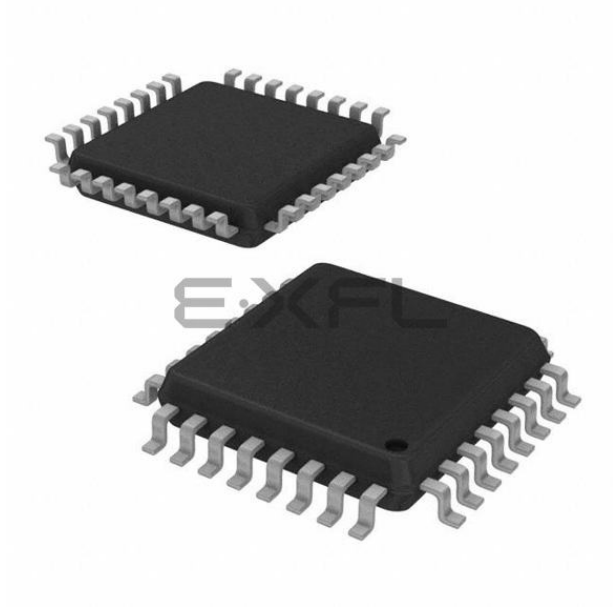


Welcome to E-XFL.COM

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	8051
Core Size	8-Bit
Speed	50MHz
Connectivity	SMBus (2-Wire/I²C), CANbus, LINbus, SPI, UART/USART
Peripherals	POR, PWM, Temp Sensor, WDT
Number of I/O	25
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	4.25K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 5.25V
Data Converters	A/D 25x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	32-LQFP
Supplier Device Package	32-LQFP (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051f506-iqr

Table 2.1. Product Selection Guide

Ordering Part Number	Flash Memory (kB)	CAN2.0B	LIN2.0	Digital Port I/Os	External Memory Interface	Package	Ordering Part Number	Flash Memory (kB)	CAN2.0B	LIN2.0	Digital Port I/Os	External Memory Interface	Package
C8051F500-IQ	64	✓	✓	40	✓	QFP-48	C8051F505-IQ	32	—	—	40	✓	QFP-48
C8051F500-IM	64	✓	✓	40	✓	QFN-48	C8051F505-IM	32	—	—	40	✓	QFN-48
C8051F501-IQ	64	—	—	40	✓	QFP-48	C8051F506-IQ	32	✓	✓	25	—	QFP-32
C8051F501-IM	64	—	—	40	✓	QFN-48	C8051F506-IM	32	✓	✓	25	—	QFN-32
C8051F502-IQ	64	✓	✓	25	—	QFP-32	C8051F507-IQ	32	—	—	25	—	QFP-32
C8051F502-IM	64	✓	✓	25	—	QFN-32	C8051F507-IM	32	—	—	25	—	QFN-32
C8051F503-IQ	64	—	—	25	—	QFP-32	C8051F508-IM	64	✓	✓	33	✓	QFN-40
C8051F503-IM	64	—	—	25	—	QFN-32	C8051F509-IM	64	—	—	33	✓	QFN-40
C8051F504-IQ	32	✓	✓	40	✓	QFP-48	C8051F510-IM	32	✓	✓	33	✓	QFN-40
C8051F504-IM	32	✓	✓	40	✓	QFN-48	C8051F511-IM	32	—	—	33	✓	QFN-40

Note: The suffix of the part number indicates the device rating and the package. All devices are RoHS compliant.

All of these devices are also available in an automotive version. For the automotive version, the -I in the ordering part number is replaced with -A. For example, the automotive version of the C8051F500-IM is the C8051F500-AM.

The -AM and -AQ devices receive full automotive quality production status, including AEC-Q100 qualification, registration with International Material Data System (IMDS) and Part Production Approval Process (PPAP) documentation. PPAP documentation is available at www.silabs.com with a registered and NDA approved user account. The -AM and -AQ devices enable high volume automotive OEM applications with their enhanced testing and processing. Please contact Silicon Labs sales for more information regarding –AM and -AQ devices for your automotive project.

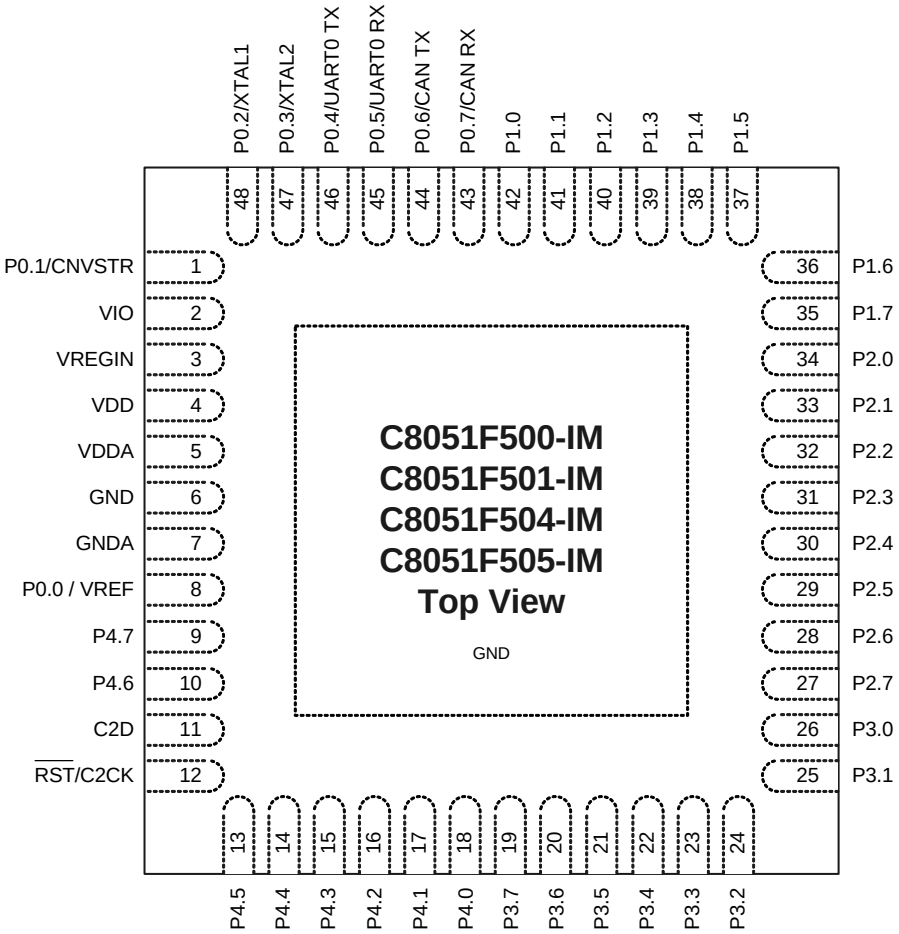


Figure 3.2. QFN-48 Pinout Diagram (Top View)

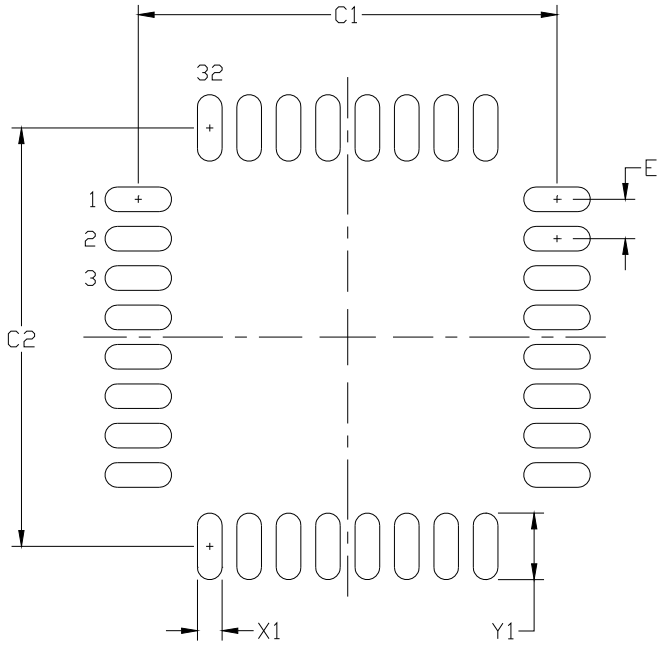


Figure 4.8. QFP-32 Package Drawing

Table 4.8. QFP-32 Landing Diagram Dimensions

Dimension	Min	Max	Dimension	Min	Max
C1	8.40	8.50	X1	0.40	0.50
C2	8.40	8.50	Y1	1.25	1.35
E	0.80 BSC				

Notes:

General

1. All dimensions shown are in millimeters (mm) unless otherwise noted.
2. This Land Pattern Design is based on the IPC-7351 guidelines.

Solder Mask Design

3. All metal pads are to be non-solder mask defined (NSMD). Clearance between the solder mask and the metal pad is to be 60µm minimum, all the way around the pad.

Stencil Design

4. A stainless steel, laser-cut and electro-polished stencil with trapezoidal walls should be used to assure good solder paste release.
5. The stencil thickness should be 0.125 mm (5 mils).
6. The ratio of stencil aperture to land pad size should be 1:1 for all perimeter pads.

Card Assembly

7. A No-Clean, Type-3 solder paste is recommended.
8. Recommended card reflow profile is per the JEDEC/IPC J-STD-020 specification for Small Body Components.

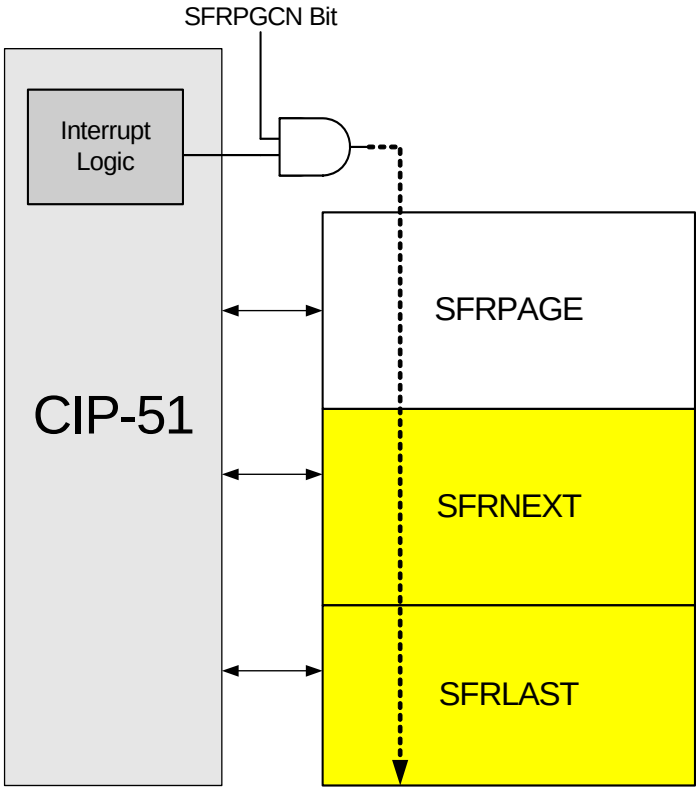


Figure 13.1. SFR Page Stack

Automatic hardware switching of the SFR Page on interrupts may be enabled or disabled as desired using the SFR Automatic Page Control Enable Bit located in the SFR Page Control Register (SFR0CN). This function defaults to “enabled” upon reset. In this way, the autoswitching function will be enabled unless disabled in software.

A summary of the SFR locations (address and SFR page) are provided in Table 13.3 in the form of an SFR memory map. Each memory location in the map has an SFR page row, denoting the page in which that SFR resides. Certain SFRs are accessible from ALL SFR pages, and are denoted by the “(ALL PAGES)” designation. For example, the Port I/O registers P0, P1, P2, and P3 all have the “(ALL PAGES)” designation, indicating these SFRs are accessible from all SFR pages regardless of the SFRPAGE register value.

13.3. SFR Page Stack Example

The following is an example that shows the operation of the SFR Page Stack during interrupts. In this example, the SFR Control register is left in the default enabled state (i.e., SFRPGEN = 1), and the CIP-51 is executing in-line code that is writing values to SPI Data Register (SFR “SPI0DAT”, located at address 0xA3 on SFR Page 0x00). The device is also using the CAN peripheral (CAN0) and the Programmable Counter Array (PCA0) peripheral to generate a PWM output. The PCA is timing a critical control function in its interrupt service round so its associated ISR that is set to low priority. At this point, the SFR page is set to access the SPI0DAT SFR (SFRPAGE = 0x00). See Figure 13.2.

Table 13.2. Special Function Register (SFR) Memory Map for Page 0xC

	0(8)	1(9)	2(A)	3(B)	4(C)	5(D)	6(E)	7(F)
F8			CAN0IF2DA2L	CAN0IF2DA2H	CAN0IF2DB1L	CAN0IF2DB1H	CAN0IF2DB2L	CAN0IF2DB2H
F0	B (All Pages)		CAN0IF2A2L	CAN0IF2A2H			CAN0IF2DA1L	CAN0IF2DA1H
E8			CAN0IF2M1L	CAN0IF2M1H	CAN0IF2M2L	CAN0IF2M2H	CAN0IF2A1L	CAN0IF2A1H
E0	ACC (All Pages)		CAN0IF2CML	CAN0IF2CMH			EIE1 (All Pages)	EIE2 (All Pages)
D8			CAN0IF1DB1L	CAN0IF1DB1H	CAN0IF1DB2L	CAN0IF1DB2H	CAN0IF2CRL	CAN0IF2CRH
D0	PSW (All Pages)		CAN0IF1MCL	CAN0IF1MCH	CAN0IF1DA1L	CAN0IF1DA1H	CAN0IF1DA2L	CAN0IF1DA2H
C8			CAN0IF1A1L	CAN0IF1A1H	CAN0IF1A2L	CAN0IF1A2H	CAN0IF2MCL	CAN0IF2MCH
C0	CAN0CN		CAN0IF1CML	CAN0IF1CMH	CAN0IF1M1L	CAN0IF1M1H	CAN0IF1M2L	CAN0IF1M2H
B8	IP (All Pages)		CAN0MV1L	CAN0MV1H	CAN0MV2L	CAN0MV2H	CAN0IF1CRL	CAN0IF1CRH
B0	P3 (All Pages)		CAN0IP2L	CAN0IP2H		P4 (All Pages)	FLSCL (All Pages)	FLKEY (All Pages)
A8	IE (All Pages)		CAN0ND1L	CAN0ND1H	CAN0ND2L	CAN0ND2H	CAN0IP1L	CAN0IP1H
A0	P2 (All Pages)	CAN0BRPE	CAN0TR1L	CAN0TR1H	CAN0TR2L	CAN0TR2H		SFRPAGE (All Pages)
98	SCON0 (All Pages)		CAN0BTL	CAN0BTH	CAN0IIDL	CAN0IIDH	CAN0TST	
90	P1 (All Pages)		CAN0CFG		CAN0STAT		CAN0ERRL	CAN0ERRH
88	TCON (All Pages)	TMOD (All Pages)	TL0 (All Pages)	TL1 (All Pages)	TH0 (All Pages)	TH1 (All Pages)	CKCON (All Pages)	
80	P0 (All Pages)	SP (All Pages)	DPL (All Pages)	DPH (All Pages)		SFRNEXT (All Pages)	SFRLAST (All Pages)	PCON (All Pages)
	0(8)	1(9)	2(A)	3(B)	4(C)	5(D)	6(E)	7(F)

(bit addressable)

Table 13.3. Special Function Registers (Continued)

SFRs are listed in alphabetical order. All undefined SFR locations are reserved

Register	Address	Description	Page
SMB0CF	0xC1	SMBus0 Configuration	232
SMB0CN	0xC0	SMBus0 Control	234
SMB0DAT	0xC2	SMBus0 Data	236
SMOD0	0xA9	UART0 Mode	249
SN0 - SN3	0xF9 - 0xFC	Serial Number Registers	96
SP	0x81	Stack Pointer	94
SPI0CFG	0xA1	SPI0 Configuration	259
SPI0CKR	0xA2	SPI0 Clock Rate Control	261
SPI0CN	0xF8	SPI0 Control	260
SPI0DAT	0xA3	SPI0 Data	261
TCON	0x88	Timer/Counter Control	271
TH0	0x8C	Timer/Counter 0 High	274
TH1	0x8D	Timer/Counter 1 High	274
TL0	0x8A	Timer/Counter 0 Low	273
TL1	0x8B	Timer/Counter 1 Low	273
TMOD	0x89	Timer/Counter Mode	272
TMR2CN	0xC8	Timer/Counter 2 Control	278
TMR2H	0xCD	Timer/Counter 2 High	280
TMR2L	0xCC	Timer/Counter 2 Low	280
TMR2RLH	0xCB	Timer/Counter 2 Reload High	279
TMR2RLL	0xCA	Timer/Counter 2 Reload Low	279
TMR3CN	0x91	Timer/Counter 3 Control	284
TMR3H	0x95	Timer/Counter 3 High	286
TMR3L	0x94	Timer/Counter 3 Low	286
TMR3RLH	0x93	Timer/Counter 3 Reload High	285
TMR3RLL	0x92	Timer/Counter 3 Reload Low	285
VDM0CN	0xFF	V_{DD} Monitor Control	144
XBR0	0xE1	Port I/O Crossbar Control 0	184
XBR1	0xE2	Port I/O Crossbar Control 1	185
XBR2	0xC7	Port I/O Crossbar Control 2	186

Note: The CAN registers are not explicitly defined in this datasheet. See Table 22.2 on page 223 for the list of all available CAN registers.

17.1. Power-On Reset

During power-up, the device is held in a reset state and the $\overline{\text{RST}}$ pin is driven low until V_{DD} settles above V_{RST}. A delay occurs before the device is released from reset; the delay decreases as the V_{DD} ramp time increases (V_{DD} ramp time is defined as how fast V_{DD} ramps from 0 V to V_{RST}). Figure 17.2. plots the power-on and V_{DD} monitor reset timing.

On exit from a power-on reset, the PORSF flag (RSTSRC.1) is set by hardware to logic 1. When PORSF is set, all of the other reset flags in the RSTSRC Register are indeterminate (PORSF is cleared by all other resets). Since all resets cause program execution to begin at the same location (0x0000) software can read the PORSF flag to determine if a power-up was the cause of reset. The content of internal data memory should be assumed to be undefined after a power-on reset. The V_{DD} monitor is enabled following a power-on reset.

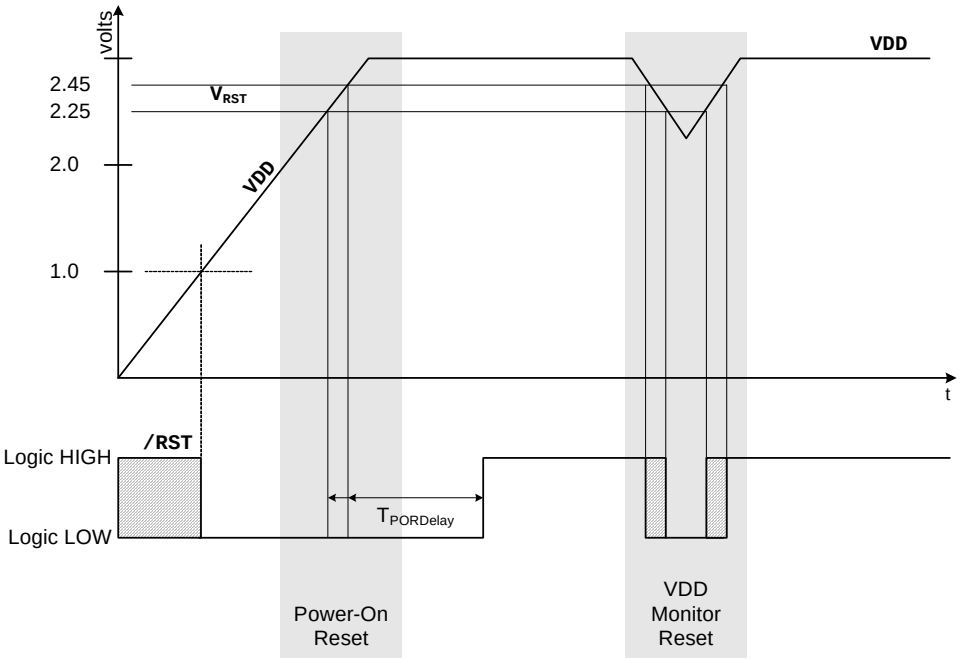


Figure 17.2. Power-On and V_{DD} Monitor Reset Timing

17.2. Power-Fail Reset/V_{DD} Monitor

When a power-down transition or power irregularity causes V_{DD} to drop below V_{RST}, the power supply monitor will drive the $\overline{\text{RST}}$ pin low and hold the CIP-51 in a reset state (see Figure 17.2). When V_{DD} returns to a level above V_{RST}, the CIP-51 will be released from the reset state. Note that even though internal data memory contents are not altered by the power-fail reset, it is impossible to determine if V_{DD} dropped below the level required for data retention. If the PORSF flag reads 1, the data may no longer be valid. The V_{DD} monitor is enabled after power-on resets. Its defined state (enabled/disabled) is not altered by any other reset source. For example, if the V_{DD} monitor is disabled by code and a software reset is performed, the V_{DD} monitor will still be disabled after the reset. **To protect the integrity of Flash contents, the V_{DD} monitor must be enabled to the higher setting (VDMLVL = 1) and selected as a reset source if software contains routines which erase or write Flash memory. If the V_{DD} monitor is not enabled and set to the high level, any erase or write performed on Flash memory will cause a Flash Error device reset.**

18.6.1. Non-Multiplexed Mode

18.6.1.1. 16-bit MOVX: EMI0CF[4:2] = 101, 110, or 111

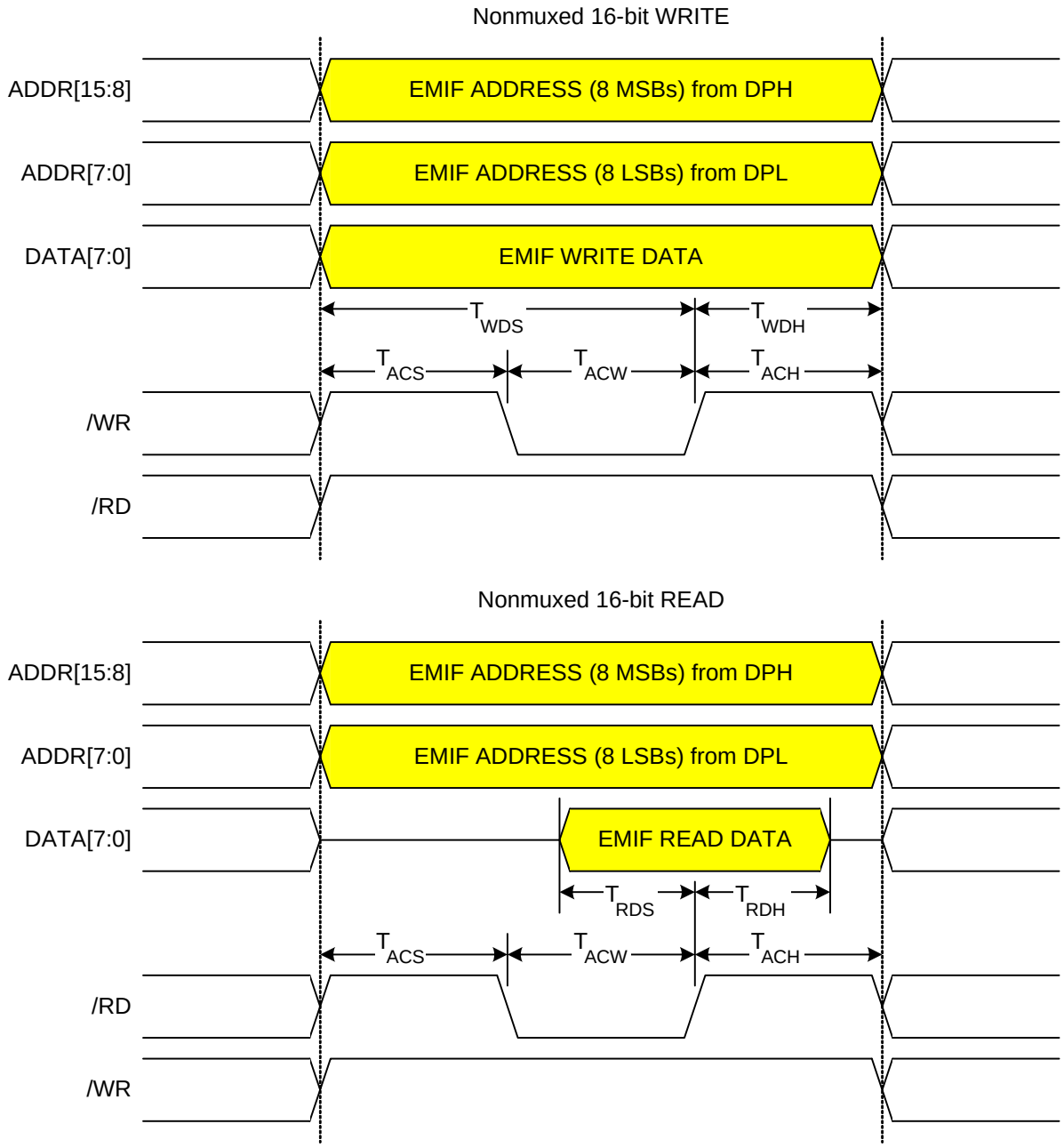


Figure 18.4. Non-multiplexed 16-bit MOVX Timing

18.6.2.3. 8-bit MOVX with Bank Select: EMI0CF[4:2] = 010

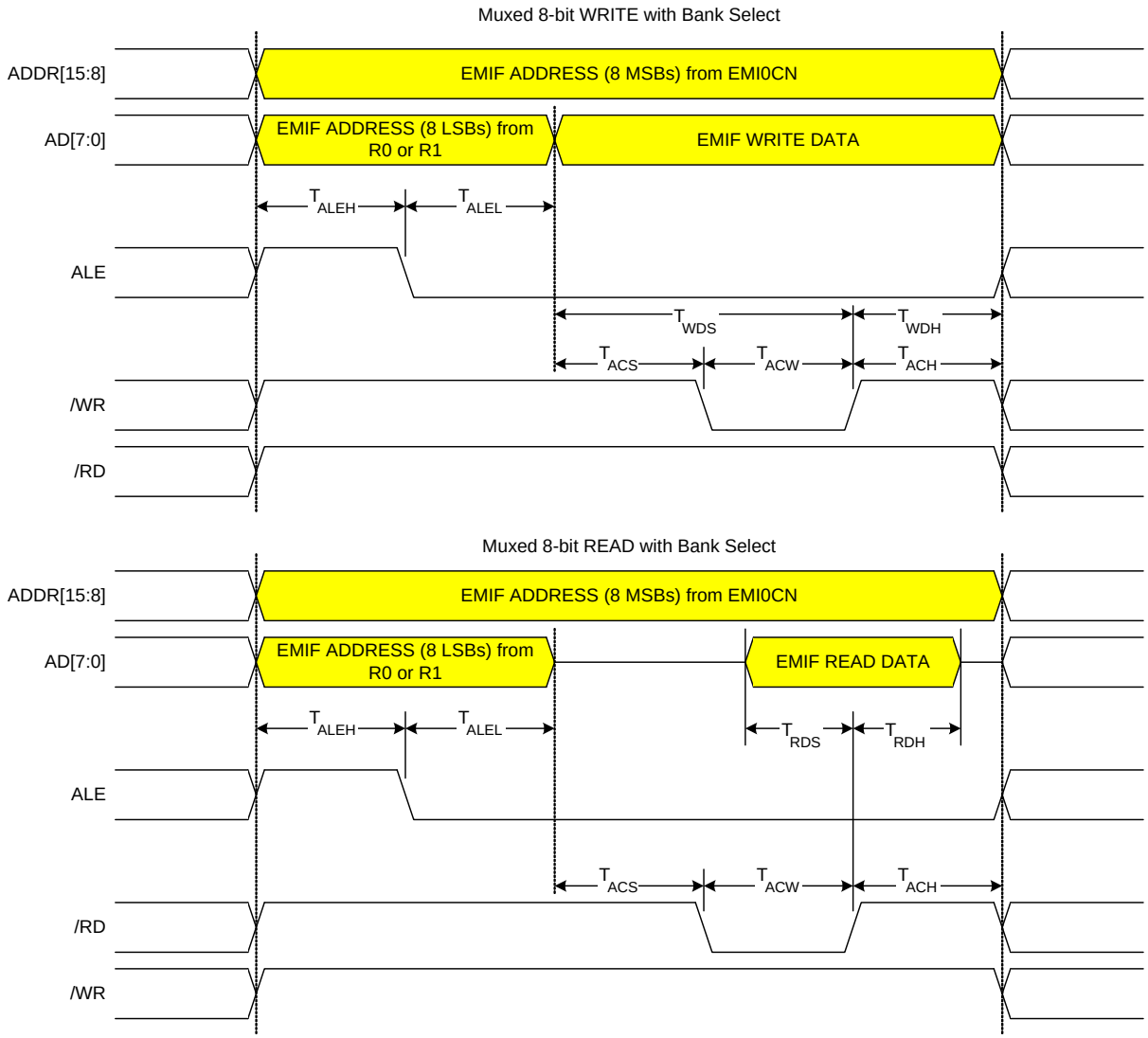


Figure 18.9. Multiplexed 8-bit MOVX with Bank Select Timing

19.3. Clock Multiplier

The Clock Multiplier generates an output clock which is 4 times the input clock frequency scaled by a programmable factor of 1, 2/3, 2/4 (or 1/2), 2/5, 2/6 (or 1/3), or 2/7. The Clock Multiplier's input can be selected from the external oscillator, or the internal or external oscillators divided by 2. This produces three possible base outputs which can be scaled by a programmable factor: Internal Oscillator x 2, External Oscillator x 2, or External Oscillator x 4. See Section 19.1 on page 165 for details on system clock selection.

The Clock Multiplier is configured via the CLKMUL register (SFR Definition 19.5). The procedure for configuring and enabling the Clock Multiplier is as follows:

1. Reset the Multiplier by writing 0x00 to register CLKMUL.
2. Select the Multiplier input source via the MULSEL bits.
3. Select the Multiplier output scaling factor via the MULDIV bits
4. Enable the Multiplier with the MULEN bit (CLKMUL | = 0x80).
5. Delay for >5 µs.
6. Initialize the Multiplier with the MULINIT bit (CLKMUL | = 0xC0).
7. Poll for MULRDY => 1.

Important Note: When using an external oscillator as the input to the Clock Multiplier, the external source must be enabled and stable before the Multiplier is initialized. See "19.4. External Oscillator Drive Circuit" on page 172 for details on selecting an external oscillator source.

The Clock Multiplier allows faster operation of the CIP-51 core and is intended to generate an output frequency between 25 and 50 MHz. The clock multiplier can also be used with slow input clocks. However, if the clock is below the minimum Clock Multiplier input frequency (FCM_{min}), the generated clock will consist of four fast pulses followed by a long delay until the next input clock rising edge. The average frequency of the output is equal to 4x the input, but the instantaneous frequency may be faster. See Figure 19.2 below for more information.

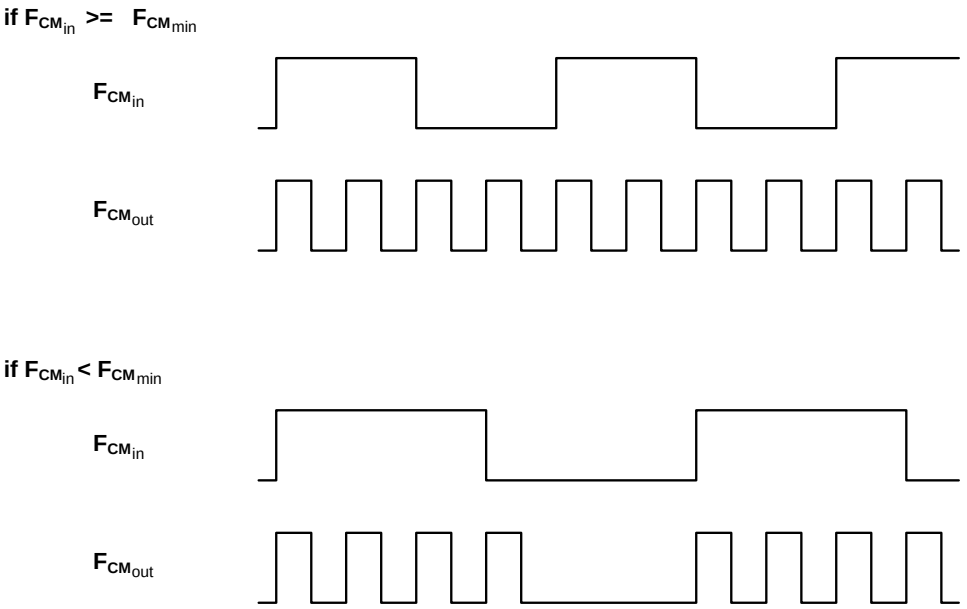


Figure 19.2. Example Clock Multiplier Output

19.4.1. External Crystal Example

If a crystal or ceramic resonator is used as an external oscillator source for the MCU, the circuit should be configured as shown in Figure 19.1, Option 1. The External Oscillator Frequency Control value (XFCN) should be chosen from the Crystal column of the table in SFR Definition 19.6 (OSCXCN register). For example, an 11.0592 MHz crystal requires an XFCN setting of 111b and a 32.768 kHz Watch Crystal requires an XFCN setting of 001b. After an external 32.768 kHz oscillator is stabilized, the XFCN setting can be switched to 000 to save power. It is recommended to enable the missing clock detector before switching the system clock to any external oscillator source.

When the crystal oscillator is first enabled, the oscillator amplitude detection circuit requires a settling time to achieve proper bias. Introducing a delay of 1 ms between enabling the oscillator and checking the XTLVLD bit will prevent a premature switch to the external oscillator as the system clock. Switching to the external oscillator before the crystal oscillator has stabilized can result in unpredictable behavior. The recommended procedure is:

1. Force XTAL1 and XTAL2 to a high state. This involves enabling the Crossbar and writing 1 to the port pins associated with XTAL1 and XTAL2.
2. Configure XTAL1 and XTAL2 as analog inputs using.
3. Enable the external oscillator.
4. Wait at least 1 ms.
5. Poll for XTLVLD => 1.
6. Enable the Missing Clock Detector.
7. Switch the system clock to the external oscillator.

Important Note on External Crystals: Crystal oscillator circuits are quite sensitive to PCB layout. The crystal should be placed as close as possible to the XTAL pins on the device. The traces should be as short as possible and shielded with ground plane from any other traces which could introduce noise or interference.

The capacitors shown in the external crystal configuration provide the load capacitance required by the crystal for correct oscillation. These capacitors are "in series" as seen by the crystal and "in parallel" with the stray capacitance of the XTAL1 and XTAL2 pins.

Note: The desired load capacitance depends upon the crystal and the manufacturer. Refer to the crystal data sheet when completing these calculations.

For example, a tuning-fork crystal of 32.768 kHz with a recommended load capacitance of 12.5 pF should use the configuration shown in Figure 19.1, Option 1. The total value of the capacitors and the stray capacitance of the XTAL pins should equal 25 pF. With a stray capacitance of 3 pF per pin, the 22 pF capacitors yield an equivalent capacitance of 12.5 pF across the crystal, as shown in Figure 19.3.

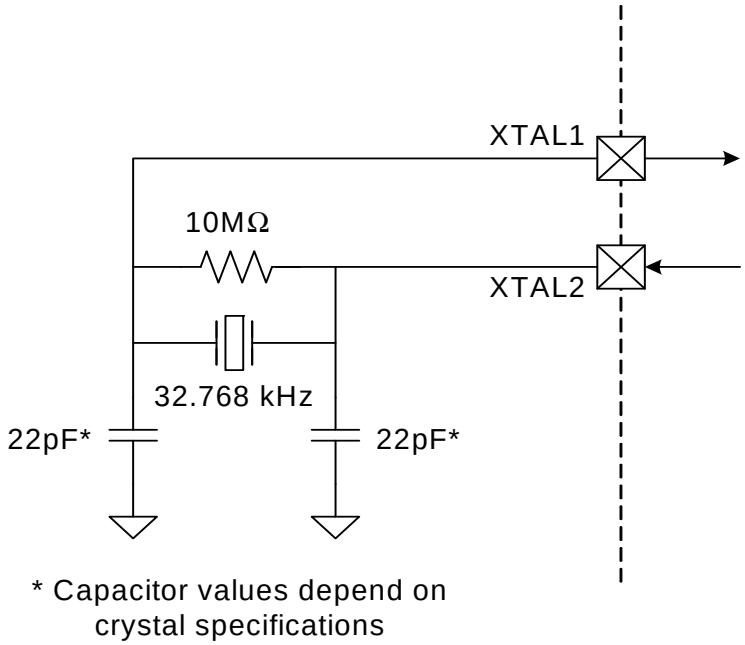


Figure 19.3. External 32.768 kHz Quartz Crystal Oscillator Connection Diagram

19.4.2. External RC Example

If an RC network is used as an external oscillator source for the MCU, the circuit should be configured as shown in Figure 19.1, Option 2. The capacitor should be no greater than 100 pF; however for very small capacitors, the total capacitance may be dominated by parasitic capacitance in the PCB layout. To determine the required External Oscillator Frequency Control value (XFCN) in the OSCXCN Register, first select the RC network value to produce the desired frequency of oscillation, according to Equation , where f = the frequency of oscillation in MHz, C = the capacitor value in pF, and R = the pull-up resistor value in kΩ.

$$f = 1.23 \times 10^3 / (R \times C)$$

Equation 19.1. RC Mode Oscillator Frequency

For example: If the frequency desired is 100 kHz, let R = 246 kΩ and C = 50 pF:

f = 1.23(10^3)/RC = 1.23(10^3)/[246 x 50] = 0.1 MHz = 100 kHz

Referring to the table in SFR Definition 19.6, the required XFCN setting is 010b.

19.4.3. External Capacitor Example

If a capacitor is used as an external oscillator for the MCU, the circuit should be configured as shown in Figure 19.1, Option 3. The capacitor should be no greater than 100 pF; however for very small capacitors, the total capacitance may be dominated by parasitic capacitance in the PCB layout. To determine the required External Oscillator Frequency Control value (XFCN) in the OSCXCN Register, select the capacitor to be used and find the frequency of oscillation according to Equation 19.2, where f = the frequency of oscillation in MHz, C = the capacitor value in pF, and V_{DD} = the MCU power supply in Volts.

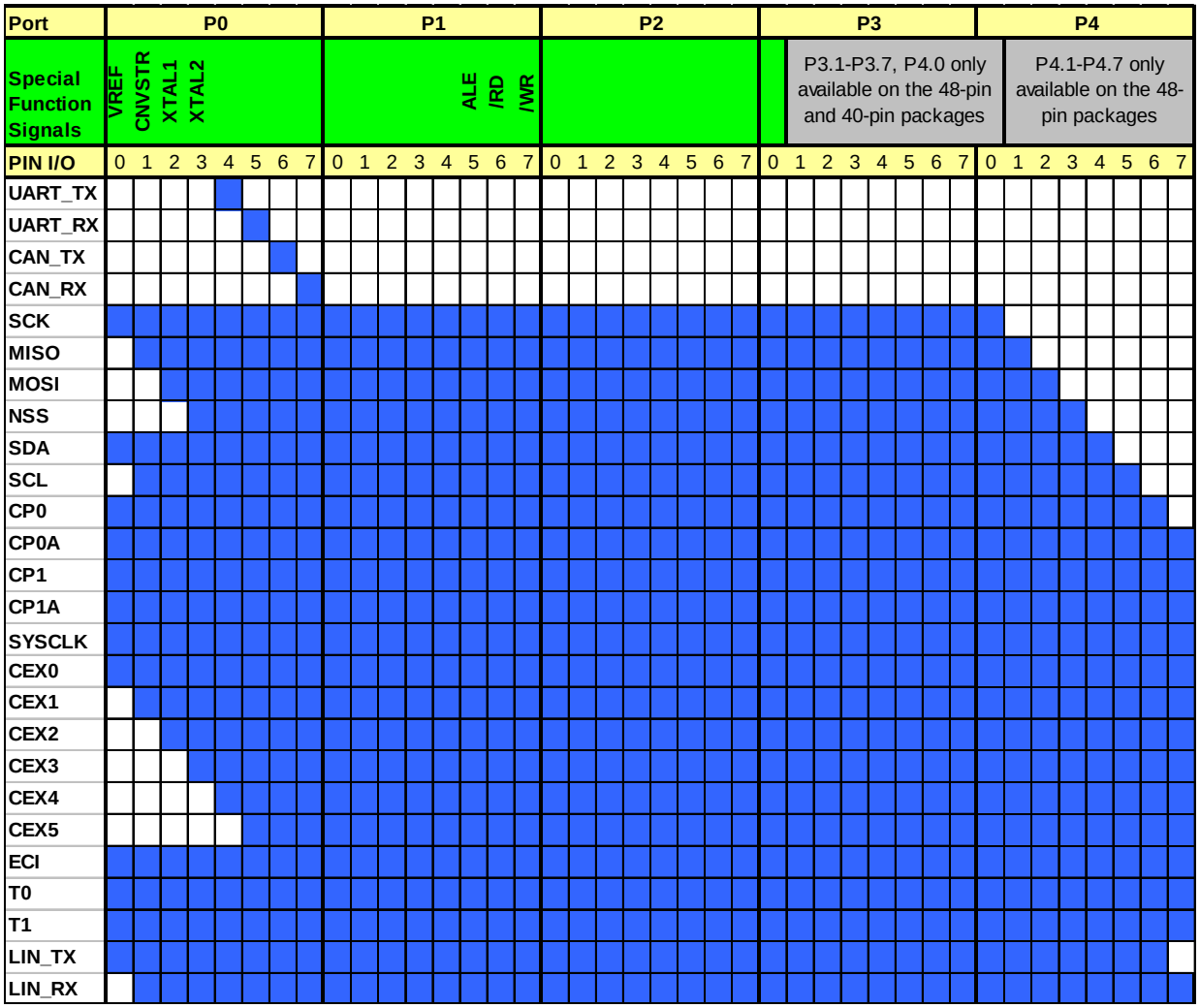


Figure 20.3. Peripheral Availability on Port I/O Pins

Registers XBR0, XBR1, and XBR2 are used to assign the digital I/O resources to the physical I/O Port pins. Note that when the SMBus is selected, the Crossbar assigns both pins associated with the SMBus (SDA and SCL); and similarly when the UART, CAN or LIN are selected, the Crossbar assigns both pins associated with the peripheral (TX and RX). UART0 pin assignments are fixed for bootloading purposes: UART TX0 is always assigned to P0.4; UART RX0 is always assigned to P0.5. CAN0 pin assignments are fixed to P0.6 for CAN_TX and P0.7 for CAN_RX. Standard Port I/Os appear contiguously after the prioritized functions have been assigned.

Important Note: The SPI can be operated in either 3-wire or 4-wire modes, pending the state of the NSSMD1–NSSMD0 bits in register SPI0CN. According to the SPI mode, the NSS signal may or may not be routed to a Port pin.

As an example configuration, if CAN0, SPI0 in 4-wire mode, and PCA0 Modules 0, 1, and 2 are enabled on the crossbar with P0.1, P0.2, and P0.5 skipped, the registers should be set as follows: XBR0 = 0x06 (CAN0 and SPI0 enabled), XBR1 = 0x0C (PCA0 modules 0, 1, and 2 enabled), XBR2 = 0x40 (Crossbar enabled), and P0SKIP = 0x26 (P0.1, P0.2, and P0.5 skipped). The resulting crossbar would look as shown in Figure 20.4.

20.13 for the PnMDIN register details.

The output driver characteristics of the I/O pins are defined using the Port Output Mode registers (PnMDOUT). Each Port Output driver can be configured as either open drain or push-pull. This selection is required even for the digital resources selected in the XBRn registers, and is not automatic. The only exception to this is the SMBus (SDA, SCL) pins, which are configured as open-drain regardless of the PnMDOUT settings. When the WEAKPUD bit in XBR2 is 0, a weak pullup is enabled for all Port I/O configured as open-drain. WEAKPUD does not affect the push-pull Port I/O. Furthermore, the weak pullup is turned off on an output that is driving a 0 to avoid unnecessary power dissipation.

Registers XBR0, XBR1, and XBR2 must be loaded with the appropriate values to select the digital I/O functions required by the design. Setting the XBARE bit in XBR2 to 1 enables the Crossbar. Until the Crossbar is enabled, the external pins remain as standard Port I/O (in input mode), regardless of the XBRn Register settings. For given XBRn Register settings, one can determine the I/O pin-out using the Priority Decode Table; as an alternative, the Configuration Wizard utility of the Silicon Labs IDE software will determine the Port I/O pin-assignments based on the XBRn Register settings.

The Crossbar must be enabled to use Port pins as standard Port I/O in output mode. Port output drivers are disabled while the Crossbar is disabled.

SFR Definition 20.6. P1MASK: Port 1 Mask Register

Bit	7	6	5	4	3	2	1	0
Name	P1MASK[7:0]							
Type	R/W							
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xF4; SFR Page = 0x00

Bit	Name	Function
7:0	P1MASK[7:0]	**Port 1 Mask Value.** Selects P1 pins to be compared to the corresponding bits in P1MAT. 0: P1.n pin logic value is ignored and cannot cause a Port Mismatch event. 1: P1.n pin logic value is compared to P1MAT.n.

SFR Definition 20.7. P1MAT: Port 1 Match Register

Bit	7	6	5	4	3	2	1	0
Name	P1MAT[7:0]							
Type	R/W							
Reset	1	1	1	1	1	1	1	1

SFR Address = 0xF3; SFR Page = 0x00

Bit	Name	Function
7:0	P1MAT[7:0]	**Port 1 Match Value.** Match comparison value used on Port 1 for bits in P1MAT which are set to 1. 0: P1.n pin logic value is compared with logic LOW. 1: P1.n pin logic value is compared with logic HIGH.

SFR Definition 20.17. P1MDIN: Port 1 Input Mode

Bit	7	6	5	4	3	2	1	0
Name	P1MDIN[7:0]							
Type	R/W							
Reset	1	1	1	1	1	1	1	1

SFR Address = 0xF2; SFR Page = 0x0F

Bit	Name	Function
7:0	P1MDIN[7:0]	**Analog Configuration Bits for P1.7–P1.0 (respectively).** Port pins configured for analog mode have their weak pull-up and digital receiver disabled. For analog mode, the pin also needs to be configured for open-drain mode in the P1MDOUT register. 0: Corresponding P1.n pin is configured for analog mode. 1: Corresponding P1.n pin is not configured for analog mode.

SFR Definition 20.18. P1MDOUT: Port 1 Output Mode

Bit	7	6	5	4	3	2	1	0
Name	P1MDOUT[7:0]							
Type	R/W							
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xA5; SFR Page = 0x0F

Bit	Name	Function
7:0	P1MDOUT[7:0]	**Output Configuration Bits for P1.7–P1.0 (respectively).** These bits are ignored if the corresponding bit in register P1MDIN is logic 0. 0: Corresponding P1.n Output is open-drain. 1: Corresponding P1.n Output is push-pull.

21.1. Software Interface with the LIN Controller

The selection of the mode (Master or Slave) and the automatic baud rate feature are done though the LIN0 Control Mode (LIN0CF) register. The other LIN registers are accessed indirectly through the two SFRs LIN0 Address (LIN0ADR) and LIN0 Data (LIN0DAT). The LIN0ADR register selects which LIN register is targeted by reads/writes of the LIN0DAT register. The full list of indirectly-accessible LIN registers is given in Table 21.4 on page 210.

21.2. LIN Interface Setup and Operation

The hardware based LIN controller allows for the implementation of both Master and Slave nodes with minimal firmware overhead and complete control of the interface status while allowing for interrupt and polled mode operation.

The first step to use the controller is to define the basic characteristics of the node:

Mode—Master or Slave

Baud Rate—Either defined manually or using the autobaud feature (slave mode only)

Checksum Type—Select between classic or enhanced checksum, both of which are implemented in hardware.

21.2.1. Mode Definition

Following the LIN specification, the controller implements in hardware both the Slave and Master operating modes. The mode is configured using the MODE bit (LIN0CF.6).

21.2.2. Baud Rate Options: Manual or Autobaud

The LIN controller can be selected to have its baud rate calculated manually or automatically. A master node must always have its baud rate set manually, but slave nodes can choose between a manual or automatic setup. The configuration is selected using the ABAUD bit (LIN0CF.5).

Both the manual and automatic baud rate configurations require additional setup. The following sections explain the different options available and their relation with the baud rate, along with the steps necessary to achieve the required baud rate.

21.2.3. Baud Rate Calculations: Manual Mode

The baud rate used by the LIN controller is a function of the System Clock (SYSCLK) and the LIN timing registers according to the following equation:

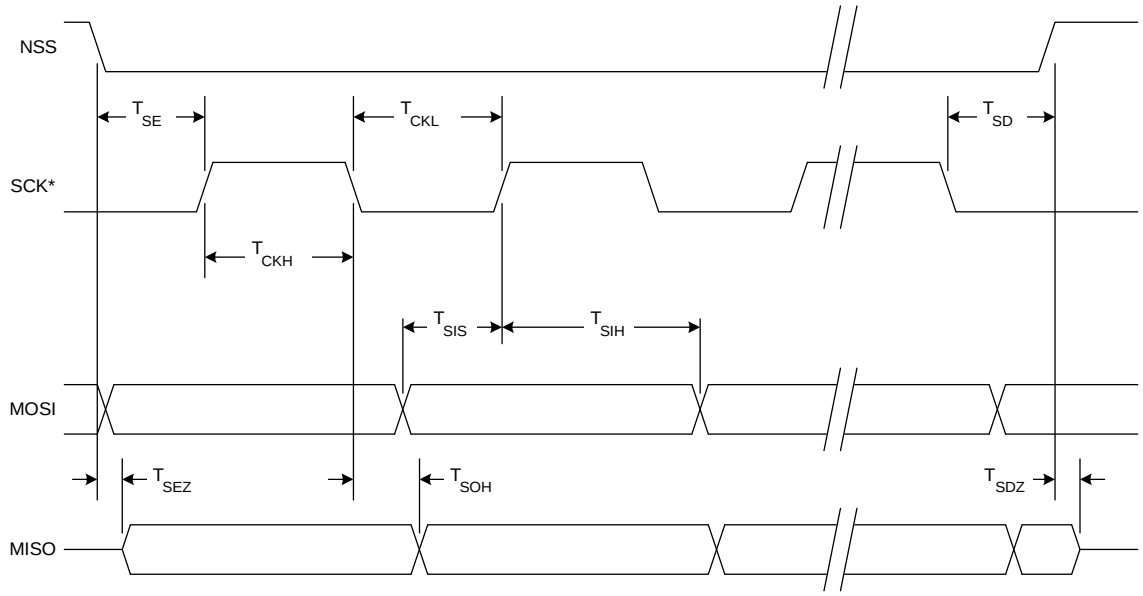
$$baud\_rate = \frac{SYSCLK}{2^{(prescaler+1)} \times divider \times (multiplier+1)}$$

The prescaler, divider and multiplier factors are part of the LIN0DIV and LIN0MUL registers and can assume values in the following range:

Table 21.1. Baud Rate Calculation Variable Ranges

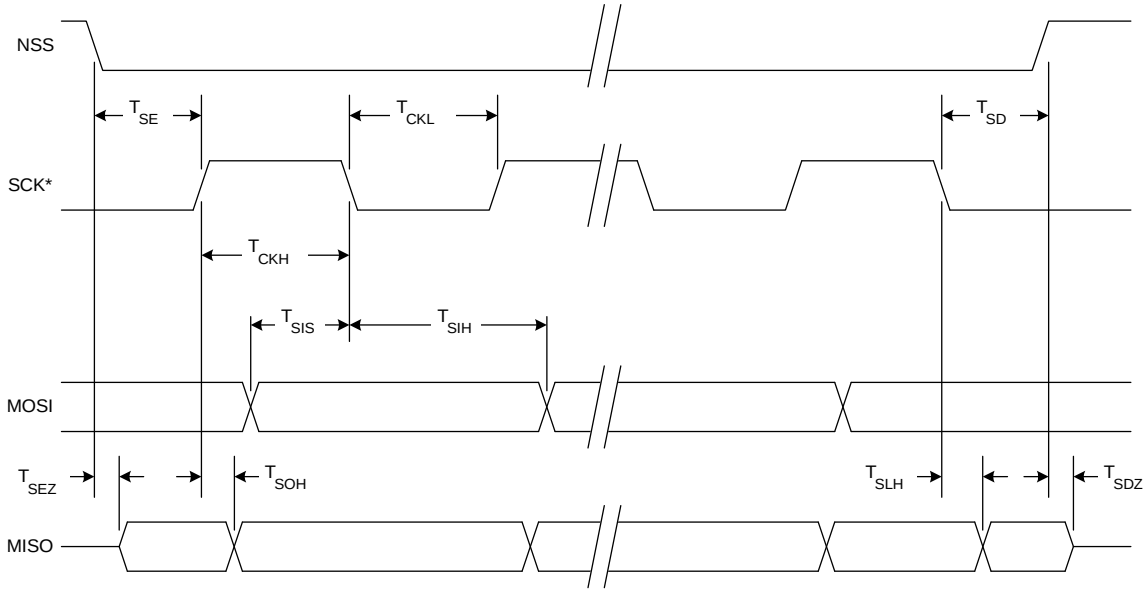
Factor	Range
prescaler	0…3
multiplier	0…31
divider	200…511

Important Note: The minimum system clock (SYSCLK) to operate the LIN controller is 8 MHz.



* SCK is shown for CKPOL = 0. SCK is the opposite polarity for CKPOL = 1.

Figure 25.10. SPI Slave Timing (CKPHA = 0)



* SCK is shown for CKPOL = 0. SCK is the opposite polarity for CKPOL = 1.

Figure 25.11. SPI Slave Timing (CKPHA = 1)

SFR Definition 26.3. TMOD: Timer Mode

Bit	7	6	5	4	3	2	1	0
Name	GATE1	C/T1	T1M[1:0]		GATE0	C/T0	T0M[1:0]	
Type	R/W	R/W	R/W		R/W	R/W	R/W	
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x89; SFR Page = All Pages

Bit	Name	Function
7	GATE1	**Timer 1 Gate Control.** 0: Timer 1 enabled when TR1 = 1 irrespective of $\overline{\text{INT1}}$ logic level. 1: Timer 1 enabled only when TR1 = 1 AND $\overline{\text{INT1}}$ is active as defined by bit IN1PL in register IT01CF (see SFR Definition 14.7).
6	C/T1	**Counter/Timer 1 Select.** 0: Timer: Timer 1 incremented by clock defined by T1M bit in register CKCON. 1: Counter: Timer 1 incremented by high-to-low transitions on external pin (T1).
5:4	T1M[1:0]	**Timer 1 Mode Select.** These bits select the Timer 1 operation mode. 00: Mode 0, 13-bit Counter/Timer 01: Mode 1, 16-bit Counter/Timer 10: Mode 2, 8-bit Counter/Timer with Auto-Reload 11: Mode 3, Timer 1 Inactive
3	GATE0	**Timer 0 Gate Control.** 0: Timer 0 enabled when TR0 = 1 irrespective of $\overline{\text{INT0}}$ logic level. 1: Timer 0 enabled only when TR0 = 1 AND $\overline{\text{INT0}}$ is active as defined by bit IN0PL in register IT01CF (see SFR Definition 14.7).
2	C/T0	**Counter/Timer 0 Select.** 0: Timer: Timer 0 incremented by clock defined by T0M bit in register CKCON. 1: Counter: Timer 0 incremented by high-to-low transitions on external pin (T0).
1:0	T0M[1:0]	**Timer 0 Mode Select.** These bits select the Timer 0 operation mode. 00: Mode 0, 13-bit Counter/Timer 01: Mode 1, 16-bit Counter/Timer 10: Mode 2, 8-bit Counter/Timer with Auto-Reload 11: Mode 3, Two 8-bit Counter/Timers

SFR Definition 27.2. PCA0MD: PCA Mode

Bit	7	6	5	4	3	2	1	0
Name	CIDL	WDTE	WDLCK		CPS[2:0]			ECF
Type	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W
Reset	0	1	0	0	0	0	0	0

SFR Address = 0xD9; SFR Page = 0x00

Bit	Name	Function
7	CIDL	**PCA Counter/Timer Idle Control.** Specifies PCA behavior when CPU is in Idle Mode. 0: PCA continues to function normally while the system controller is in Idle Mode. 1: PCA operation is suspended while the system controller is in Idle Mode.
6	WDTE	**Watchdog Timer Enable** If this bit is set, PCA Module 5 is used as the watchdog timer. 0: Watchdog Timer disabled. 1: PCA Module 5 enabled as Watchdog Timer.
5	WDLCK	**Watchdog Timer Lock** This bit locks/unlocks the Watchdog Timer Enable. When WDLCK is set, the Watchdog Timer may not be disabled until the next system reset. 0: Watchdog Timer Enable unlocked. 1: Watchdog Timer Enable locked.
4	Unused	Read = 0b, Write = Don't care.
3:1	CPS[2:0]	**PCA Counter/Timer Pulse Select.** These bits select the timebase source for the PCA counter 000: System clock divided by 12 001: System clock divided by 4 010: Timer 0 overflow 011: High-to-low transitions on ECI (max rate = system clock divided by 4) 100: System clock 101: External clock divided by 8 (synchronized with the system clock) 11x: Reserved
0	ECF	**PCA Counter/Timer Overflow Interrupt Enable.** This bit sets the masking of the PCA Counter/Timer Overflow (CF) interrupt. 0: Disable the CF interrupt. 1: Enable a PCA Counter/Timer Overflow interrupt request when CF (PCA0CN.7) is set.
Note: When the WDTE bit is set to 1, the other bits in the PCA0MD register cannot be modified. To change the contents of the PCA0MD register, the Watchdog Timer must first be disabled.		