configuration results in an internal clock and CLKOUT signal of the same frequency as the input signal, with a tight skew between the external clock and the internal clock and CLKOUT signals. To enable this mode, MODCK must be held low during reset, and V_{CCSYN} should be connected to a quiet 5-V source.

If an input signal loss for either of the clock modes utilizing the PLL occurs, chip operation can continue in limp mode with the VCO running at approximately one-half the operating speed (affected by the value of the X-bit in the SYNCR), using an internal voltage reference. The SLIMP bit in the SYNCR indicates that a loss of input signal reference has been detected. The RSTEN bit in the SYNCR controls whether an input signal loss causes a system reset or causes the device to operate in limp mode. The SLOCK bit in the SYNCR indicates when the VCO has locked onto the desired frequency or if an external clock is being used.

4.2.3.1 PHASE COMPARATOR AND FILTER. The phase comparator takes the output of the frequency divider and compares it to an external input signal reference. The result of

For More Information On This Product,
Go to: www.freescale.com

Freescale Semiconductor, Inc.

DSCLK, the gated serial clock, is normally high, but it pulses low for each bit to be transferred. At the end of the seventeenth clock period, it remains high until the start of the next transmission. Clock frequency is implementation dependent and may range from DC to the maximum specified frequency. Although performance considerations might dictate a hardware implementation, software solutions can be used provided serial bus timing is maintained.

5.6.2.8 COMMAND SET. The following paragraphs describe the command set available in BDM.

5.6.2.8.1 Command Format. The following standard bit format is utilized by all BDM commands.

15 — 10	9	8	7 — 6	5	4	3	2 — 0
OPERATION	0	R/W	OP SIZE	0	0	A/D	REGISTER
EXTENSION WORD(S)							

Bits 15–0—Operation Field

The operation field specifies the commands. This 6-bit field provides for a maximum of 64 unique commands.

R/W Field

The R/W field specifies the direction of operand transfer. When the bit is set, the transfer is from CPU to development system. When the bit is cleared, data is written to the CPU or to memory from the development system.

Operand Size

For sized operations, this field specifies the operand data size. All addresses are expressed as 32-bit absolute values. The size field is encoded as listed in Table 5-22.

Table 5-22. Size Field Encoding

Encoding	Operand Size
00	Byte
01	Word
10	Long
11	Reserved

Address/Data (A/D) Field

The A/D field is used by commands that operate on address and data registers. It determines whether the register field specifies a data or address register. One indicates an address register; zero indicates a data register. For other commands, this field may be interpreted differently.

For More Information On This Product,
Go to: www.freescale.com

Freescale Semiconductor, Inc.

starting address of the block and to retrieve the first result. Subsequent operands are retrieved with the DUMP command. The initial address is incremented by the operand size (1, 2, or 4) and saved in a temporary register. Subsequent DUMP commands use this address, increment it by the current operand size, and store the updated address back in the temporary register.

NOTE

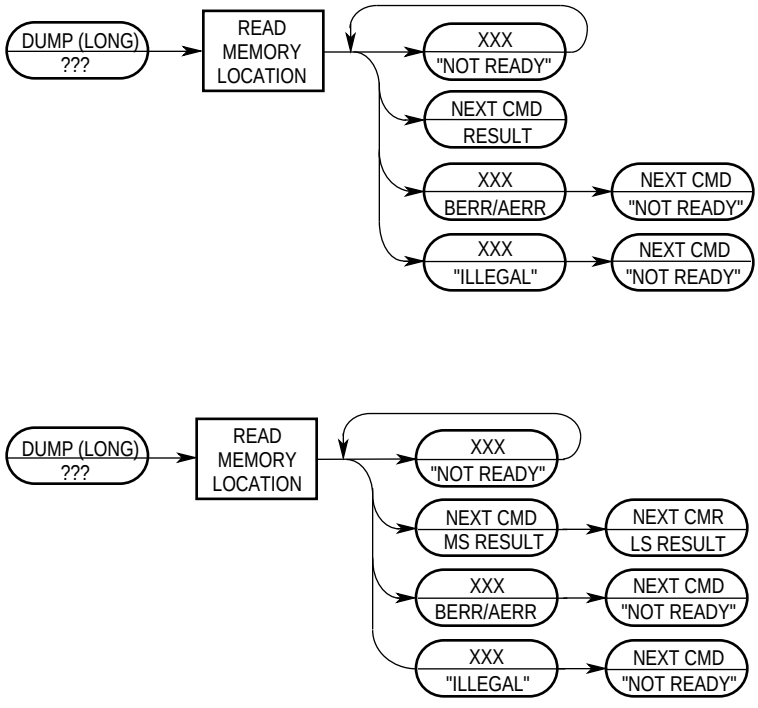
The DUMP command does not check for a valid address in the temporary register—DUMP is a valid command only when preceded by another DUMP or by a READ command. Otherwise, the results are undefined. The NOP command can be used for intercommand padding without corrupting the address pointer.

The size field is examined each time a DUMP command is given, allowing the operand size to be altered dynamically.

Command Format:

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	1	1	1	0	1	OP SIZE		0	0	0	0	0	0

Command Sequence:



Freescale Semiconductor, Inc.

For More Information On This Product,
Go to: www.freescale.com

The bus controller and microsequencer operate concurrently. The bus controller can perform a read or write or schedule a prefetch while the microsequencer controls EA calculation or sets condition codes.

The microsequencer can also request a bus cycle that the bus controller cannot perform immediately. When this happens, the bus cycle is queued, and the bus controller runs the cycle when the current cycle is complete.

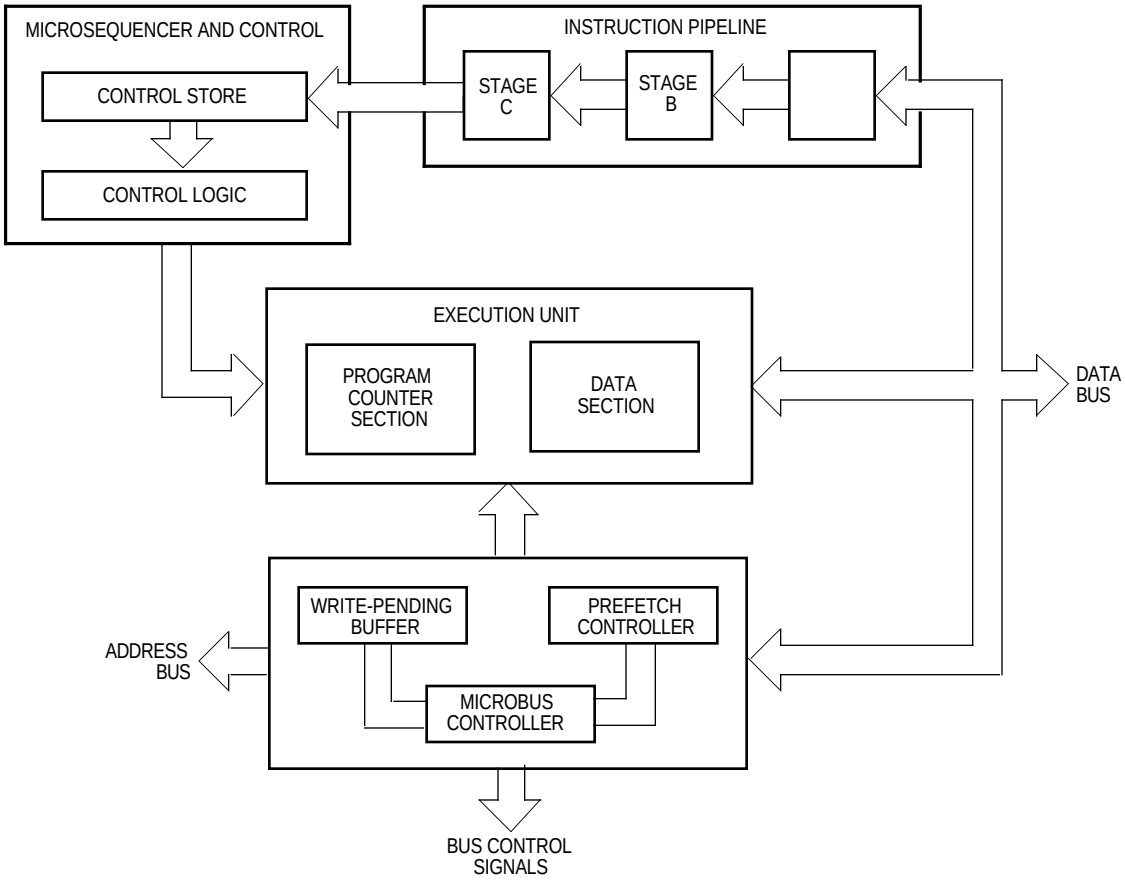


Figure 5-30. Block Diagram of Independent Resources

5.7.1.3.1 Prefetch Controller. The instruction prefetch controller receives an initial request from the microsequencer to initiate prefetching at a given address. Subsequent prefetches are initiated by the prefetch controller whenever a pipeline stage is invalidated, either through instruction completion or through use of extension words. Prefetch occurs as soon as the bus is free of operand accesses previously requested by the microsequencer. Additional state information permits the controller to inhibit prefetch requests when a change in instruction flow (e.g., a jump or branch instruction) is anticipated.

In a typical program, 10 to 25 percent of the instructions cause a change of flow. Each time a change occurs, the instruction pipeline must be flushed and refilled from the new instruction stream. If instruction prefetches, rather than operand accesses, were given

For More Information On This Product,
Go to: www.freescale.com

Freescale Semiconductor, Inc.

SECTION 6
DMA CONTROLLER MODULE

The direct memory access (DMA) controller module provides for high-speed transfer capability to/from an external peripheral or for memory-to-memory data transfer. The DMA module, shown in Figure 6-1, provides two channels that allow byte, word, or long-word operand transfers. These transfers can be either single or dual address and to either on- or off-chip devices. The DMA contains the following features:

- Two, Independent, Fully Programmable DMA Channels
- Single-Address Transfers with 32-Bit Address and 32-Bit Data Capability
- Dual-Address Transfers with 32-Bit Address and 16-Bit Data Capability
- Two 32-Bit Transfer Counters
- Four 32-Bit Address Pointers That Can Increment or Remain Constant
- Operand Packing and Unpacking for Dual-Address Transfers
- Supports All Bus-Termination Modes
- Provides Two-Clock-Cycle Internal Module Access
- Provides Two-Clock-Cycle External Access Using MC68340 Chip Selects
- Provides Full DMA Handshake for Burst Transfers and Cycle Steal

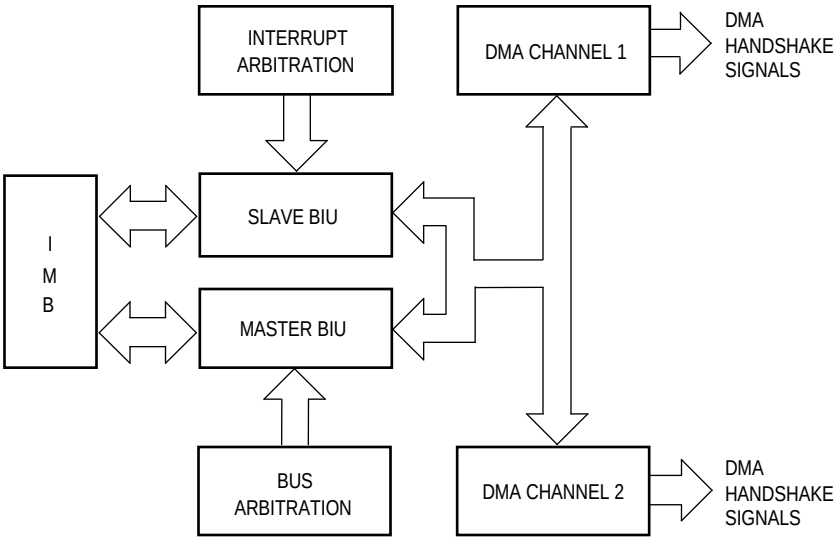


Figure 6-1. DMA Block Diagram

Freescale Semiconductor, Inc.

For More Information On This Product,
Go to: www.freescale.com

6.2 DMA MODULE SIGNAL DEFINITIONS

This section contains a brief description of the DMA module signals used to provide handshake control for either a source or destination external device.

NOTE

The terms *assertion* and *negation* are used throughout this section to avoid confusion when dealing with a mixture of active-low and active-high signals. The term *assert* or *assertion* indicates that a signal is active or true, independent of the level represented by a high or low voltage. The term *negate* or *negation* indicates that a signal is inactive or false.

6.2.1 DMA Request (DREQ≈)

This active-low input is asserted by a peripheral device to request an operand transfer between that peripheral and memory. The assertion of DREQ≈ starts the DMA process. The assertion level in external burst mode is level sensitive; in external cycle steal mode, it is falling-edge sensitive.

6.2.2 DMA Acknowledge (DACK≈)

This active-low output is asserted by the DMA to signal to a peripheral that an operand is being transferred in response to a previous transfer request.

6.2.3 DMA Done (DONE≈)

This active-low bidirectional signal is asserted by the DMA or a peripheral device during any DMA bus cycle to indicate that the last data transfer is being performed. DONE≈ is an active input in any mode. As an output, DONE≈ is only active in external request mode. An external pullup resistor is required even if operating only in the internal request mode.

6.3 TRANSFER REQUEST GENERATION

The DMA channel supports two types of request generation methods: internal and external. Internally generated requests can be programmed to limit the amount of bus utilization. Externally generated requests can be either burst mode or cycle steal mode. The request generation method used for the channel is programmed by the channel control register (CCR) in the REQ field.

6.3.1 Internal Request Generation

Internal requests are accessed in two clocks by the intermodule bus (IMB). The channel is started as soon as the STR bit in the CCR is set. The channel immediately requests the bus and begins transferring data. Only internal requests can limit the amount of bus utilization. The percentage of the bandwidth that the DMA channel can use during a transfer can be selected by the CCR BB field.

For More Information On This Product,
Go to: www.freescale.com

Freescale Semiconductor, Inc.

```
        MOVE.W      #$0742,DMAINT1(A0)

* Initialize channel status reg.
* Clear the DONE, BES, BED, CONF and BRKP bits to allow channel to startup.
        MOVE.B      #$7C,DMACSR1(A0)

* Initialize function code reg.
* DMA space, supervisor data space for source and destination.
        MOVE.B      #$DD,DMAFCR1(A0)

* Initialize source operand address
* Source address is equal to $6000.
        MOVE.L      SARADD,DMASAR1(A0)

* Initialize destination operand address
* Destination address is equal to $8000.
        MOVE.L      DARADD,DMADAR1(A0)

* Initialize the byte transfer count register
* The number of bytes to be transferred is $64 or 50 words
        MOVE.L      NUMBYTE,DMABTC1(A0)

* Channel control reg. init. and Start DMA transfers
* No interrupts are enabled, destination (write) cycle.
* Source address is not incremented. Increment the destination address.
* Source size is word, destination size is word. REQ is internal.
* 100% of bus bandwidth, dual-address transfers, start the DMA transfers.
        MOVE.W      #$068D,DMACCR1(A0)

*************************************************************************
        END
*************************************************************************
```

Example 4: Cycle Steal Request Generation, Dual-Address Transfers.

```
*************************************************************************
* MC68340 basic DMA channel register initialization example code.
* This code is used to initialize the 68340's internal DMA channel
* registers, providing basic functions for operation.
* The code sets up channel 1 for external cycle steal request generation,
* dual-address transfers. DMA 16-bit wide data from an odd address to an
* even address. Control signals are asserted on the DMA read cycle.
*************************************************************************
*************************************************************************
* SIM40 equates
*************************************************************************
MBAR        EQU   $0003FF00   Address of SIM40 Module Base Address Reg.
MODBASE     EQU   $FFFFF000   SIM40 MBAR address value
```

Freescale Semiconductor, Inc.

For More Information On This Product,
Go to: www.freescale.com

7.1.1 Serial Communication Channels A and B

Each communication channel provides a full-duplex asynchronous/synchronous receiver and transmitter using an operating frequency independently selected from a baud rate generator or an external clock input.

The transmitter accepts parallel data from the IMB, converts it to a serial bit stream, inserts the appropriate start, stop, and optional parity bits, then outputs a composite serial data stream on the channel transmitter serial data output (TxDx). Refer to **7.3.2.1 Transmitter** for additional information.

The receiver accepts serial data on the channel receiver serial data input (RxDx), converts it to parallel format, checks for a start bit, stop bit, parity (if any), or break condition, and transfers the assembled character onto the IMB during read operations. Refer to **7.3.2.2 Receiver** for additional information.

7.1.2 Baud Rate Generator Logic

The crystal oscillator operates directly from a 3.6864-MHz crystal connected across the X1 input and the X2 output or from an external clock of the same frequency connected to X1. The clock serves as the basic timing reference for the baud rate generator and other internal circuits.

The baud rate generator operates from the oscillator or external TTL clock input and is capable of generating 19 commonly used data communication baud rates ranging from 50 to 76.8k by producing internal clock outputs at 16 times the actual baud rate. Refer to **7.2 Serial Module Signal Definitions** and **7.3.1 Baud Rate Generator** for additional information.

The external clock input (SCLK), which bypasses the baud rate generator, provides a synchronous clock mode of operation when used as a divide-by-1 clock and an asynchronous clock mode when used as a divide-by-16 clock. The external clock input allows the user to use SCLK as the only clock source for the serial module if multiple baud rates are not required.

7.1.3 Internal Channel Control Logic

The serial module receives operation commands from the host and, in turn, issues appropriate operation signals to the internal serial module control logic. This mechanism allows the registers within the module to be accessed and various commands to be performed. Refer to **7.4 Register Description and Programming** for additional information.

7.1.4 Interrupt Control Logic

Seven interrupt request (IRQ7–IRQ1) signals are provided to notify the CPU32 that an interrupt has occurred. These interrupts are described in **7.4 Register Description and Programming**. The interrupt status register (ISR) is read by the CPU32 to determine all

Freescale Semiconductor, Inc.

For More Information On This Product,
Go to: www.freescale.com

assembled in the receiver shift register and loaded into the top empty receiver holding register position of the FIFO. Thus, data flowing from the receiver to the CPU32 is quadruple buffered.

In addition to the data byte, three status bits, PE, FE, and RB, are appended to each data character in the FIFO; OE is not appended. By programming the ERR bit in the channel's mode register (MR1), status is provided in character or block modes.

The RxRDY bit in the SR is set whenever one or more characters are available to be read by the CPU32. A read of the receiver buffer produces an output of data from the top of the FIFO stack. After the read cycle, the data at the top of the FIFO stack and its associated status bits are 'popped', and new data can be added at the bottom of the stack by the receiver shift register. The FIFO-full status bit (FFULL) is set if all three stack positions are filled with data. Either the RxRDY or FFULL bit can be selected to cause an interrupt.

In the character mode, status provided in the SR is given on a character-by-character basis and thus applies only to the character at the top of the FIFO. In the block mode, the status provided in the SR is the logical OR of all characters coming to the top of the FIFO stack since the last reset error command. A continuous logical OR function of the corresponding status bits is produced in the SR as each character reaches the top of the FIFO stack. The block mode is useful in applications where the software overhead of checking each character's error cannot be tolerated. In this mode, entire messages are received, and only one data integrity check is performed at the end of the message. This mode allows a data-reception speed advantage, but does have a disadvantage since each character is not individually checked for error conditions by software. If an error occurs within the message, the error is not recognized until the final check is performed, and no indication exists as to which character in the message is at fault.

In either mode, reading the SR does not affect the FIFO. The FIFO is 'popped' only when the receive buffer is read. The SR should be read prior to reading the receive buffer. If all three of the FIFO's receiver holding registers are full when a new character is received, the new character is held in the receiver shift register until a FIFO position is available. If an additional character is received during this state, the contents of the FIFO are not affected. However, the character previously in the receiver shift register is lost, and the OE bit in the SR is set when the receiver detects the start bit of the new overrunning character.

To support control flow capability, the receiver can be programmed to automatically negate and assert RTS≈. When in this mode, RTS≈ is automatically negated by the receiver when a valid start bit is detected and the FIFO stack is full. When a FIFO position becomes available, RTS≈ is asserted by the receiver. Using this mode of operation, overrun errors are prevented by connecting the RTS≈ to the CTS≈ input of the transmitting device.

If the FIFO stack contains characters and the receiver is disabled, the characters in the FIFO can still be read by the CPU32. If the receiver is reset, the FIFO stack and all receiver status bits, corresponding output ports, and interrupt request are reset. No additional characters are received until the receiver is re-enabled.

Freescale Semiconductor, Inc.

For More Information On This Product,
Go to: www.freescale.com

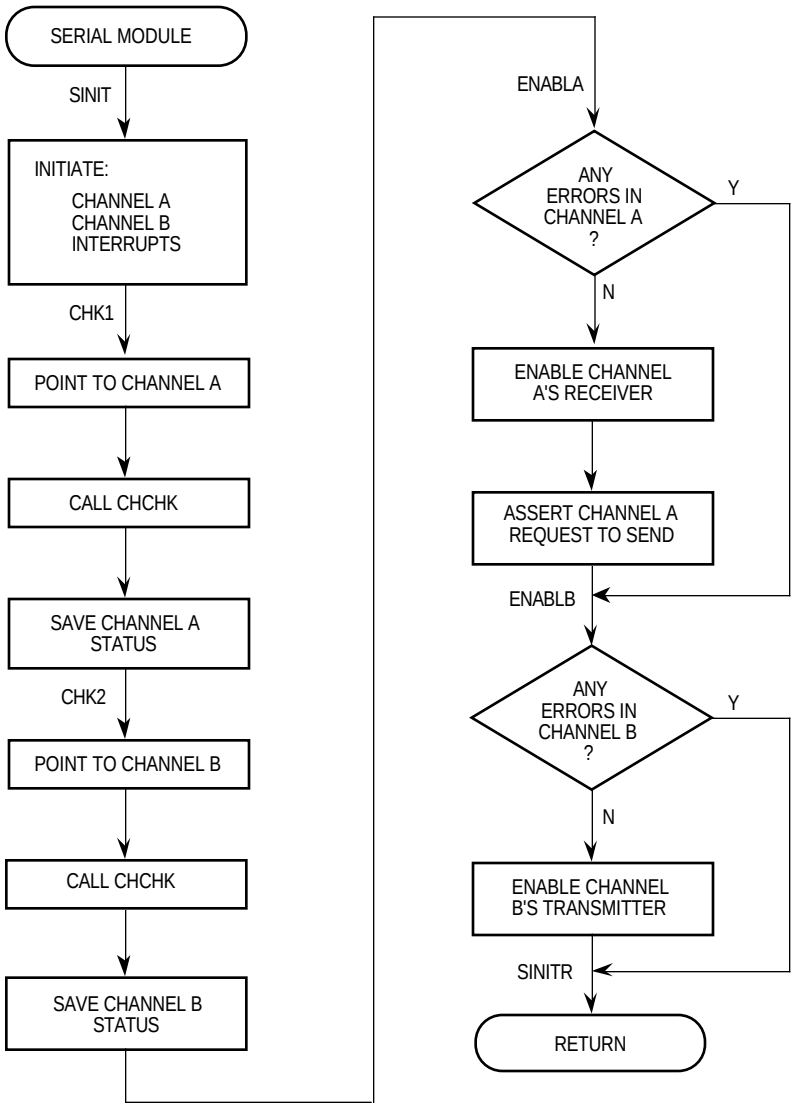


Figure 7-10. Serial Module Programming Flowchart (1 of 5)

Freescale Semiconductor, Inc.

For More Information On This Product,
Go to: www.freescale.com

reasserted, the timer is re-enabled and begins counting from the value attained when TGATE≈ was negated. The SR ON bit is set again.

If TGATE≈ is disabled (TGE = 0), TGATE≈ has no effect on the operation of the timer. In this case, the counter begins counting on the falling edge of the counter clock immediately after the SWR and CPE bits in the CR are set. The TG bit of the SR cannot be set. At all times, TGL in the SR reflects the level of TGATE≈.

If the counter counts down to the value stored in the COM register, then the COM and TC bits in the SR are set. The counter continues counting down to timeout. At this time, the SR TO bit is set, and the SR COM bit is cleared. The next falling edge of the counter clock after timeout causes the value in PREL1 to be loaded back into the counter, and the counter begins counting down from this value.

The period of the square-wave generator can be changed dynamically by writing a new value into the PREL1. Caution must be used because, if PREL1 is accessed simultaneously by the counting logic and a CPU32 write, the old PREL1 value may actually get loaded into the counter at timeout.

Periodic interrupt generation can be accomplished by enabling the TO, TG, and/or TC bits in the SR to generate interrupts by programming the CR IE bits. When enabled, the programmed IRQ≈ signal is asserted whenever the specified bits are set.

8.3.3 Variable Duty-Cycle Square-Wave Generator

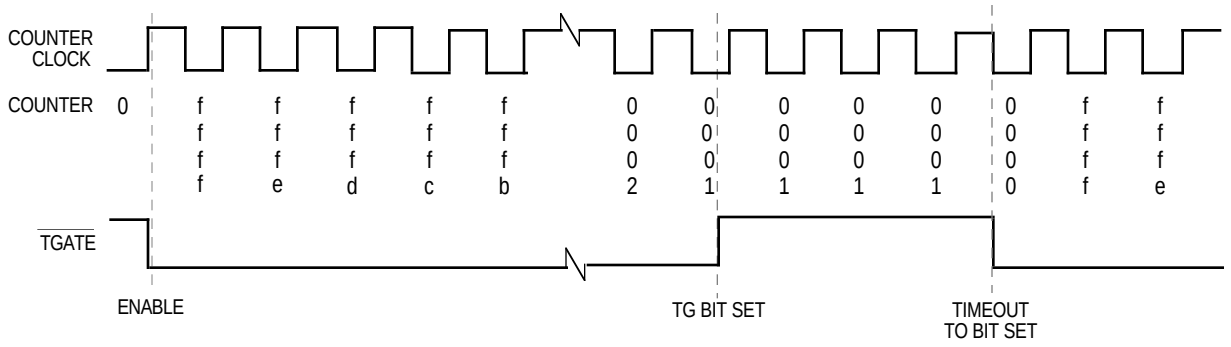
In this mode, both the PREL1 and PREL2 registers are used to generate a square wave with virtually any duty cycle. The square wave is generated by counting down from the value in the PREL1 to timeout (count value $0000), then loading that value from PREL2 and again counting down to timeout. When this second timeout occurs, the value from PREL1 is loaded into the counter, and the cycle repeats. TOUTx can be programmed to change state with every timeout, thus generating a variable duty-cycle square wave. This mode can be selected by programming the MODE bits in the CR to 010.

The timer is enabled by setting both the SWR and CPE bits in the CR and, if TGATE≈ is enabled (CR TGE bit is set), then asserting TGATE≈. When the timer is enabled, the ON bit in the SR is set. On the next falling edge of the counter clock, the counter is loaded with the value stored in the PREL1 register (N1). With each successive falling edge of the counter clock, the counter decrements. The time between enabling the timer and the first timeout can range from N1 to N1+1 periods. When TGATE≈ is used to enable the timer, the enabling of the timer is asynchronous; however, if timing is carefully considered, the time to the first timeout can be known. For additional details on timing, see the **Section 11 Electrical Characteristics**.

If the counter counts down to the value stored in the COM register, the COM and timer compare interrupt (TC) bits in the SR are set. The counter continues counting down to timeout. At this time, the TO bit in the SR is set, and the COM bit is cleared. The next falling edge of the counter clock after timeout causes the value in PREL2 (N2) to be loaded into the counter, and the counter begins counting down from this value. Each

For More Information On This Product,
Go to: www.freescale.com

Freescale Semiconductor, Inc.



MODEx Bits in Control Register = 110
TGE Bit of the Control Register = 1

Figure 8-10. Event Count Mode

The timer is enabled by setting the SWR and CPE bits in the CR and, if TGATE≈ is enabled (TGE bit of the CR is set), then asserting TGATE≈. When the timer is enabled, the SR ON bit is set. On the next falling edge of the counter clock, the counter is loaded with the value of $FFFF. With each successive falling edge of the counter clock, the counter decrements. The PREL1 and PREL2 registers are not used in this mode.

If TGATE≈ is not enabled (CR TGE bit is cleared), then TGATE≈ does not start or stop the timer or affect the TG bit of the SR. In this case, the counter would begin counting on the falling edge of the counter clock immediately after the SWR and CPE bits in the CR are set.

If TGATE≈ is enabled (CR TGE bit is set), then the assertion of TGATE≈ starts the counter. The negation of TGATE≈ disables the counter, sets the SR TG bit, and clears the ON bit in the SR. If TGATE≈ is reasserted, the timer resumes counting from where it was stopped, and the ON bit is set again. Further assertions and negations of TGATE≈ have the same effect. The TGL bit in the SR reflects the level of TGATE≈ at all times.

If the counter counts down to the value stored in the COM register, the COM and TC bits in the SR are set. If the counter counts down to $0000, a timeout is detected. This event sets the TO in the SR and clears the COM bit. At timeout, the next falling edge of the counter clock reloads the counter with $FFFF. TOUTx transitions at timeout or is disabled as programmed by the CR OC bits. The SR OUT bit reflects the level on TOUTx.

To determine the number of cycles counted, the value in the CNTR must be read, inverted, and incremented by 1 (the first count is $FFFF which, in effect, includes a count of zero). The counter counts in a true 2^{16} fashion. For measuring pulses of even greater duration, the value in the POx bits in the SR are readable and can be thought of as an extension of the least significant bits in the CNTR.

Freescale Semiconductor, Inc.

For More Information On This Product,
Go to: www.freescale.com

POT2–POT0—Prescaler Output Tap

If PCLK is set, these bits encode which of the prescaler's output taps act as the counter clock. A division of the selected clock is applied to the counter as listed in Table 8-4.

Table 8-4. POT Encoding

POT2	POT1	POT0	Division of Selected Clock
0	0	1	Divide by 2
0	1	0	Divide by 4
0	1	1	Divide by 8
1	0	0	Divide by 16
1	0	1	Divide by 32
1	1	0	Divide by 64
1	1	1	Divide by 128
0	0	0	Divide by 256

MODE2–MODE0—Operation Mode

These bits select one of the eight modes of operation for the timer as listed in Table 8-5. Refer to **8.3 Operating Modes** for more information on the individual modes.

Table 8-5. MODEx Encoding

MODE2	MODE1	MODE0	OPERATION MODE
0	0	0	Input Capture/Output Compare
0	0	1	Square-Wave Generator
0	1	0	Variable Duty-Cycle Square-Wave Generator
0	1	1	Variable-Width Single-Shot Pulse Generator
1	0	0	Pulse-Width Measurement
1	0	1	Period Measurement
1	1	0	Event Count
1	1	1	Timer Bypass (Simple Test Mode)

OC1–OC0—Output Control

These bits select the conditions under which TOUTx changes (see Table 8-6). These bits may have a different effect when in the input capture/output compare mode. Caution should be used when modifying the OC bits near timer events.

Table 8-6. OCx Encoding

OC1	OC0	TOUTx MODE
0	0	Disabled
0	1	Toggle Mode
1	0	Zero Mode
1	1	One Mode

Freescale Semiconductor, Inc.

For More Information On This Product,
Go to: www.freescale.com

Table 9-2. Boundary Scan Bit Definitions

Bit Num	Cell Type	Pin/Cell Name	Pin Type	Output CTL Cell	Bit Num	Cell Type	Pin/Cell Name	Pin Type	Output CTL Cell
0	IO.Cell	FC3	I/O*	ab.ctl	35	O.Latch	R≈RDYA	Output	—
1	IO.Cell	FC2	I/O*	ab.ctl	36	O.Latch	T≈RDYA	Output	—
2	IO.Cell	FC1	I/O*	ab.ctl	37	I.Pin	RxDB	Input	—
3	IO.Cell	FC0	I/O*	ab.ctl	38	O.Latch	TxDB	Output	—
4	IO.Cell	A23	I/O*	ab.ctl	39	O.Latch	RTSB	Output	—
5	IO.Cell	A22	I/O*	ab.ctl	40	I.Pin	CTSB	Input	—
6	IO.Cell	A21	I/O*	ab.ctl	41	I.Pin	SCLK	Input	—
7	IO.Cell	A20	I/O*	ab.ctl	42	I.Pin	X1	Input	—
8	IO.Cell	A19	I/O*	ab.ctl	43	I.Pin	DREQ1	Input	—
9	IO.Cell	A18	I/O*	ab.ctl	44	O.Latch	DACK1	Output	—
10	IO.Cell	A17	I/O*	ab.ctl	45	O.Latch	DONE1	OD-I/O	—
11	IO.Cell	A16	I/O*	ab.ctl	46	I.Pin	DONE1	OD-I/O	—
12	IO.Cell	A15	I/O*	ab.ctl	47	I.Pin	DREQ2	Input	—
13	IO.Cell	A14	I/O*	ab.ctl	48	O.Latch	DACK2	Output	—
14	IO.Cell	A13	I/O*	ab.ctl	49	O.Latch	DONE2	OD-I/O	—
15	IO.Cell	A12	I/O*	ab.ctl	50	I.Pin	DONE2	OD-I/O	—
16	IO.Cell	A11	I/O*	ab.ctl	51	IO.Cell	IRQ7	I/O	irq7.ctl
17	IO.Cell	A10	I/O*	ab.ctl	52	IO.Ctl0	irq7.ctl	—	—
18	IO.Cell	A9	I/O*	ab.ctl	53	IO.Cell	IRQ6	I/O	irq6.ctl
19	IO.Cell	A8	I/O*	ab.ctl	54	IO.Ctl0	irq6.ctl	—	—
20	IO.Cell	A7	I/O*	ab.ctl	55	IO.Cell	IRQ5	I/O	irq5.ctl
21	IO.Cell	A6	I/O*	ab.ctl	56	IO.Ctl0	irq5.ctl	—	—
22	IO.Cell	A5	I/O*	ab.ctl	57	IO.Cell	CS3	I/O	cs3.ctl
23	IO.Cell	A4	I/O*	ab.ctl	58	IO.Ctl0	cs3.ctl	—	—
24	IO.Cell	A3	I/O*	ab.ctl	59	IO.Cell	IRQ3	I/O	irq3.ctl
25	IO.Cell	A2	I/O*	ab.ctl	60	IO.Ctl0	irq3.ctl	—	—
26	IO.Cell	A1	I/O*	ab.ctl	61	IO.Cell	CS2	I/O	cs2.ctl
27	I.Pin	TGATE2	Input	—	62	IO.Ctl0	cs2.ctl	—	—
28	O.Latch	TOUT2	TS-Output	tout2.ctl	63	IO.Cell	CS1	I/O	cs1.ctl
29	IO.Ctl0	tout2.ctl	—	—	64	IO.Ctl0	cs1.ctl	—	—
30	I.Pin	TIN2	Input	—	65	IO.Cell	CS0	I/O	cs0.ctl
31	I.Pin	RxDA	Input	—	66	IO.Ctl0	cs0.ctl	—	—
32	O.Latch	TxDA	Output	—	67	IO.Cell	D0	I/O	db.ctl
33	O.Latch	RTSA	Output	—	68	IO.Cell	D1	I/O	db.ctl
34	I.Pin	CTSA	Input	—	69	IO.Cell	D2	I/O	db.ctl

Freescale Semiconductor, Inc.

For More Information On This Product,
Go to: www.freescale.com

11.6 AC ELECTRICAL SPECIFICATIONS CONTROL TIMING (See notes (a), (b), (c), and (d) corresponding to part operation, GND = 0 Vdc, TA = 0 to 70°C; see numbered notes)

Num.	Characteristic	Symbol	3.3 V 8.39 MHz Min	3.3 V 8.39 MHz Max	3.3 V or 5.0 V 16.78 MHz Min	3.3 V or 5.0 V 16.78 MHz Max	5.0 V 25.16 MHz Min	5.0 V 25.16 MHz Max	Unit
	System Frequency[1]	f_{sys}	dc	8.39	dc	16.78	dc	25.16	MHz
	Crystal Frequency	f_{XTAL}	25	50	25	50	25	50	kHz
	On-Chip VCO System Frequency	f_{sys}	0.13	8.39	0.13	16.78	0.13	25.16	MHz
	On-Chip VCO Frequency Range	f_{VCO}	0.1	16.78	0.1	33.5	0.1	50.3	MHz
	External Clock Operation	f_{sys}	0	8	0	16	0	25	MHz
	PLL Start-up Time[2]	t_{rc}	—	20	—	20	—	20	ms
	Limp Mode Clock Frequency[3] SYNCR X-bit = 0 SYNCR X-bit = 1	f_{limp}	 — —	 $f_{sys}/2$ f_{sys}	 — —	 $f_{sys}/2$ f_{sys}	 — —	 $f_{sys}/2$ f_{sys}	kHz
	CLKOUT stability[4]	Δ_{CLK}	–1	+1	–1	+1	–1	+1	%
1[5]	CLKOUT Period in Crystal Mode	t_{cyc}	119.2	—	59.6	—	40	—	ns
1B[6]	External Clock Input Period	t_{EXTcyc}	125	—	62.5	—	40	—	ns
1C[7]	External Clock Input Period with PLL	t_{EXTcyc}	125	—	62.5	—	40	—	ns
2,3[8]	CLKOUT Pulse Width in Crystal Mode	t_{CW}	56	—	28	—	19	—	ns
2B, 3B[9]	CLKOUT Pulse Width in External Mode	t_{EXTCW}	56	—	28	—	18	—	ns
2C, 3C[10]	CLKOUT Pulse Width in External w/PLL Mode	t_{EXTCW}	62.5	—	31	—	20	—	ns
4,5	CLKOUT Rise and Fall Times	t_{Crf}	—	10	—	5	—	4	ns

NOTES:

(a) The electrical specifications in this document for both the 8.39 and 16.78 MHz @ 3.3 V ±0.3 V are preliminary and apply only to the appropriate MC68340V low voltage part.
(b) The 16.78-MHz specifications apply to the MC68340 @ 5.0 V ±5% operation.
(c) The 25.16 MHz @ 5.0 V ±5% electrical specifications are preliminary.
(d) For extended temperature parts T_A = –40 to +85°C. These specifications are preliminary.
1. All internal registers retain data at 0 Hz.
2. Assumes that a stable V_{CCSYN} is applied, that an external filter capacitor with a value of 0.1 µF is attached to the XFC pin, and that the crystal oscillator is stable. Lock time is measured from power-up to RESET release. This specification also applies to the period required for PLL lock after changing the W and Y frequency control bits in the synthesizer control register (SYNCR) while the PLL is running, and to the period required for the clock to lock after LPSTOP.
3. Determined by the initial control voltage applied to the on-chip VCO. The X-bit in the SYNCR controls a divide-by-two scaler on the system clock output.
4. CLKOUT stability is the average deviation from programmed frequency measured at maximum f_{sys}. Measurement is made with a stable external clock input applied using the PLL.
5. All crystal mode clock specifications are based on using a 32.768-kHz crystal for the input.
6. When using the external clock input mode (MODCK reset value = 0 V), the minimum allowable t_{EXTcyc} period will be reduced when the duty cycle of the signal applied to EXTAL exceeds 5% tolerance. The relationship between external clock input duty cycle and minimum t_{EXTcyc} is expressed:
   Minimum t_{EXTcyc} period = minimum t_{EXTCW} / (50% – external clock input duty cycle tolerance).
   Minimum external clock low and high times are based on a 45% duty cycle.
7. When using the external clock input mode with the PLL (MODCK reset value = 0 V), the external clock input duty cycle can be at minimum 20% to produce a CLKOUT with a 50% duty cycle.
8. For crystal mode operation, the minimum CLKOUT pulse width is based on a 47% duty cycle.
9. For external clock mode operation, the minimum CLKOUT pulse width is based on a 45% duty cycle, with a 50% duty cycle input clock.

Freescale Semiconductor, Inc.

For More Information On This Product,
Go to: www.freescale.com

11.11 IEEE 1149.1 ELECTRICAL SPECIFICATIONS (See notes (a), (b), (c), and (d) corresponding to part operation, GND = 0 Vdc, TA = 0 to 70°C; see Figures 11-19–11-21)

Num.	Characteristic	3.3 V 8.39 MHz Min	3.3 V 8.39 MHz Max	3.3 V or 5.0 V 16.78 MHz Min	3.3 V or 5.0 V 16.78 MHz Max	5.0 V 25.16 MHz Min	5.0 V 25.16 MHz Max	Unit
	TCK Frequency of Operation	0	8.39	0	16.78	0	25	MHz
1	TCK Cycle Time in Crystal Mode	119.2	—	59.6	—	40	—	ns
2	TCK Clock Pulse Width Measured at 1.5 V	56	—	28	—	18	—	ns
3	TCK Rise and Fall Times	0	10	0	5	0	3	ns
6	Boundary Scan Input Data Setup Time	32	—	16	—	10	—	ns
7	Boundary Scan Input Data Hold Time	52	—	26	—	18	—	ns
8	TCK Low to Output Data Valid	0	80	0	40	0	26	ns
9	TCK Low to Output High Impedance	0	120	0	60	0	40	ns
10	TMS, TDI Data Setup Time	30	—	15	—	10	—	ns
11	TMS, TDI Data Hold Time	30	—	15	—	10	—	ns
12	TCK Low to TDO Data Valid	0	50	0	25	0	16	ns
13	TCK Low to TDO High Impedance	0	50	0	25	0	16	ns

NOTES:
(a) The electrical specifications in this document for both the 8.39 and 16.78 MHz @ 3.3 V ±0.3 V are preliminary, and apply only to the appropriate MC68340V low voltage part.
(b) The 16.78-MHz specifications apply to the MC68340 @ 5.0 V ±5% operation.
(c) The 25.16 MHz @ 5.0 V ±5% electrical specifications are preliminary.
(d) For extended temperature parts T_A = –40 to +85°C. These specifications are preliminary.

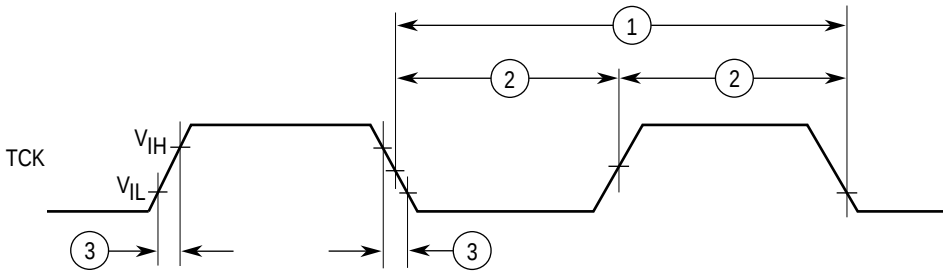
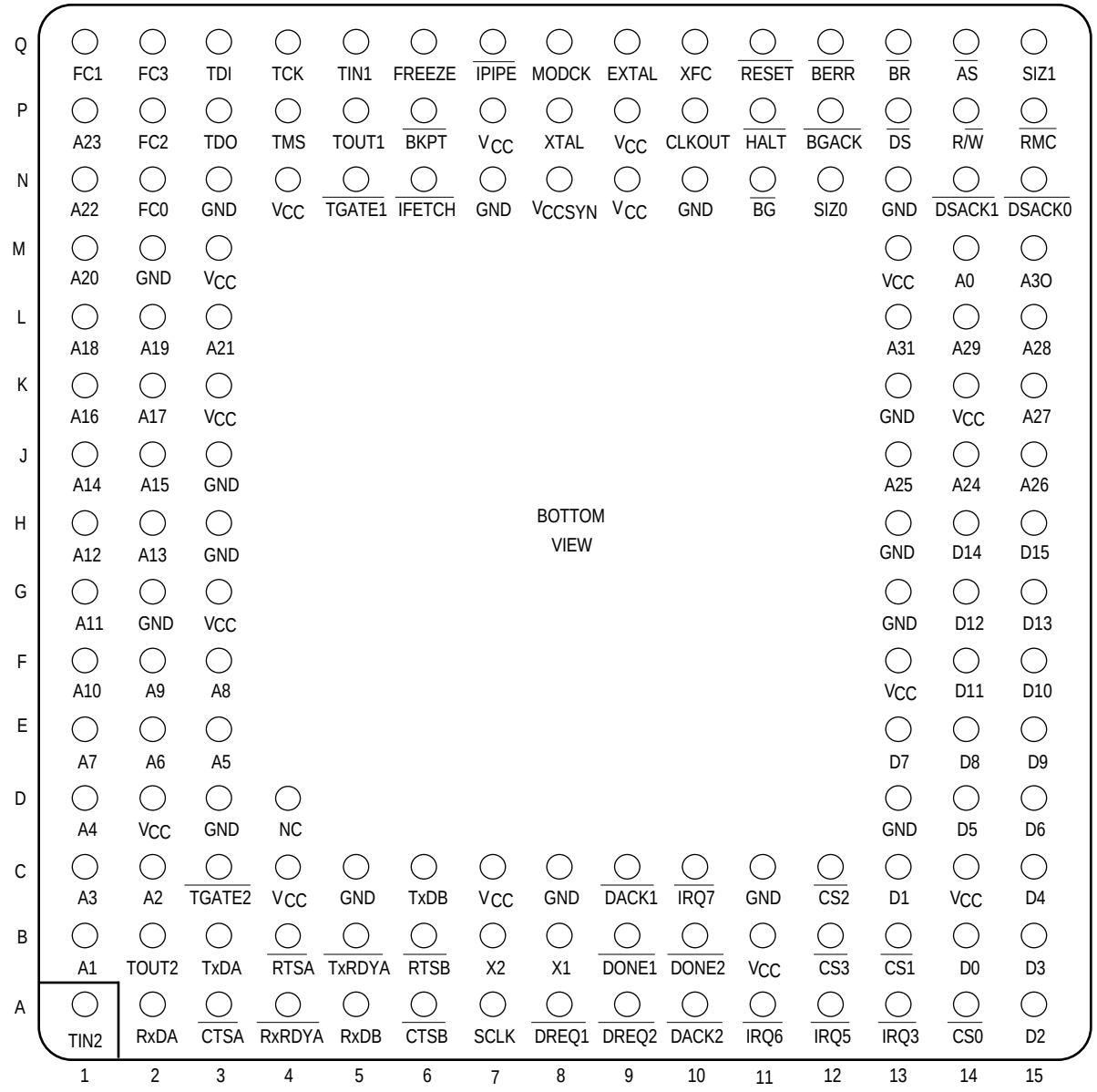


Figure 11-19. Test Clock Input Timing Diagram

Freescale Semiconductor, Inc.

For More Information On This Product,
Go to: www.freescale.com

12.2.2 145-Lead Plastic Pin Grid Array (RP Suffix)



Freescale Semiconductor, Inc.

For More Information On This Product,
Go to: www.freescale.com

— R —

For More Information On This Product,
Go to: www.freescale.com

Freescale Semiconductor, Inc.