**Welcome to E-XFL.COM**

## What is "Embedded - Microcontrollers"?

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

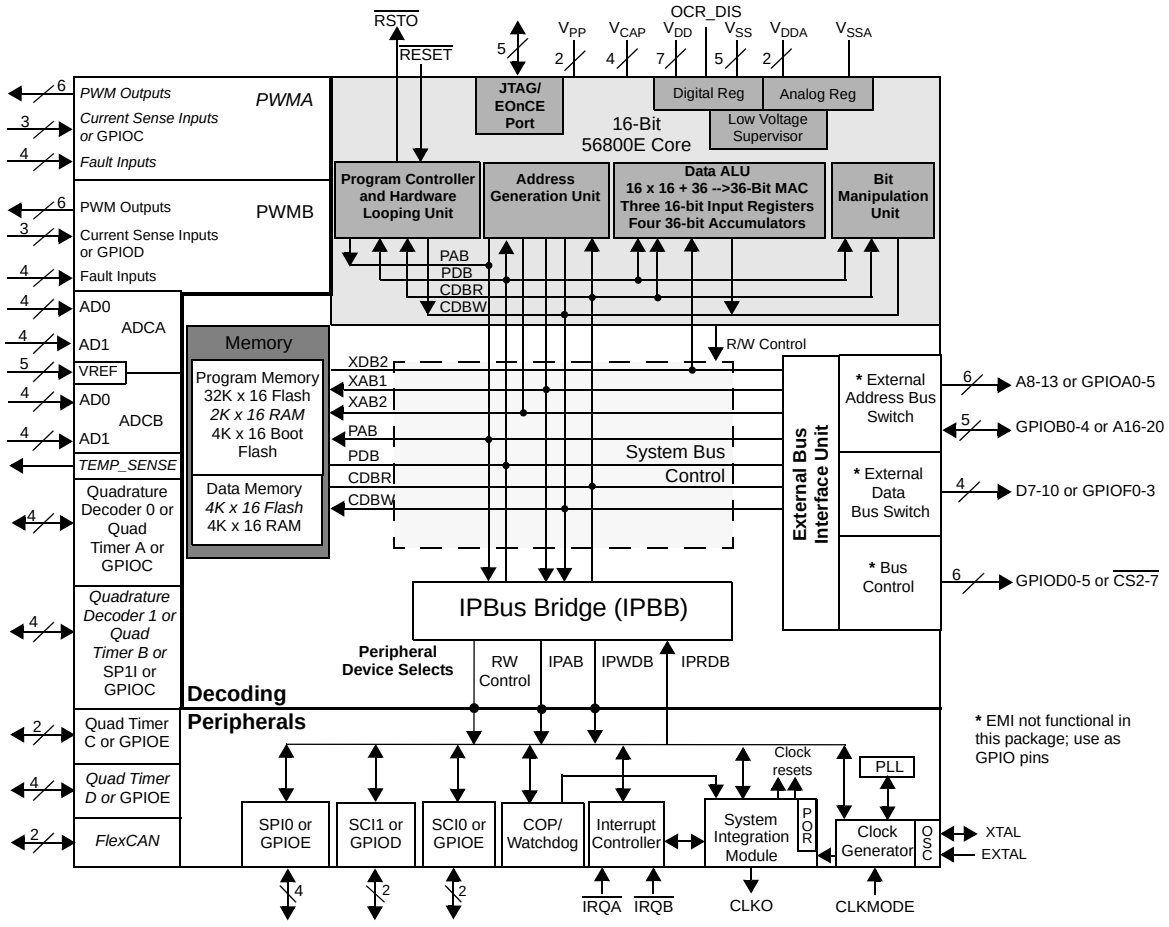
## Applications of "Embedded - Microcontrollers"

| Details | |
|---|---|
| Product Status | Active |
| Core Processor | 56800E |
| Core Size | 16-Bit |
| Speed | 60MHz |
| Connectivity | CANbus, EBI/EMI, SCI, SPI |
| Peripherals | POR, PWM, Temp Sensor, WDT |
| Number of I/O | 49 |
| Program Memory Size | 64KB (32K x 16) |
| Program Memory Type | FLASH |
| EEPROM Size | - |
| RAM Size | 6K x 16 |
| Voltage - Supply (Vcc/Vdd) | 2.25V ~ 3.6V |
| Data Converters | A/D 16x12b |
| Oscillator Type | External |
| Operating Temperature | -40°C ~ 105°C (TA) |
| Mounting Type | Surface Mount |
| Package / Case | 128-LQFP |
| Supplier Device Package | 128-LQFP (14x20) |
| Purchase URL | https://www.e-xfl.com/product-detail/nxp-semiconductors/mc56f8335vfge |

# 56F8335/56F8135 General Description

**Note:** *Features in italics are NOT available in the 56F8135 device.*

- Up to 60 MIPS at 60MHz core frequency
- DSP and MCU functionality in a unified, C-efficient architecture
- 64KB Program Flash
- *4KB Program RAM*
- *8KB Data Flash*
- 8KB Data RAM
- 8KB Boot Flash
- Up to two 6-channel PWM modules
- Four 4-channel, 12-bit ADCs
- *Temperature Sensor*
- *Up to two* Quadrature Decoders
- *FlexCAN module*
- Optional On-Chip Regulator
- Two Serial Communication Interfaces (SCIs)
- Up to two Serial Peripheral Interface (SPIs)
- Up to four general-purpose Quad Timers
- Computer Operating Properly (COP)/Watchdog
- JTAG/Enhanced On-Chip Emulation (OnCE™) for unobtrusive, real-time debugging
- Up to 49 GPIO lines
- 128-pin LQFP Package




**56F8335/56F8135 Block Diagram - 128 LQFP**

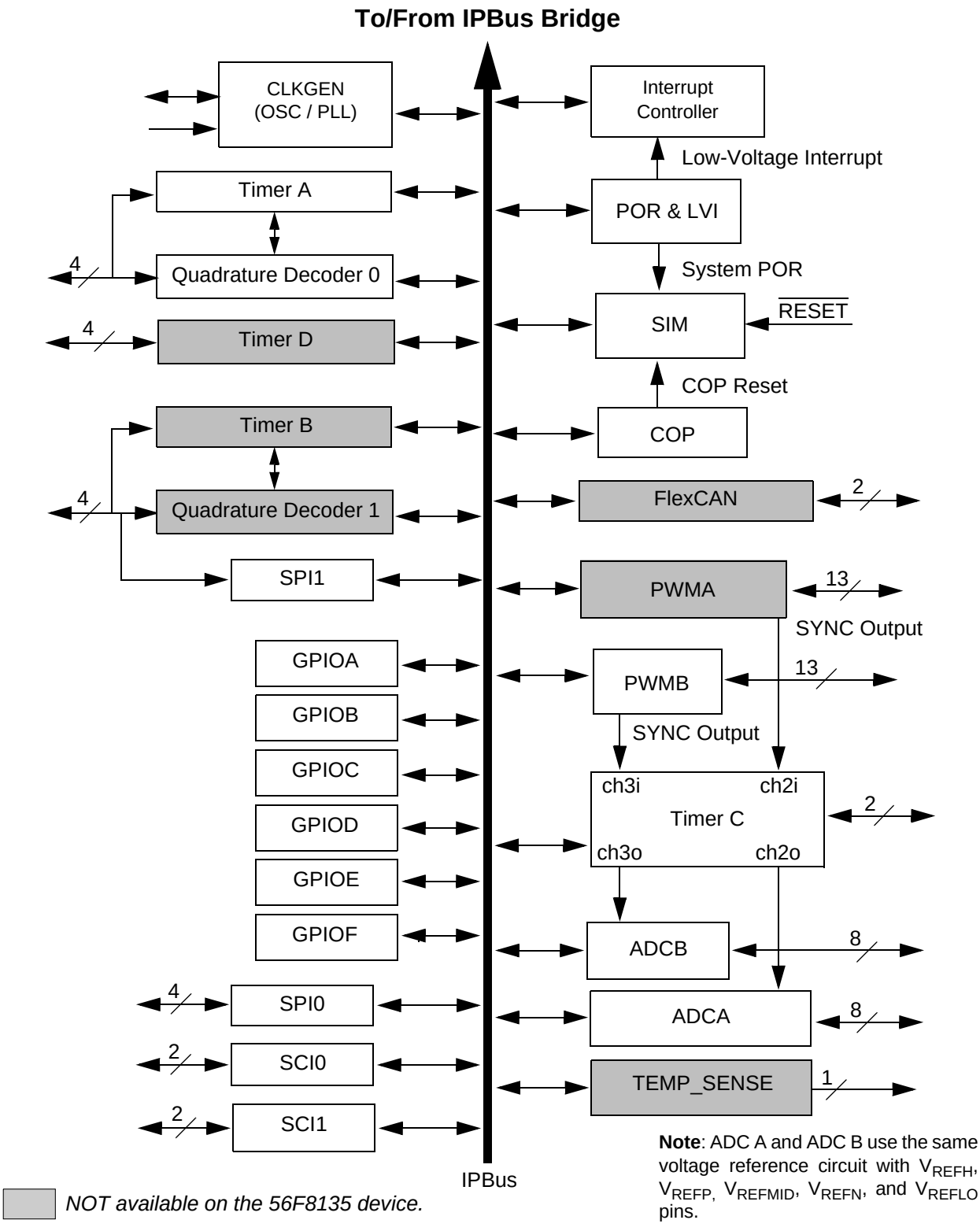



**Figure 1-2 Peripheral Subsystem**

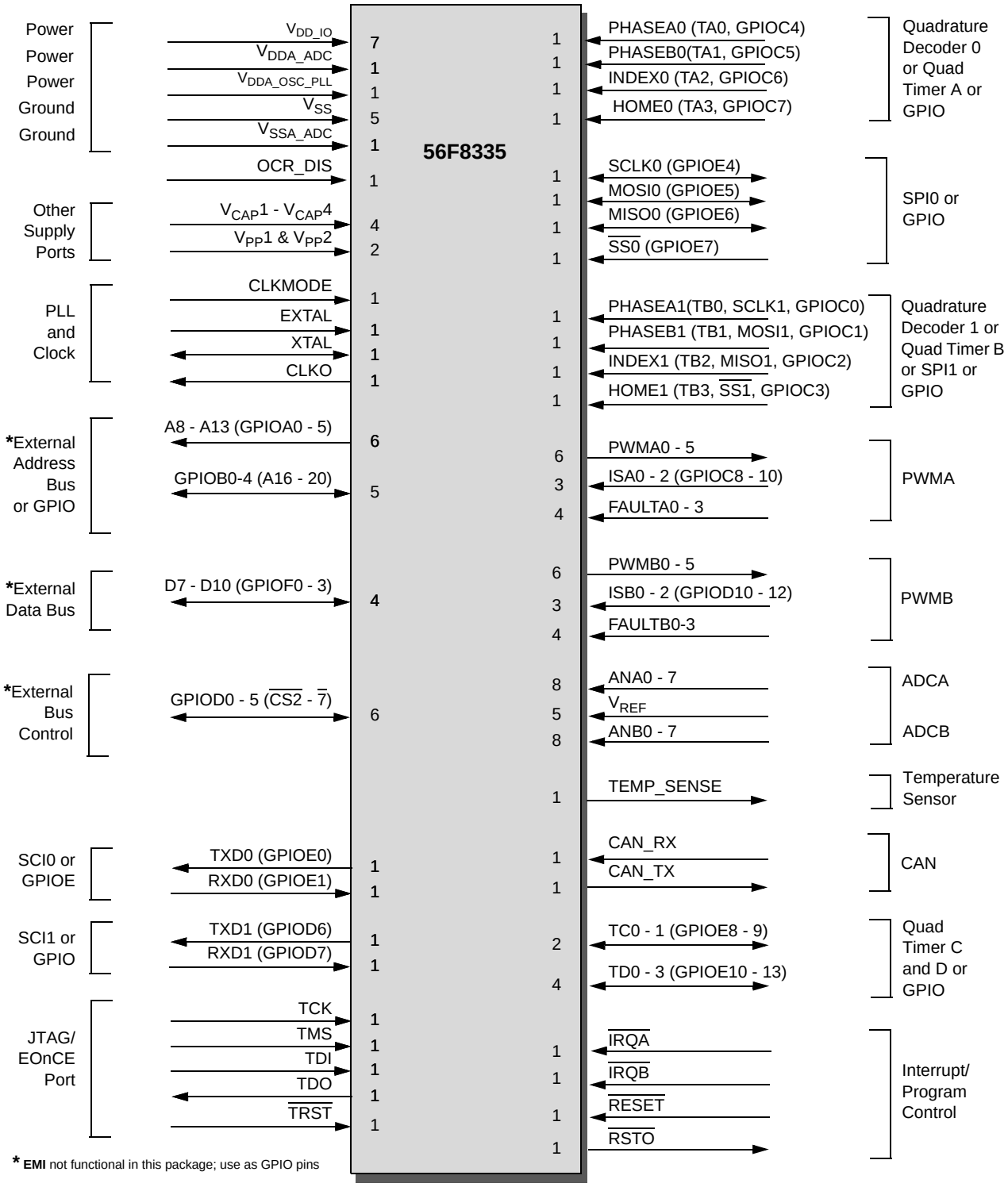



**Figure 2-1 56F8335 Signals Identified by Functional Group[1] (128-Pin LQFP)**

1. Alternate pin functionality is shown in parenthesis; pin direction/type shown is the default functionality.

## Table 2-2 Signal and Package Information for the 128-Pin LQFP (Continued)

| Signal Name | Pin No. | Type | State During Reset | Signal Description |
|---|---|---|---|---|
| **$\overline{\text{TRST}}$** | 114 | Schmitt Input | Input, pulled high internally | **Test Reset** — As an input, a low signal on this pin provides a reset signal to the JTAG TAP controller. To ensure complete hardware reset, $\overline{\text{TRST}}$ should be asserted whenever $\overline{\text{RESET}}$ is asserted. The only exception occurs in a debugging environment when a hardware device reset is required and the JTAG/EOnCE module must not be reset. In this case, assert $\overline{\text{RESET}}$, but do not assert $\overline{\text{TRST}}$.<br><br>To deactivate the internal pull-up resistor, set the JTAG bit in the SIM_PUDR register.<br><br>**Note:** For normal operation, connect $\overline{\text{TRST}}$ directly to $V_{SS}$. If the design is to be used in a debugging environment, $\overline{\text{TRST}}$ may be tied to $V_{SS}$ through a 1K resistor. |
| **PHASEA0**<br><br>**(TA0)**<br><br>**(GPIOC4)** | 127 | Schmitt Input<br><br>Schmitt Input/ Output<br><br>Schmitt Input/ Output | Input, pull-up enabled | **Phase A** — Quadrature Decoder 0, PHASEA input<br><br>**TA0** — Timer A, Channel 0<br><br>**Port C GPIO** — This GPIO pin can be individually programmed as an input or output pin.<br><br>After reset, the default state is PHASEA0.<br><br>To deactivate the internal pull-up resistor, clear bit 4 of the GPIOC_PUR register. |
| **PHASEB0**<br><br>**(TA1)**<br><br>**(GPIOC5)** | 128 | Schmitt Input<br><br>Schmitt Input/ Output<br><br>Schmitt Input/ Output | Input, pull-up enabled | **Phase B** — Quadrature Decoder 0, PHASEB input<br><br>**TA1** — Timer A, Channel 1<br><br>**Port C GPIO** — This GPIO pin can be individually programmed as an input or output pin.<br><br>After reset, the default state is PHASEB0.<br><br>To deactivate the internal pull-up resistor, clear bit 5 of the GPIOC_PUR register. |

## Table 2-2 Signal and Package Information for the 128-Pin LQFP (Continued)

| Signal Name | Pin No. | Type | State During Reset | Signal Description |
|---|---|---|---|---|
| **MOSI0**<br><br>**(GPIOE5)** | 126 | Input/ Output<br><br>Input/ Output | In reset, output is disabled, pull-up is enabled | **SPI 0 Master Out/Slave In** — This serial data pin is an output from a master device and an input to a slave device. The master device places data on the MOSI line a half-cycle before the clock edge the slave device uses to latch the data.<br><br>**Port E GPIO** — This GPIO pin can be individually programmed as an input or output pin.<br><br>After reset, the default state is MOSI0.<br><br>To deactivate the internal pull-up resistor, clear bit 5 in the GPIOE_PUR register. |
| **MISO0**<br><br>**(GPIOE6)** | 125 | Input/ Output<br><br>Input/ Output | Input, pull-up enabled | **SPI 0 Master In/Slave Out** — This serial data pin is an input to a master device and an output from a slave device. The MISO line of a slave device is placed in the high-impedance state if the slave device is not selected. The slave device places data on the MISO line a half-cycle before the clock edge the master device uses to latch the data.<br><br>**Port E GPIO** — This GPIO pin can be individually programmed as an input or output pin.<br><br>After reset, the default state is MISO0.<br><br>To deactivate the internal pull-up resistor, clear bit 6 in the GPIOE_PUR register. |
| **$\overline{\text{SS0}}$**<br><br>**(GPIOE7)** | 123 | Input<br><br>Input/ Output | Input, pull-up enabled | **SPI 0 Slave Select** — $\overline{\text{SS0}}$ is used in slave mode to indicate to the SPI module that the current transfer is to be received.<br><br>**Port E GPIO** — This GPIO pin can be individually programmed as an input or output pin.<br><br>After reset, the default state is $\overline{\text{SS0}}$.<br><br>To deactivate the internal pull-up resistor, clear bit 7 in the GPIOE_PUR register. |

# Part 3 On-Chip Clock Synthesis (OCCS)

## 3.1 Introduction

Refer to the OCCS chapter of the **56F8300 Peripheral User Manual** for a full description of the OCCS. The material contained here identifies the specific features of the OCCS design. **Figure 3-1** shows the specific OCCS block diagram to reference in the OCCS chapter of the **56F8300 Peripheral User Manual**.

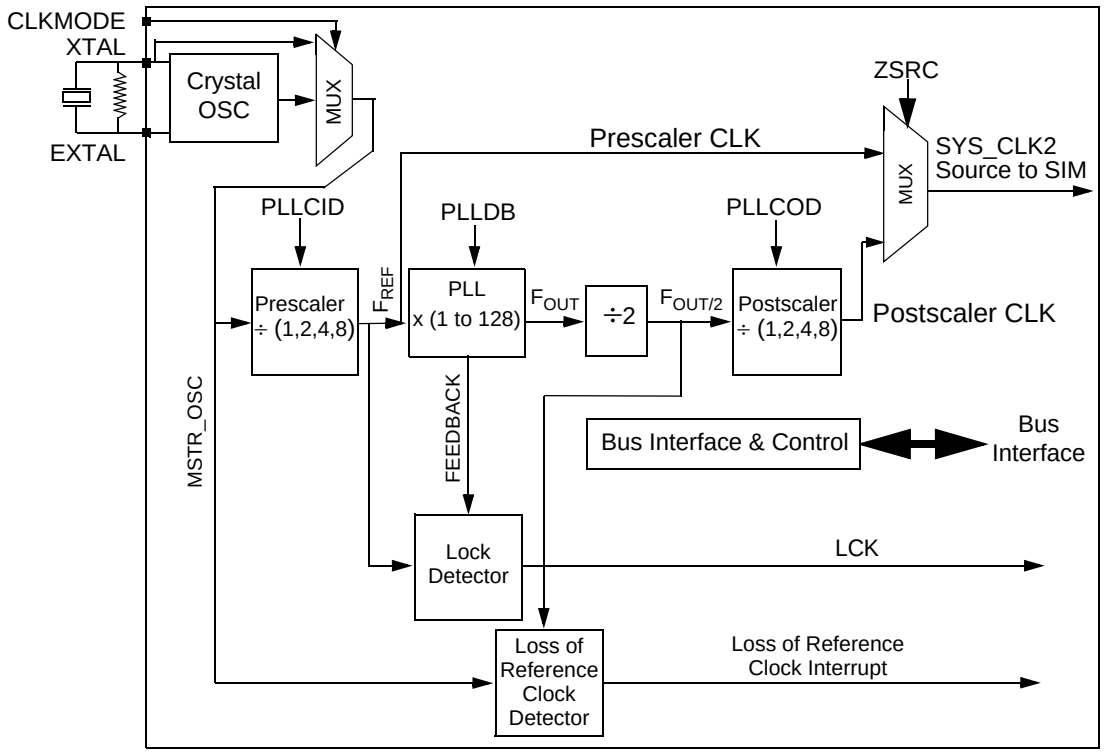



**Figure 3-1 OCCS Block Diagram**

## 3.2 External Clock Operation

The system clock can be derived from an external crystal, ceramic resonator, or an external system clock signal. To generate a reference frequency using the internal oscillator, a reference crystal or ceramic resonator must be connected between the EXTAL and XTAL pins.

### 3.2.1 Crystal Oscillator

The internal oscillator is designed to interface with a parallel-resonant crystal resonator in the frequency range specified for the external crystal in **Table 10-15**. A recommended crystal oscillator circuit is shown in **Figure 3-2**. Follow the crystal supplier's recommendations when selecting a crystal, since crystal parameters determine the component values required to provide maximum stability and reliable start-up. The crystal and associated components should be mounted as near as possible to the EXTAL and XTAL

pins to minimize output distortion and start-up stabilization time.

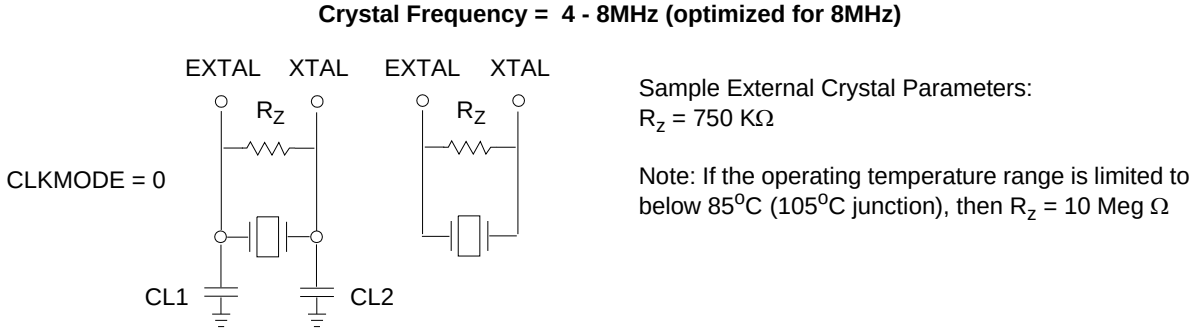



**Figure 3-2 Connecting to a Crystal Oscillator**

**Note:** The OCCS_COHL bit must be set to 1 when a crystal oscillator is used. The reset condition on the OCCS_COHL bit is 0. Please see the COHL bit in the Oscillator Control (OSCTL) register, discussed in the **56F8300 Peripheral User's Manual**.

## 3.2.2 Ceramic Resonator (Default)

It is also possible to drive the internal oscillator with a ceramic resonator, assuming the overall system design can tolerate the reduced signal integrity. A typical ceramic resonator circuit is shown in **Figure 3-3**. Refer to the supplier's recommendations when selecting a ceramic resonator and associated components. The resonator and components should be mounted as near as possible to the EXTAL and XTAL pins.

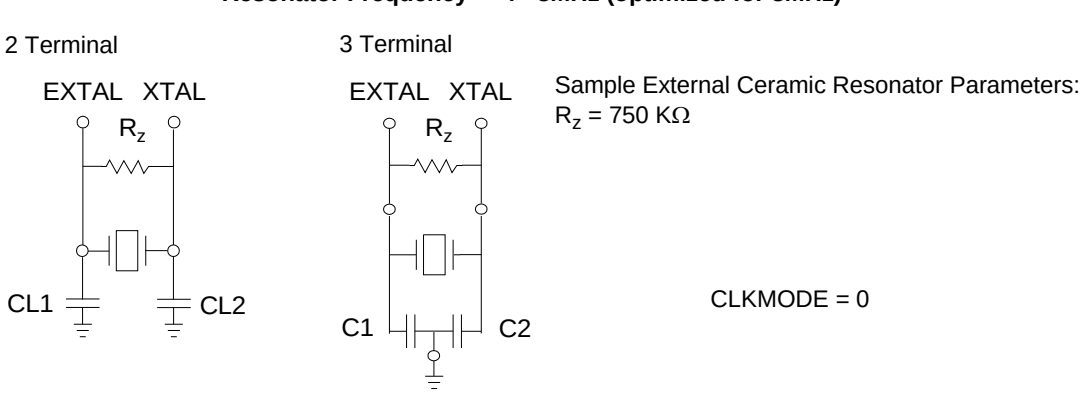



**Figure 3-3 Connecting a Ceramic Resonator**

**Note:** The OCCS_COHL bit must be set to 0 when a ceramic resonator is used. The reset condition on the OCCS_COHL bit is 0. Please see the COHL bit in the Oscillator Control (OSCTL) register, discussed in the **56F8300 Peripheral User's Manual**.

## 3.2.3 External Clock Source

The recommended method of connecting an external clock is illustrated in **Figure 3-4**. The external clock source is connected to XTAL and the EXTAL pin is grounded. Set OCCS_COHL bit high when using an external clock source as well.

## 4.5 Flash Memory Map

**Figure 4-1** illustrates the Flash Memory (FM) map on the system bus.

The Flash Memory is divided into three functional blocks. The Program and boot memories reside on the Program Memory buses. They are controlled by one set of banked registers. Data Memory Flash resides on the Data Memory buses and is controlled separately by its own set of banked registers.

The top nine words of the Program Memory Flash are treated as special memory locations. The content of these words is used to control the operation of the Flash Controller. Because these words are part of the Flash Memory content, their state is maintained during power-down and reset. During chip initialization, the content of these memory locations is loaded into Flash Memory control registers, detailed in the Flash Memory chapter of the **56F8300 Peripheral User Manual**. These configuration parameters are located between $00_FFF7 and $00_FFFF.

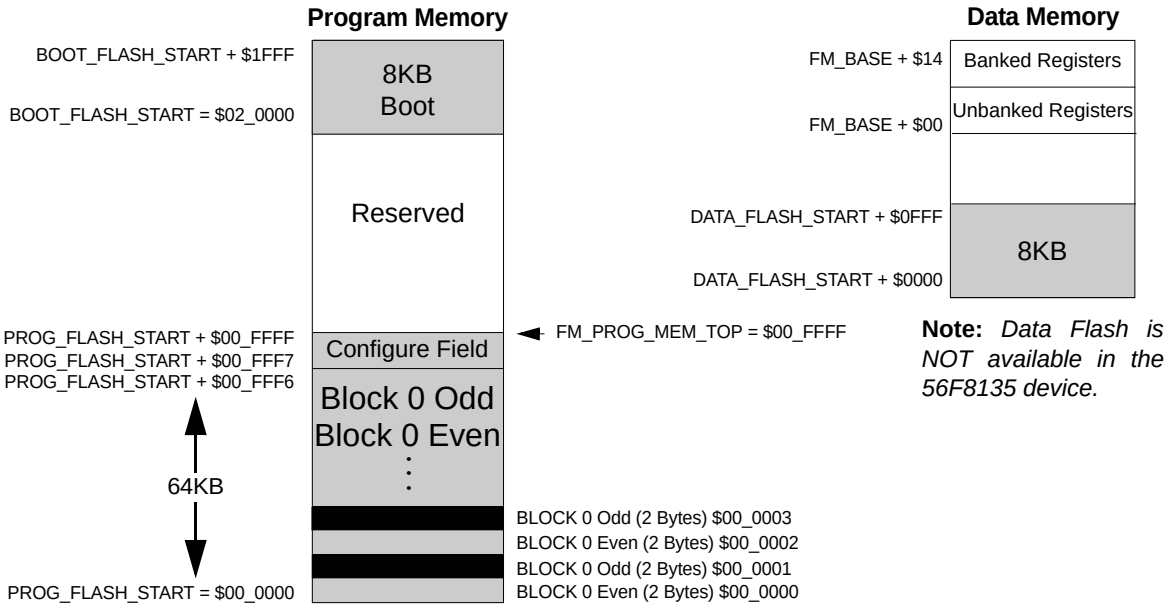



**Figure 4-1 Flash Array Memory Maps**

**Table 4-7** shows the page and sector sizes used within each Flash memory block on the chip.

**Note:** *Data Flash is NOT available on the 56F8135 device.*

**Table 4-7. Flash Memory Partitions**

| | Flash Size | Sectors | Sector Size | Page Size |
|---|---|---|---|---|
| Program Flash | 64KB | 16 | 2K x 16 bits | 512 x 16 bits |

## Table 4-20 Analog-to-Digital Converter Registers Address Map (Continued) (ADCA_BASE = $00 F200)

| Register Acronym | Address Offset | Register Description |
|---|---|---|
| ADCA_POWER | $29 | Power Control Register |
| ADCA_CAL | $2A | ADC Calibration Register |

## Table 4-21 Analog-to-Digital Converter Registers Address Map (ADCB_BASE = $00 F240)

| Register Acronym | Address Offset | Register Description |
|---|---|---|
| ADCB_CR1 | $0 | Control Register 1 |
| ADCB_CR2 | $1 | Control Register 2 |
| ADCB_ZCC | $2 | Zero Crossing Control Register |
| ADCB_LST 1 | $3 | Channel List Register 1 |
| ADCB_LST 2 | $4 | Channel List Register 2 |
| ADCB_SDIS | $5 | Sample Disable Register |
| ADCB_STAT | $6 | Status Register |
| ADCB_LSTAT | $7 | Limit Status Register |
| ADCB_ZCSTAT | $8 | Zero Crossing Status Register |
| ADCB_RSLT 0 | $9 | Result Register 0 |
| ADCB_RSLT 1 | $A | Result Register 1 |
| ADCB_RSLT 2 | $B | Result Register 2 |
| ADCB_RSLT 3 | $C | Result Register 3 |
| ADCB_RSLT 4 | $D | Result Register 4 |
| ADCB_RSLT 5 | $E | Result Register 5 |
| ADCB_RSLT 6 | $F | Result Register 6 |
| ADCB_RSLT 7 | $10 | Result Register 7 |
| ADCB_LLMT 0 | $11 | Low Limit Register 0 |
| ADCB_LLMT 1 | $12 | Low Limit Register 1 |
| ADCB_LLMT 2 | $13 | Low Limit Register 2 |
| ADCB_LLMT 3 | $14 | Low Limit Register 3 |
| ADCB_LLMT 4 | $15 | Low Limit Register 4 |
| ADCB_LLMT 5 | $16 | Low Limit Register 5 |
| ADCB_LLMT 6 | $17 | Low Limit Register 6 |
| ADCB_LLMT 7 | $18 | Low Limit Register 7 |
| ADCB_HLMT 0 | $19 | High Limit Register 0 |
| ADCB_HLMT 1 | $1A | High Limit Register 1 |

## Table 4-38 FlexCAN Registers Address Map (FC_BASE = $00 F800)
### *FlexCAN is NOT available in the 56F8135 device*

| Register Acronym | Address Offset | Register Description |
|---|---|---|
| FCMCR | $0 | Module Configuration Register |
| | | Reserved |
| FCCTL0 | $3 | Control Register 0 Register |
| FCCTL1 | $4 | Control Register 1 Register |
| FCTMR | $5 | Free-Running Timer Register |
| FCMAXMB | $6 | Maximum Message Buffer Configuration Register |
| | | Reserved |
| FCRXGMASK_H | $8 | Receive Global Mask High Register |
| FCRXGMASK_L | $9 | Receive Global Mask Low Register |
| FCRX14MASK_H | $A | Receive Buffer 14 Mask High Register |
| FCRX14MASK_L | $B | Receive Buffer 14 Mask Low Register |
| FCRX15MASK_H | $C | Receive Buffer 15 Mask High Register |
| FCRX15MASK_L | $D | Receive Buffer 15 Mask Low Register |
| | | Reserved |
| FCSTATUS | $10 | Error and Status Register |
| FCIMASK1 | $11 | Interrupt Masks 1 Register |
| FCIFLAG1 | $12 | Interrupt Flags 1 Register |
| FCR/T_ERROR_CNTRS | $13 | Receive and Transmit Error Counters Register |
| | | Reserved |
| | | Reserved |
| | | Reserved |
| FCMB0_CONTROL | $40 | Message Buffer 0 Control / Status Register |
| FCMB0_ID_HIGH | $41 | Message Buffer 0 ID High Register |
| FCMB0_ID_LOW | $42 | Message Buffer 0 ID Low Register |
| FCMB0_DATA | $43 | Message Buffer 0 Data Register |
| FCMB0_DATA | $44 | Message Buffer 0 Data Register |
| FCMB0_DATA | $45 | Message Buffer 0 Data Register |
| FCMB0_DATA | $46 | Message Buffer 0 Data Register |
| | | Reserved |
| FCMSB1_CONTROL | $48 | Message Buffer 1 Control / Status Register |
| FCMSB1_ID_HIGH | $49 | Message Buffer 1 ID High Register |

## Table 4-38 FlexCAN Registers Address Map (Continued) (FC_BASE = $00 F800)
### *FlexCAN is NOT available in the 56F8135 device*

| Register Acronym | Address Offset | Register Description |
|---|---|---|
| FCMB5_DATA | $6B | Message Buffer 5 Data Register |
| FCMB5_DATA | $6C | Message Buffer 5 Data Register |
| FCMB5_DATA | $6D | Message Buffer 5 Data Register |
| FCMB5_DATA | $6E | Message Buffer 5 Data Register |
| | | Reserved |
| FCMB6_CONTROL | $70 | Message Buffer 6 Control / Status Register |
| FCMB6_ID_HIGH | $71 | Message Buffer 6 ID High Register |
| FCMB6_ID_LOW | $72 | Message Buffer 6 ID Low Register |
| FCMB6_DATA | $73 | Message Buffer 6 Data Register |
| FCMB6_DATA | $74 | Message Buffer 6 Data Register |
| FCMB6_DATA | $75 | Message Buffer 6 Data Register |
| FCMB6_DATA | $76 | Message Buffer 6 Data Register |
| | | Reserved |
| FCMB7_CONTROL | $78 | Message Buffer 7 Control / Status Register |
| FCMB7_ID_HIGH | $79 | Message Buffer 7 ID High Register |
| FCMB7_ID_LOW | $7A | Message Buffer 7 ID Low Register |
| FCMB7_DATA | $7B | Message Buffer 7 Data Register |
| FCMB7_DATA | $7C | Message Buffer 7 Data Register |
| FCMB7_DATA | $7D | Message Buffer 7 Data Register |
| FCMB7_DATA | $7E | Message Buffer 7 Data Register |
| | | Reserved |
| FCMB8_CONTROL | $80 | Message Buffer 8 Control / Status Register |
| FCMB8_ID_HIGH | $81 | Message Buffer 8 ID High Register |
| FCMB8_ID_LOW | $82 | Message Buffer 8 ID Low Register |
| FCMB8_DATA | $83 | Message Buffer 8 Data Register |
| FCMB8_DATA | $84 | Message Buffer 8 Data Register |
| FCMB8_DATA | $85 | Message Buffer 8 Data Register |
| FCMB8_DATA | $86 | Message Buffer 8 Data Register |
| | | Reserved |
| FCMB9_CONTROL | $88 | Message Buffer 9 Control / Status Register |
| FCMB9_ID_HIGH | $89 | Message Buffer 9 ID High Register |

**Table 4-38 FlexCAN Registers Address Map (Continued)**
**(FC_BASE = $00 F800)**
***FlexCAN is NOT available in the 56F8135 device***

| Register Acronym | Address Offset | Register Description |
|---|---|---|
| FCMB9_ID_LOW | $8A | Message Buffer 9 ID Low Register |
| FCMB9_DATA | $8B | Message Buffer 9 Data Register |
| FCMB9_DATA | $8C | Message Buffer 9 Data Register |
| FCMB9_DATA | $8D | Message Buffer 9 Data Register |
| FCMB9_DATA | $8E | Message Buffer 9 Data Register |
| | | Reserved |
| FCMB10_CONTROL | $90 | Message Buffer 10 Control / Status Register |
| FCMB10_ID_HIGH | $91 | Message Buffer 10 ID High Register |
| FCMB10_ID_LOW | $92 | Message Buffer 10 ID Low Register |
| FCMB10_DATA | $93 | Message Buffer 10 Data Register |
| FCMB10_DATA | $94 | Message Buffer 10 Data Register |
| FCMB10_DATA | $95 | Message Buffer 10 Data Register |
| FCMB10_DATA | $96 | Message Buffer 10 Data Register |
| | | Reserved |
| FCMB11_CONTROL | $98 | Message Buffer 11 Control / Status Register |
| FCMB11_ID_HIGH | $99 | Message Buffer 11 ID High Register |
| FCMB11_ID_LOW | $9A | Message Buffer 11 ID Low Register |
| FCMB11_DATA | $9B | Message Buffer 11 Data Register |
| FCMB11_DATA | $9C | Message Buffer 11 Data Register |
| FCMB11_DATA | $9D | Message Buffer 11 Data Register |
| FCMB11_DATA | $9E | Message Buffer 11 Data Register |
| | | Reserved |
| FCMB12_CONTROL | $A0 | Message Buffer 12 Control / Status Register |
| FCMB12_ID_HIGH | $A1 | Message Buffer 12 ID High Register |
| FCMB12_ID_LOW | $A2 | Message Buffer 12 ID Low Register |
| FCMB12_DATA | $A3 | Message Buffer 12 Data Register |
| FCMB12_DATA | $A4 | Message Buffer 12 Data Register |
| FCMB12_DATA | $A5 | Message Buffer 12 Data Register |
| FCMB12_DATA | $A6 | Message Buffer 12 Data Register |
| | | Reserved |
| FCMB13_CONTROL | $A8 | Message Buffer 13 Control / Status Register |
| FCMB13_ID_HIGH | $A9 | Message Buffer 13 ID High Register |
| FCMB13_ID_LOW | $AA | Message Buffer 13 ID Low Register |

## 5.6 Register Descriptions

A register address is the sum of a base address and an address offset. The base address is defined at the system level and the address offset is defined at the module level. The ITCN peripheral has 24 registers.

**Table 5-3 ITCN Register Summary (ITCN_BASE = $00F1A0)**

| Register Acronym | Base Address + | Register Name | Section Location |
|---|---|---|---|
| IPR0 | $0 | Interrupt Priority Register 0 | 5.6.1 |
| IPR1 | $1 | Interrupt Priority Register 1 | 5.6.2 |
| IPR2 | $2 | Interrupt Priority Register 2 | 5.6.3 |
| IPR3 | $3 | Interrupt Priority Register 3 | 5.6.4 |
| IPR4 | $4 | Interrupt Priority Register 4 | 5.6.5 |
| IPR5 | $5 | Interrupt Priority Register 5 | 5.6.6 |
| IPR6 | $6 | Interrupt Priority Register 6 | 5.6.7 |
| IPR7 | $7 | Interrupt Priority Register 7 | 5.6.8 |
| IPR8 | $8 | Interrupt Priority Register 8 | 5.6.9 |
| IPR9 | $9 | Interrupt Priority Register 9 | 5.6.10 |
| VBA | $A | Vector Base Address Register | 5.6.11 |
| FIM0 | $B | Fast Interrupt 0 Match Register | 5.6.12 |
| FIVAL0 | $C | Fast Interrupt 0 Vector Address Low Register | 5.6.13 |
| FIVAH0 | $D | Fast Interrupt 0 Vector Address High Register | 5.6.14 |
| FIM1 | $E | Fast Interrupt 1 Match Register | 5.6.15 |
| FIVAL1 | $F | Fast Interrupt 1 Vector Address Low Register | 5.6.16 |
| FIVAH1 | $10 | Fast Interrupt 1 Vector Address High Register | 5.6.17 |
| IRQP0 | $11 | IRQ Pending Register 0 | 5.6.18 |
| IRQP1 | $12 | IRQ Pending Register 1 | 5.6.19 |
| IRQP2 | $13 | IRQ Pending Register 2 | 5.6.20 |
| IRQP3 | $14 | IRQ Pending Register 3 | 5.6.21 |
| IRQP4 | $15 | IRQ Pending Register 4 | 5.6.22 |
| IRQP5 | $16 | IRQ Pending Register 5 | 5.6.23 |
| | | Reserved | |
| ICTL | $1D | Interrupt Control Register | 5.6.30 |

| Add. Offset | Register Name | | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $0 | IPR0 | R | 0 | 0 | BKPT_U0 IPL | | STPCNT IPL | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | | W | | | | | | | | | | | | | | | | |
| $1 | IPR1 | R | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | RX_REG IPL | | TX_REG IPL | | TRBUF IPL | |
| | | W | | | | | | | | | | | | | | | | |
| $2 | IPR2 | R | FMCBE IPL | | FMCC IPL | | FMERR IPL | | LOCK IPL | | LVI IPL | | 0 | 0 | IRQB IPL | | IRQA IPL | |
| | | W | | | | | | | | | | | | | | | | |
| $3 | IPR3 | R | GPIOD IPL | | GPIOE IPL | | GPIOF IPL | | FCMSGBUF IPL | | FCWKUP IPL | | FCERR IPL | | FCBOFF IPL | | 0 | 0 |
| | | W | | | | | | | | | | | | | | | | |
| $4 | IPR4 | R | SPI0_RCV IPL | | SPI1_XMIT IPL | | SPI1_RCV IPL | | 0 | 0 | 0 | 0 | GPIOA IPL | | GPIOB IPL | | GPIOC IPL | |
| | | W | | | | | | | | | | | | | | | | |
| $5 | IPR5 | R | DEC1_XIRQ IPL | | DEC1_HIRQ IPL | | SCI1_RCV IPL | | SCI1_RERR IPL | | 0 | 0 | SCI1_TIDL IPL | | SCI1_XMIT IPL | | SPI0_XMIT IPL | |
| | | W | | | | | | | | | | | | | | | | |
| $6 | IPR6 | R | TMRC0 IPL | | TMRD3 IPL | | TMRD2 IPL | | TMRD1 IPL | | TMRD0 IPL | | 0 | 0 | DEC0_XIRQ IPL | | DEC0_HIRQ IPL | |
| | | W | | | | | | | | | | | | | | | | |
| $7 | IPR7 | R | TMRA0 IPL | | TMRB3 IPL | | TMRB2 IPL | | TMRB1 IPL | | TMRB0 IPL | | TMRC3 IPL | | TMRC2 IPL | | TMRC1 IPL | |
| | | W | | | | | | | | | | | | | | | | |
| $8 | IPR8 | R | SCI0_RCV IPL | | SCI0_RERR IPL | | 0 | 0 | SCI0_TIDL IPL | | SCI0_XMIT IPL | | TMRA3 IPL | | TMRA2 IPL | | TMRA1 IPL | |
| | | W | | | | | | | | | | | | | | | | |
| $9 | IPR9 | R | PWMA_F IPL | | PWMB_F IPL | | PWMA_RL IPL | | PWMB_RL IPL | | ADCA_ZC IPL | | ABCB_ZC IPL | | ADCA_CC IPL | | ADCB_CC IPL | |
| | | W | | | | | | | | | | | | | | | | |
| $A | VBA | R | 0 | 0 | 0 | VECTOR BASE ADDRESS | | | | | | | | | | | | |
| | | W | | | | | | | | | | | | | | | | |
| $B | FIM0 | R | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | FAST INTERRUPT 0 | | | | | | |
| | | W | | | | | | | | | | | | | | | | |
| $C | FIVAL0 | R | FAST INTERRUPT 0 VECTOR ADDRESS LOW | | | | | | | | | | | | | | | |
| | | W | | | | | | | | | | | | | | | | |
| $D | FIVAH0 | R | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | FAST INTERRUPT 0 VECTOR ADDRESS HIGH | | | | |
| | | W | | | | | | | | | | | | | | | | |
| $E | FIM1 | R | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | FAST INTERRUPT 1 | | | | | | |
| | | W | | | | | | | | | | | | | | | | |
| $F | FIVAL1 | R | FAST INTERRUPT 1 VECTOR ADDRESS LOW | | | | | | | | | | | | | | | |
| | | W | | | | | | | | | | | | | | | | |
| $10 | FIVAH1 | R | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | FAST INTERRUPT 1 VECTOR ADDRESS HIGH | | | | |
| | | W | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | |
| $11 | IRQP0 | R | PENDING [16:2] | | | | | | | | | | | | | | | 1 |
| | | W | | | | | | | | | | | | | | | | |
| $12 | IRQP1 | R | PENDING [32:17] | | | | | | | | | | | | | | | |
| | | W | | | | | | | | | | | | | | | | |
| $13 | IRQP2 | R | PENDING [48:33] | | | | | | | | | | | | | | | |
| | | W | | | | | | | | | | | | | | | | |
| $14 | IRQP3 | R | PENDING [64:49] | | | | | | | | | | | | | | | |
| | | W | | | | | | | | | | | | | | | | |
| $15 | IRQP4 | R | PENDING [80:65] | | | | | | | | | | | | | | | |
| | | W | | | | | | | | | | | | | | | | |
| $16 | IRQP5 | R | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | PEND-ING [81] |
| | | W | | | | | | | | | | | | | | | | |
| | Reserved | | | | | | | | | | | | | | | | | |
| $1D | ICTL | R | INT | IPIC | | | VAB | | | | | | INT_DIS | 1 | $\overline{\text{IRQB}}$ STATE | $\overline{\text{IRQA}}$ STATE | $\overline{\text{IRQB}}$ EDG | $\overline{\text{IRQA}}$ EDG |
| | | W | | | | | | | | | | | | | | | | |

= Reserved

## 5.6.3 Interrupt Priority Register 2 (IPR2)

| Base + $2 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Read | FMCBE IPL | | FMCC IPL | | FMERR IPL | | LOCK IPL | | LVI IPL | | 0 | 0 | IRQB IPL | | IRQA IPL | |
| Write | | | | | | | | | | | | | | | | |
| RESET | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Figure 5-5 Interrupt Priority Register 2 (IPR2)**

### 5.6.3.1 Flash Memory Command, Data, Address Buffers Empty Interrupt Priority Level (FMCBE IPL)—Bits 15–14

This field is used to set the interrupt priority level for IRQs. This IRQ is limited to priorities 0 through 2. It is disabled by default.

- 00 = IRQ disabled (default)
- 01 = IRQ is priority level 0
- 10 = IRQ is priority level 1
- 11 = IRQ is priority level 2

### 5.6.3.2 Flash Memory Command Complete Priority Level (FMCC IPL)—Bits 13–12

This field is used to set the interrupt priority level for IRQs. This IRQ is limited to priorities 0 through 2. It is disabled by default.

- 00 = IRQ disabled (default)
- 01 = IRQ is priority level 0
- 10 = IRQ is priority level 1
- 11 = IRQ is priority level 2

### 5.6.3.3 Flash Memory Error Interrupt Priority Level (FMERR IPL)—Bits 11–10

This field is used to set the interrupt priority level for IRQs. This IRQ is limited to priorities 0 through 2. It is disabled by default.

- 00 = IRQ disabled (default)
- 01 = IRQ is priority level 0
- 10 = IRQ is priority level 1
- 11 = IRQ is priority level 2

### 5.6.3.4 PLL Loss of Lock Interrupt Priority Level (LOCK IPL)—Bits 9–8

This field is used to set the interrupt priority level for IRQs. This IRQ is limited to priorities 0 through 2. It is disabled by default.

- 00 = IRQ disabled (default)
- 01 = IRQ is priority level 0
- 10 = IRQ is priority level 1
- 11 = IRQ is priority level 2

- 1 = No IRQ pending for this vector number

## 5.6.23 IRQ Pending 5 Register (IRQP5)

| Base + $16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Read | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | PEND-ING [81] |
| Write | | | | | | | | | | | | | | | | |
| RESET | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

**Figure 5-25 IRQ Pending Register 5 (IRQP5)**

### 5.6.23.1 Reserved—Bits 96–82

This bit field is reserved or not implemented. The bits are read as 1 and cannot be modified by writing.

### 5.6.23.2 IRQ Pending (PENDING)—Bit 81

This register combines with the other five to represent the pending IRQs for interrupt vector numbers 2 through 81.

- 0 = IRQ pending for this vector number
- 1 = No IRQ pending for this vector number

## 5.6.24 Reserved—Base + 17

## 5.6.25 Reserved—Base + 18

## 5.6.26 Reserved—Base + 19

## 5.6.27 Reserved—Base + 1A

## 5.6.28 Reserved—Base + 1B

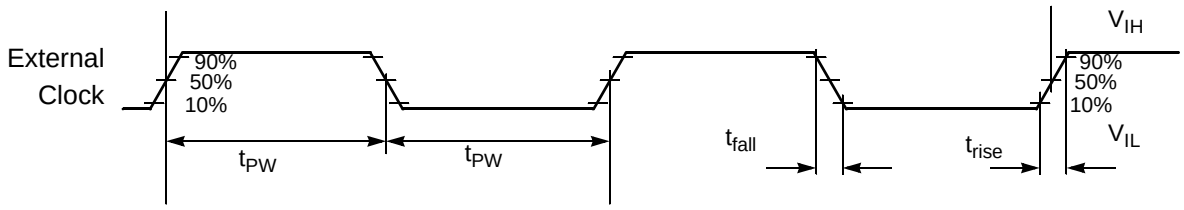
## 5.6.29 Reserved—Base + 1C

# 10.5 External Clock Operation Timing

**Table 10-13 External Clock Operation Timing Requirements[1]**

| Characteristic | Symbol | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| Frequency of operation (external clock driver)[2] | $f_{osc}$ | 0 | — | 120 | MHz |
| Clock Pulse Width[3] | $t_{PW}$ | 3.0 | — | — | ns |
| External clock input rise time[4] | $t_{rise}$ | — | — | 10 | ns |
| External clock input fall time[5] | $t_{fall}$ | — | — | 10 | ns |

1. Parameters listed are guaranteed by design.
2. See **Figure 10-3** for details on using the recommended connection of an external clock driver.
3. The high or low pulse width must be no smaller than 8.0ns or the chip will not function.
4. External clock input rise time is measured from 10% to 90%.
5. External clock input fall time is measured from 90% to 10%.




Note: The midpoint is $V_{IL} + (V_{IH} – V_{IL})/2$.

**Figure 10-3 External Clock Timing**

# 10.6 Phase Locked Loop Timing

**Table 10-14 PLL Timing**

| Characteristic | Symbol | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| External reference crystal frequency for the PLL[1] | $f_{osc}$ | 4 | 8 | 8.4 | MHz |
| PLL output frequency[2] ($f_{OUT}$) | $f_{op}$ | 160 | — | 260 | MHz |
| PLL stabilization time[3] -40° to +125°C | $t_{plls}$ | — | 1 | 10 | ms |

1. An externally supplied reference clock should be as free as possible from any phase jitter for the PLL to work correctly. The PLL is optimized for 8MHz input crystal.
2. ZCLK may not exceed 60MHz. For additional information on ZCLK and ($f_{OUT}$/2), please refer to the OCCS chapter in the **56F8300 Peripheral User Manual**.
3. This is the minimum time required after the PLL set up is changed to ensure reliable operation.

## 10.7 Crystal Oscillator Timing

**Table 10-15 Crystal Oscillator Parameters**

| Characteristic | Symbol | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| Crystal Start-up time | $T_{CS}$ | 4 | 5 | 10 | ms |
| Resonator Start-up time | $T_{RS}$ | 0.1 | 0.18 | 1 | ms |
| Crystal ESR | $R_{ESR}$ | — | — | 120 | ohms |
| Crystal Peak-to-Peak Jitter | $T_D$ | 70 | — | 250 | ps |
| Crystal Min-Max Period Variation | $T_{PV}$ | 0.12 | — | 1.5 | ns |
| Resonator Peak-to-Peak Jitter | $T_{RJ}$ | — | — | 300 | ps |
| Resonator Min-Max Period Variation | $T_{RP}$ | — | — | 300 | ps |
| Bias Current, high-drive mode | $I_{BIASH}$ | — | 250 | 290 | µA |
| Bias Current, low-drive mode | $I_{BIASL}$ | — | 80 | 110 | µA |
| Quiescent Current, power-down mode | $I_{PD}$ | — | 0 | 1 | µA |

## 10.8 Reset, Stop, Wait, Mode Select, and Interrupt Timing

**Table 10-16 Reset, Stop, Wait, Mode Select, and Interrupt Timing[1,2]**

| Characteristic | Symbol | Typical Min | Typical Max | Unit | See Figure |
|---|---|---|---|---|---|
| Minimum $\overline{\text{RESET}}$ Assertion Duration | $t_{RA}$ | 16T | — | ns | 10-4 |
| Edge-sensitive Interrupt Request Width | $t_{IRW}$ | 1.5T | — | ns | 10-5 |
| $\overline{\text{IRQA}}$, $\overline{\text{IRQB}}$ Assertion to General Purpose Output Valid, caused by first instruction execution in the interrupt service routine | $t_{IG}$ | 18T | — | ns | 10-6 |
| | $t_{IG}$ - FAST | 14T | — | | |
| $\overline{\text{IRQA}}$ Width Assertion to Recover from Stop State[3] | $t_{IW}$ | 1.5T | — | ns | 10-8 |

1. In the formulas, T = clock cycle. For an operating frequency of 60MHz, T = 16.67ns. At 8MHz (used during Reset and Stop modes), T = 125ns.
2. Parameters listed are guaranteed by design.
3. The interrupt instruction fetch is visible on the pins only in Mode 3.

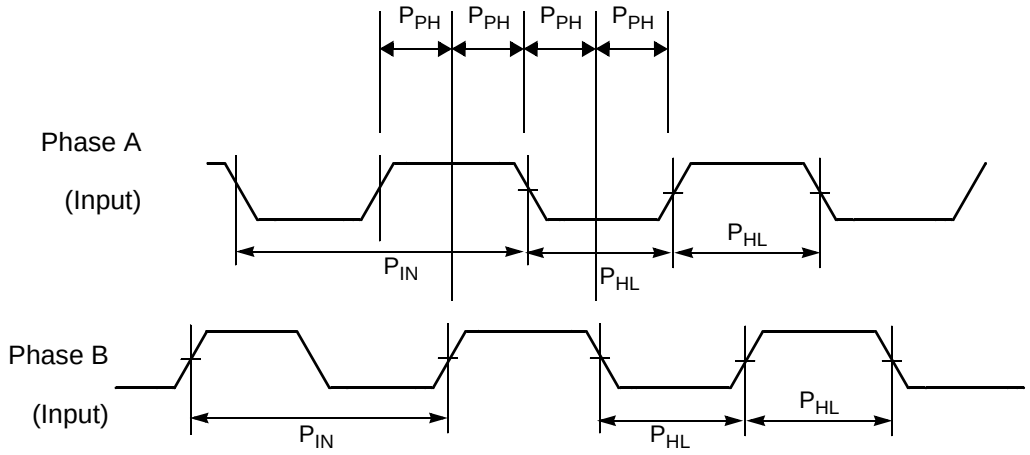



**Figure 10-14 Quadrature Decoder Timing**

## 10.12 Serial Communication Interface (SCI) Timing

**Table 10-20 SCI Timing[1]**

| Characteristic | Symbol | Min | Max | Unit | See Figure |
|---|---|---|---|---|---|
| Baud Rate[2] | BR | — | ($f_{MAX}$/16) | Mbps | — |
| RXD[3] Pulse Width | $RXD_{PW}$ | 0.965/BR | 1.04/BR | ns | 10-15 |
| TXD[4] Pulse Width | $TXD_{PW}$ | 0.965/BR | 1.04/BR | ns | 10-16 |

1. Parameters listed are guaranteed by design.
2. $f_{MAX}$ is the frequency of operation of the system clock, ZCLK, in MHz, which is 60MHz for the 56F8335 device and 40MHz for the 56F8135 device.
3. The RXD pin in SCI0 is named RXD0 and the RXD pin in SCI1 is named RXD1.
4. The TXD pin in SCI0 is named TXD0 and the TXD pin in SCI1 is named TXD1.

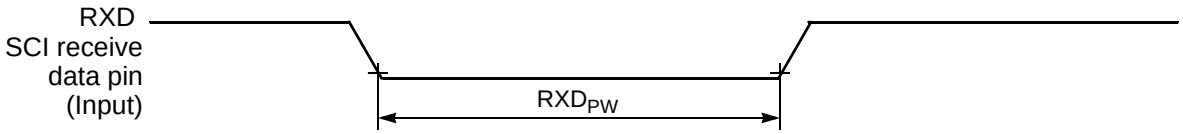



**Figure 10-15 RXD Pulse Width**


**Figure 10-16 TXD Pulse Width**

## Table 11-2 56F8135 128-Pin LQFP Package Identification by Pin Number (Continued)

| Pin No. | Signal Name | Pin No. | Signal Name | Pin No. | Signal Name | Pin No. | Signal Name |
|---|---|---|---|---|---|---|---|
| 3 | $V_{SS}$ | 35 | $V_{SS}$ | 67 | NC | 99 | ANB3 |
| 4 | $V_{DD\_IO}$ | 36 | $V_{DD\_IO}$ | 68 | NC | 100 | ANB4 |
| 5 | $V_{PP}2$ | 37 | PWMB3 | 69 | NC | 101 | ANB5 |
| 6 | CLKO | 38 | PWMB4 | 70 | NC | 102 | ANB6 |
| 7 | TXD0 | 39 | PWMB5 | 71 | OCR_DIS | 103 | ANB7 |
| 8 | RXD0 | 40 | TXD1 | 72 | $V_{DDA\_OSC\_PLL}$ | 104 | GPIOC8 |
| 9 | SCLK1 | 41 | RXD1 | 73 | XTAL | 105 | GPIOC9 |
| 10 | MOSI1 | 42 | GPIOD0 | 74 | EXTAL | 106 | GPIOC10 |
| 11 | MISO1 | 43 | GPIOD1 | 75 | $V_{CAP}3$ | 107 | GPIOE10 |
| 12 | $\overline{SS1}$ | 44 | GPIOD2 | 76 | $V_{DD\_IO}$ | 108 | GPIOE11 |
| 13 | $V_{CAP}4$ | 45 | GPIOD3 | 77 | $\overline{RSTO}$ | 109 | GPIOE12 |
| 14 | $V_{DD\_IO}$ | 46 | GPIOD4 | 78 | $\overline{RESET}$ | 110 | GPIOE13 |
| 15 | GPIOA0[1] | 47 | GPIOD5 | 79 | CLKMODE | 111 | TC0 |
| 16 | GPIOA1[1] | 48 | ISB0 | 80 | ANA0 | 112 | $V_{DD\_IO}$ |
| 17 | GPIOA2[1] | 49 | $V_{CAP}1$ | 81 | ANA1 | 113 | TC1 |
| 18 | GPIOA3[1] | 50 | ISB1 | 82 | ANA2 | 114 | $\overline{TRST}$ |
| 19 | GPIOA4[1] | 51 | ISB2 | 83 | ANA3 | 115 | TCK |
| 20 | GPIOA5[1] | 52 | $\overline{IRQA}$ | 84 | ANA4 | 116 | TMS |
| 21 | $V_{SS}$ | 53 | $\overline{IRQB}$ | 85 | ANA5 | 117 | TDI |
| 22 | GPIOF0[1] | 54 | FAULTB0 | 86 | ANA6 | 118 | TDO |
| 23 | GPIOF1[1] | 55 | FAULTB1 | 87 | ANA7 | 119 | $V_{PP}1$ |
| 24 | GPIOF2[1] | 56 | FAULTB2 | 88 | NC | 120 | NC |
| 25 | $V_{DD\_IO}$ | 57 | FAULTB3 | 89 | $V_{REFLO}$ | 121 | NC |
| 1. Primary function is not available in this package configuration; GPIO function must be used instead | | | | | | | |
| 26 | GPIOF3[1] | 58 | NC | 90 | $V_{REFN}$ | 122 | $V_{CAP}2$ |
| 27 | GPIOB0 | 59 | $V_{SS}$ | 91 | $V_{REFMID}$ | 123 | $\overline{SS0}$ |