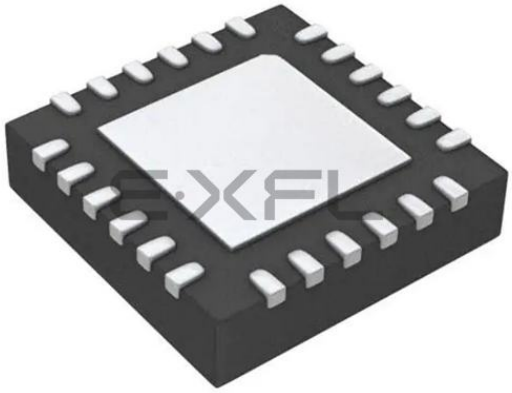

Welcome to E-XFL.COM

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	Obsolete
Core Processor	8051
Core Size	8-Bit
Speed	25MHz
Connectivity	SMBus (2-Wire/I²C), SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, Temp Sensor, WDT
Number of I/O	16
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	4.25K x 8
Voltage - Supply (Vcc/Vdd)	0.9V ~ 3.6V
Data Converters	A/D 15x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	24-WFQFN Exposed Pad
Supplier Device Package	24-QFN (4x4)
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051f921-gm

Table 3.1. Pin Definitions for the C8051F92x-C8051F93x (Continued)

Name	Pin Numbers		Type	Description
	‘F920/30	‘F921/31		
P1.0	24	16	D I/O or A In	Port 1.0. See Port I/O Section for a complete description. May also be used as SCK for SPI1.
AD0*			D I/O	Address/Data 0.
P1.1	23	15	D I/O or A In	Port 1.1. See Port I/O Section for a complete description. May also be used as MISO for SPI1.
AD1*			D I/O	Address/Data 1.
P1.2	22	14	D I/O or A In	Port 1.2. See Port I/O Section for a complete description. May also be used as MOSI for SPI1.
AD2*			D I/O	Address/Data 2.
P1.3	21	13	D I/O or A In	Port 1.3. See Port I/O Section for a complete description. May also be used as NSS for SPI1.
AD3*			D I/O	Address/Data 3.
P1.4	20	12	D I/O or A In	Port 1.4. See Port I/O Section for a complete description.
AD4*			D I/O	Address/Data 4.
P1.5	19	11	D I/O or A In	Port 1.5. See Port I/O Section for a complete description.
AD5*			D I/O	Address/Data 5.
P1.6	18	10	D I/O or A In	Port 1.6. See Port I/O Section for a complete description.
AD6*			D I/O	Address/Data 6.
P1.7*	17		D I/O or A In	Port 1.7. See Port I/O Section for a complete description.
AD7*			D I/O	Address/Data 7.
P2.0*	16		D I/O or A In	Port 2.0. See Port I/O Section for a complete description.
AD8*			D I/O	Address/Data 8.
***Note:** Available only on the C8051F920/30.				

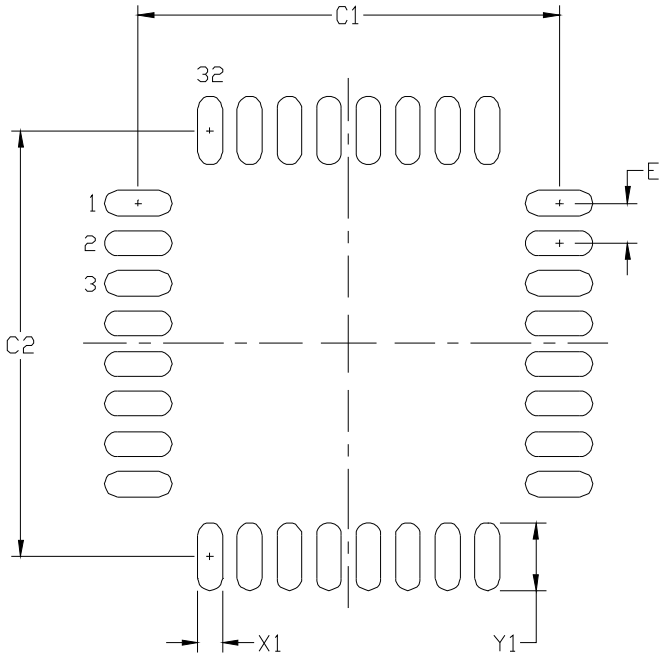

Figure 3.9. Typical LQFP-32 Landing Diagram

Table 3.7. PCB Land Pattern

Dimension	MIN	MAX
C1	8.40	8.50
C2	8.40	8.50
E	0.80 BSC	
X1	0.40	0.50
Y1	1.25	1.35

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

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

Stencil Design

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

Card Assembly

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

Table 4.4. Reset Electrical Characteristics

V_{DD} = 1.8 to 3.6 V, –40 to +85 °C unless otherwise specified.

Parameter	Conditions	Min	Typ	Max	Units
$\overline{RST}$ Output Low Voltage	I_{OL} = 1.4 mA,	—	—	0.6	V
$\overline{RST}$ Input High Voltage	V_{DD} = 2.0 to 3.6 V	V_{DD} – 0.6	—	—	V
	V_{DD} = 0.9 to 2.0 V	0.7 x V_{DD}	—	—	V
$\overline{RST}$ Input Low Voltage	V_{DD} = 2.0 to 3.6 V	—	—	0.6	V
	V_{DD} = 0.9 to 2.0 V	—	—	0.3 x V_{DD}	V
$\overline{RST}$ Input Pullup Current	$\overline{RST}$ = 0.0 V, VDD = 1.8 V	—	4	—	µA
	$\overline{RST}$ = 0.0 V, VDD = 3.6 V	—	20	30	
VDD/DC+ Monitor Threshold (V_{RST})	Early Warning	1.8	1.85	1.9	V
	Reset Trigger (all power modes except Sleep)	1.7	1.75	1.8	
VBAT Ramp Time for Power On	VBAT Ramp from 0–0.9 V	—	—	3	ms
VBAT Monitor Threshold (V_{POR})	Initial Power-On (VBAT Rising)	—	0.75	—	V
	Brownout Condition (VBAT Falling)	0.7	0.8	0.9	
	Recovery from Brownout (VBAT Rising)	—	0.95	—	
Missing Clock Detector Timeout	Time from last system clock rising edge to reset initiation	100	650	1000	µs
Minimum System Clock w/ Missing Clock Detector Enabled	System clock frequency which triggers a missing clock detector timeout	—	7	10	kHz
Reset Time Delay	Delay between release of any reset source and code execution at location 0x0000	—	10	—	µs
Minimum $\overline{RST}$ Low Time to Generate a System Reset		15	—	—	µs
V_{DD} Monitor Turn-on Time		—	300	—	ns
V_{DD} Monitor Supply Current		—	7	—	µA

10.5. External Memory Interface Operating Modes

The external data memory space can be configured in one of four operating modes, shown in Figure 10.3, based on the EMIF Mode bits in the EMI0CF register (SFR Definition 10.2). These modes are summarized below. Timing diagrams for the different modes can be found in Section “10.6. External Memory Interface Timing” on page 118.

10.5.1. Internal XRAM Only

When EMI0CF.[3:2] are set to 00, all MOVX instructions will target the internal XRAM space on the device. Memory accesses to addresses beyond the populated space will wrap, and will always target on-chip XRAM. As an example, if the entire address space is consecutively written and the data pointer is incremented after each write, the write pointer will always point to the first byte of on-chip XRAM after the last byte of on-chip XRAM has been written.

- 8-bit MOVX operations use the contents of EMI0CN to determine the high-byte of the effective address and R0 or R1 to determine the low-byte of the effective address.
- 16-bit MOVX operations use the contents of the 16-bit DPTR to determine the effective address.

10.5.2. Split Mode without Bank Select

When EMI0CF.[3:2] are set to 01, the XRAM memory map is split into two areas, on-chip space and off-chip space.

- Effective addresses below the on-chip XRAM boundary will access on-chip XRAM space.
- Effective addresses above the on-chip XRAM boundary will access off-chip space.
- 8-bit MOVX operations use the contents of EMI0CN to determine whether the memory access is on-chip or off-chip. However, in the “No Bank Select” mode, **an 8-bit MOVX operation will not drive the upper 4-bits A[11:8] of the Address Bus during an off-chip access.** This allows the user to manipulate the upper address bits at will by setting the Port state directly via the port latches. This behavior is in contrast with “Split Mode with Bank Select” described below. The lower 8-bits of the Address Bus A[7:0] are driven, determined by R0 or R1.
- 16-bit MOVX operations use the contents of DPTR to determine whether the memory access is on-chip or off-chip, and unlike 8-bit MOVX operations, the full 12-bits of the Address Bus A[11:0] are driven during the off-chip transaction.

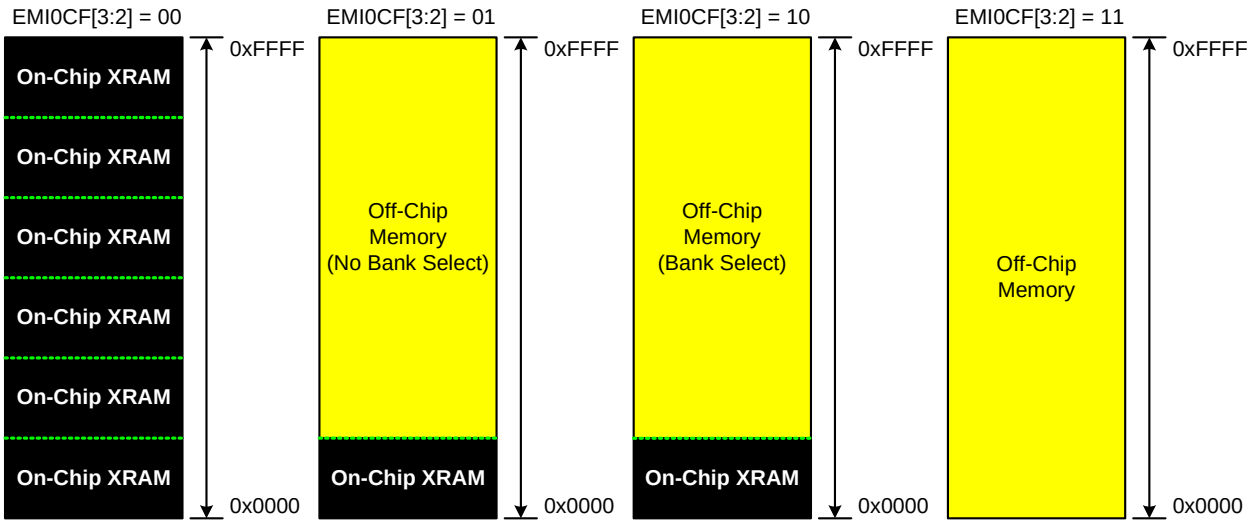

Figure 10.3. EMIF Operating Modes

Table 11.3. Special Function Registers (Continued)

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

Register	Address	SFR Page	Description	Page
TMOD	0x89	0x0	Timer/Counter Mode	285
TMR2CN	0xC8	0x0	Timer/Counter 2 Control	291
TMR2H	0xCD	0x0	Timer/Counter 2 High	293
TMR2L	0xCC	0x0	Timer/Counter 2 Low	293
TMR2RLH	0xCB	0x0	Timer/Counter 2 Reload High	292
TMR2RLL	0xCA	0x0	Timer/Counter 2 Reload Low	292
TMR3CN	0x91	0x0	Timer/Counter 3 Control	297
TMR3H	0x95	0x0	Timer/Counter 3 High	299
TMR3L	0x94	0x0	Timer/Counter 3 Low	299
TMR3RLH	0x93	0x0	Timer/Counter 3 Reload High	298
TMR3RLL	0x92	0x0	Timer/Counter 3 Reload Low	298
TOFFH	0x86	0xF	Temperature Offset High	85
TOFFL	0x85	0xF	Temperature Offset Low	85
VDM0CN	0xFF	0x0	VDD Monitor Control	183
XBR0	0xE1	0x0	Port I/O Crossbar Control 0	219
XBR1	0xE2	0x0	Port I/O Crossbar Control 1	220
XBR2	0xE3	0x0	Port I/O Crossbar Control 2	221

SFR Definition 13.1. PSCTL: Program Store R/W Control

Bit	7	6	5	4	3	2	1	0
Name						SFLE	PSEE	PSWE
Type	R	R	R	R	R	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Page =0x0; SFR Address = 0x8F

Bit	Name	Function
7:3	Unused	**Unused.** Read = 00000b, Write = don't care.
2	SFLE	**Scratchpad Flash Memory Access Enable.** When this bit is set, Flash MOVC reads and MOVX writes from user software are directed to the Scratchpad Flash sector. Flash accesses outside the address range 0x0000-0x03FF should not be attempted and may yield undefined results when SFLE is set to 1. 0: Flash access from user software directed to the Program/Data Flash sector. 1: Flash access from user software directed to the Scratchpad Sector.
1	PSEE	**Program Store Erase Enable.** Setting this bit (in combination with PSWE) allows an entire page of Flash program memory to be erased. If this bit is logic 1 and Flash writes are enabled (PSWE is logic 1), a write to Flash memory using the MOVX instruction will erase the entire page that contains the location addressed by the MOVX instruction. The value of the data byte written does not matter. 0: Flash program memory erasure disabled. 1: Flash program memory erasure enabled.
0	PSWE	**Program Store Write Enable.** Setting this bit allows writing a byte of data to the Flash program memory using the MOVX write instruction. The Flash location should be erased before writing data. 0: Writes to Flash program memory disabled. 1: Writes to Flash program memory enabled; the MOVX write instruction targets Flash memory.

17. Voltage Regulator (VREG0)

C8051F93x-C8051F92x devices include an internal voltage regulator (VREG0) to regulate the internal core supply to 1.8 V from a VDD/DC+ supply of 1.8 to 3.6 V. Electrical characteristics for the on-chip regulator are specified in the Electrical Specifications chapter.

The REG0CN register allows the Precision Oscillator Bias to be disabled, reducing supply current in all non-sleep power modes. This bias should only be disabled when the precision oscillator is not being used.

The internal regulator (VREG0) is disabled when the device enters sleep mode and remains enabled when the device enters suspend mode. See Section “14. Power Management” on page 156 for complete details about low power modes.

SFR Definition 17.1. REG0CN: Voltage Regulator Control

Bit	7	6	5	4	3	2	1	0
Name		Reserved	Reserved	OSCBIAS				Reserved
Type	R	R/W	R/W	R/W	R	R	R	R/W
Reset	0	0	0	1	0	0	0	0

SFR Page = 0x0; SFR Address = 0xC9

Bit	Name	Function
7	Unused	**Unused.** Read = 0b. Write = Don’t care.
6	Reserved	**Reserved.** Read = 0b. Must Write 0b.
5	Reserved	**Reserved.** Read = 0b. Must Write 0b.
4	OSCBIAS	**Precision Oscillator Bias.** When set to 1, the bias used by the precision oscillator is forced on. If the precision oscillator is not being used, this bit may be cleared to 0 to save approximately 80 µA of supply current in all non-Sleep power modes. If disabled then re-enabled, the precision oscillator bias requires 4 µs of settling time.
3:1	Unused	**Unused.** Read = 000b. Write = Don’t care.
0	Reserved	**Reserved.** Read = 0b. Must Write 0b.

17.1. Voltage Regulator Electrical Specifications

See Table 4.15 on page 64 for detailed Voltage Regulator Electrical Specifications.

19. Clocking Sources

C8051F93x-C8051F92x devices include a programmable precision internal oscillator, an external oscillator drive circuit, a low power internal oscillator, and a SmaRTClock real time clock oscillator. The precision internal oscillator can be enabled/disabled and calibrated using the OSCICN and OSCICL registers, as shown in Figure 19.1. The external oscillator can be configured using the OSCXCN register. The low power internal oscillator is automatically enabled and disabled when selected and deselected as a clock source. SmaRTClock operation is described in the SmaRTClock oscillator chapter.

The system clock (SYSCLK) can be derived from the precision internal oscillator, external oscillator, low power internal oscillator, or SmaRTClock oscillator. The global clock divider can generate a system clock that is 1, 2, 4, 8, 16, 32, 64, or 128 times slower that the selected input clock source. Oscillator electrical specifications can be found in the Electrical Specifications Chapter.

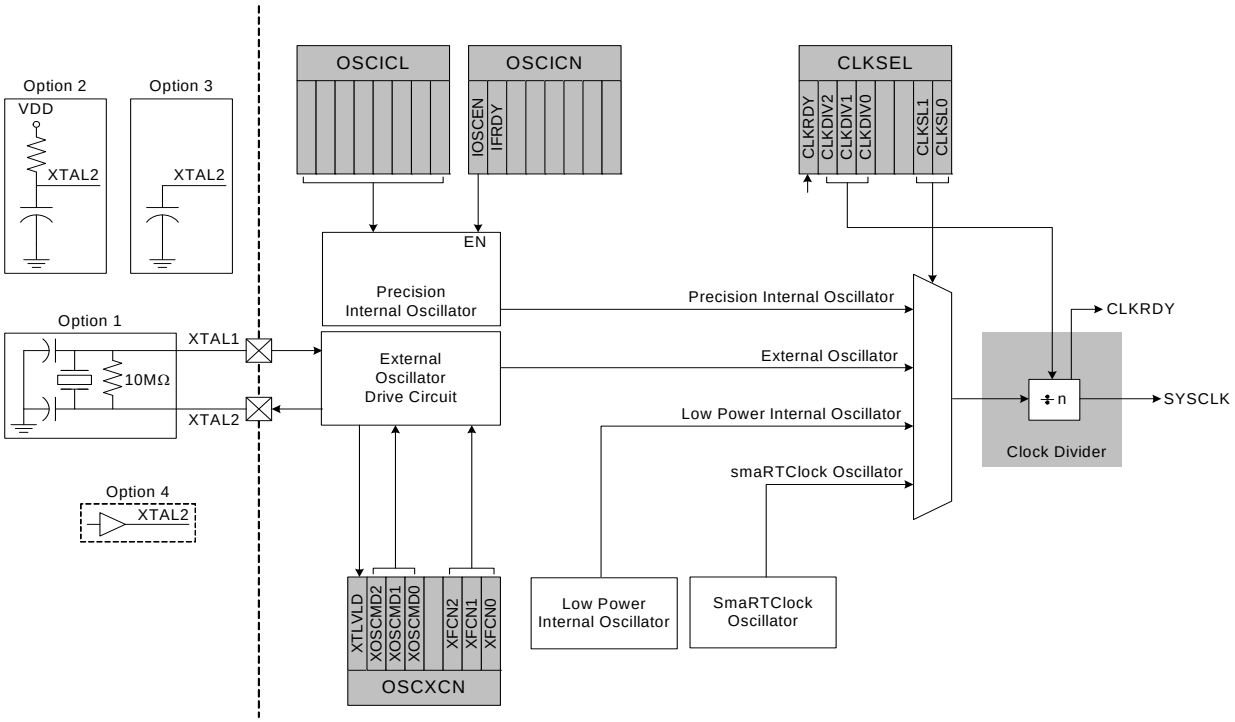

Figure 19.1. Clocking Sources Block Diagram

The proper way of changing the system clock when both the clock source and the clock divide value are being changed is as follows:

If switching from a fast “undivided” clock to a slower “undivided” clock:

a. Change the clock divide value.

b. Poll for CLKRDY > 1.

c. Change the clock source.

If switching from a slow “undivided” clock to a faster “undivided” clock:

a. Change the clock source.

b. Change the clock divide value.

c. Poll for CLKRDY > 1.

SFR Definition 21.2. XBR1: Port I/O Crossbar Register 1

Bit	7	6	5	4	3	2	1	0
Name		SPI1E	T1E	T0E	ECIE	PCA0ME[2:0]		
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Page = 0x0; SFR Address = 0xE2

Bit	Name	Function
7	Unused	**Unused.** Read = 0b; Write = Don't Care.
6	SPI1E	**SPI1 I/O Enable.** 0: SPI0 I/O unavailable at Port pin. 1: SCK (for SPI1) routed to P1.0. MISO (for SPI1) routed to P1.1. MOSI (for SPI1) routed to P1.2. NSS (for SPI1) routed to P1.3 only if SPI1 is configured to 4-wire mode.
5	T1E	**Timer1 Input Enable.** 0: T1 input unavailable at Port pin. 1: T1 input routed to Port pin.
4	T0E	**Timer0 Input Enable.** 0: T0 input unavailable at Port pin. 1: T0 input routed to Port pin.
3	ECIE	**PCA0 External Counter Input (ECI) Enable.** 0: PCA0 external counter input unavailable at Port pin. 1: PCA0 external counter input routed to Port pin.
2:0	PCA0ME	**PCA0 Module I/O Enable.** 000: All PCA0 I/O unavailable at Port pin. 001: CEX0 routed to Port pin. 010: CEX0, CEX1 routed to Port pins. 011: CEX0, CEX1, CEX2 routed to Port pins. 100: CEX0, CEX1, CEX2 CEX3 routed to Port pins. 101: CEX0, CEX1, CEX2, CEX3, CEX4 routed to Port pins. 110: CEX0, CEX1, CEX2, CEX3, CEX4, CEX5 routed to Port pins. 111: Reserved.
Note: SPI1 can be assigned either 3 or 4 Port I/O pins.		

21.4. Port Match

Port match functionality allows system events to be triggered by a logic value change on P0 or P1. A software controlled value stored in the PnMAT registers specifies the expected or normal logic values of P0 and P1. A Port mismatch event occurs if the logic levels of the Port's input pins no longer match the software controlled value. This allows Software to be notified if a certain change or pattern occurs on P0 or P1 input pins regardless of the XBRn settings. **Note: On C8051F931/21 devices, Port Match is not available on P1.6 or P1.7.**

The PnMASK registers can be used to individually select which P0 and P1 pins should be compared against the PnMAT registers. A Port mismatch event is generated if (P0 & P0MASK) does not equal (PnMAT & P0MASK) or if (P1 & P1MASK) does not equal (PnMAT & P1MASK).

A Port mismatch event may be used to generate an interrupt or wake the device from a low power mode. See Section "12. Interrupt Handler" on page 133 and Section "14. Power Management" on page 156 for more details on interrupt and wake-up sources.

SFR Definition 21.4. P0MASK: Port0 Mask Register

Bit	7	6	5	4	3	2	1	0
Name	P0MASK[7:0]							
Type	R/W							
Reset	0	0	0	0	0	0	0	0

SFR Page= 0x0; SFR Address = 0xC7

Bit	Name	Function
7:0	P0MASK[7:0]	**Port0 Mask Value.** Selects the P0 pins to be compared with the corresponding bits in P0MAT. 0: P0.n pin pad logic value is ignored and cannot cause a Port Mismatch event. 1: P0.n pin pad logic value is compared to P0MAT.n.

SFR Definition 21.5. P0MAT: Port0 Match Register

Bit	7	6	5	4	3	2	1	0
Name	P0MAT[7:0]							
Type	R/W							
Reset	1	1	1	1	1	1	1	1

SFR Page= 0x0; SFR Address = 0xD7

Bit	Name	Function
7:0	P0MAT[7:0]	**Port 0 Match Value.** Match comparison value used on Port 0 for bits in P0MASK which are set to 1. 0: P0.n pin logic value is compared with logic LOW. 1: P0.n pin logic value is compared with logic HIGH.

SFR Definition 21.12. P0DRV: Port0 Drive Strength

Bit	7	6	5	4	3	2	1	0
Name	P0DRV[7:0]							
Type	R/W							
Reset	0	0	0	0	0	0	0	0

SFR Page = 0xF; SFR Address = 0xA4

Bit	Name	Function
7:0	P0DRV[7:0]	**Drive Strength Configuration Bits for P0.7–P0.0 (respectively).** Configures digital I/O Port cells to high or low output drive strength. 0: Corresponding P0.n Output has low output drive strength. 1: Corresponding P0.n Output has high output drive strength.

SFR Definition 21.15. P1MDIN: Port1 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 Page = 0x0; SFR Address = 0xF2

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 pullup and digital receiver disabled. The digital driver is not explicitly disabled. 0: Corresponding P1.n pin is configured for analog mode. 1: Corresponding P1.n pin is not configured for analog mode.
Note: Pin P1.7 is only available in 32-pin devices.		

SFR Definition 21.16. P1MDOUT: Port1 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 Page = 0x0; SFR Address = 0xA5

Bit	Name	Function
7:0	P1MDOUT[7:0]	**Output Configuration Bits for P1.7–P1.0 (respectively).** These bits control the digital driver even when 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.
Note: Pin P1.7 is only available in 32-pin devices.		

22.1. Supporting Documents

It is assumed the reader is familiar with or has access to the following supporting documents:

1. The I^2C-Bus and How to Use It (including specifications), Philips Semiconductor.
2. The I^2C-Bus Specification—Version 2.0, Philips Semiconductor.
3. System Management Bus Specification—Version 1.1, SBS Implementers Forum.

22.2. SMBus Configuration

Figure 22.2 shows a typical SMBus configuration. The SMBus specification allows any recessive voltage between 3.0 V and 5.0 V; different devices on the bus may operate at different voltage levels. The bi-directional SCL (serial clock) and SDA (serial data) lines must be connected to a positive power supply voltage through a pullup resistor or similar circuit. Every device connected to the bus must have an open-drain or open-collector output for both the SCL and SDA lines, so that both are pulled high (recessive state) when the bus is free. The maximum number of devices on the bus is limited only by the requirement that the rise and fall times on the bus not exceed 300 ns and 1000 ns, respectively.

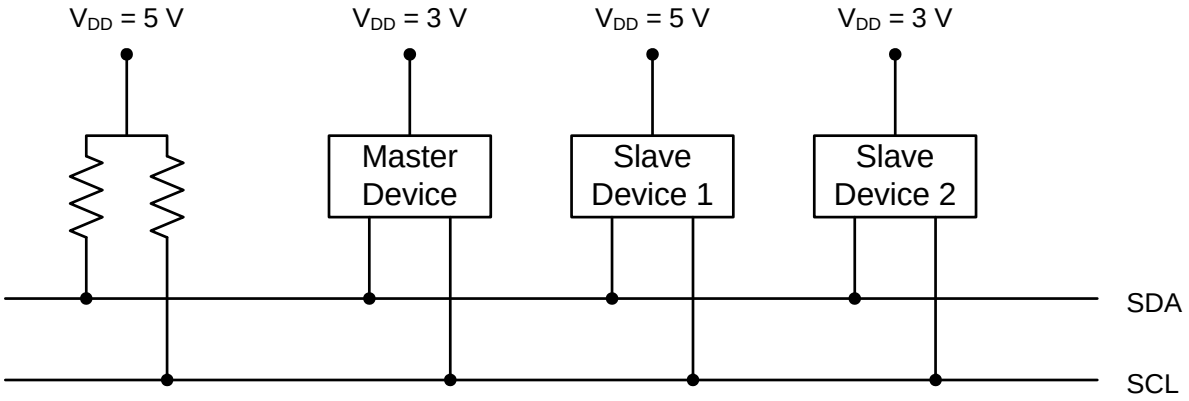

Figure 22.2. Typical SMBus Configuration

22.4.1. SMBus Configuration Register

The SMBus Configuration register (SMB0CF) is used to enable the SMBus Master and/or Slave modes, select the SMBus clock source, and select the SMBus timing and timeout options. When the ENSMB bit is set, the SMBus is enabled for all master and slave events. Slave events may be disabled by setting the INH bit. With slave events inhibited, the SMBus interface will still monitor the SCL and SDA pins; however, the interface will NACK all received addresses and will not generate any slave interrupts. When the INH bit is set, all slave events will be inhibited following the next START (interrupts will continue for the duration of the current transfer).

Table 22.1. SMBus Clock Source Selection

SMBCS1	SMBCS0	SMBus Clock Source
0	0	Timer 0 Overflow
0	1	Timer 1 Overflow
1	0	Timer 2 High Byte Overflow
1	1	Timer 2 Low Byte Overflow

The SMBCS1–0 bits select the SMBus clock source, which is used only when operating as a master or when the Free Timeout detection is enabled. When operating as a master, overflows from the selected source determine the absolute minimum SCL low and high times as defined in Equation 22.1. The selected clock source may be shared by other peripherals so long as the timer is left running at all times. For example, Timer 1 overflows may generate the SMBus and UART baud rates simultaneously. Timer configuration is covered in Section “25. Timers” on page 278.

$$T_{HighMin} = T_{LowMin} = \frac{1}{f_{ClockSourceOverflow}}$$

Equation 22.1. Minimum SCL High and Low Times

The selected clock source should be configured to establish the minimum SCL High and Low times as per Equation 22.1. When the interface is operating as a master (and SCL is not driven or extended by any other devices on the bus), the typical SMBus bit rate is approximated by Equation 22.2.

$$BitRate = \frac{f_{ClockSourceOverflow}}{3}$$

Equation 22.2. Typical SMBus Bit Rate

Figure 22.4 shows the typical SCL generation described by Equation 22.2. Notice that T_{HIGH} is typically twice as large as T_{LOW}. The actual SCL output may vary due to other devices on the bus (SCL may be extended low by slower slave devices, or driven low by contending master devices). The bit rate when operating as a master will never exceed the limits defined by equation Equation 22.1.

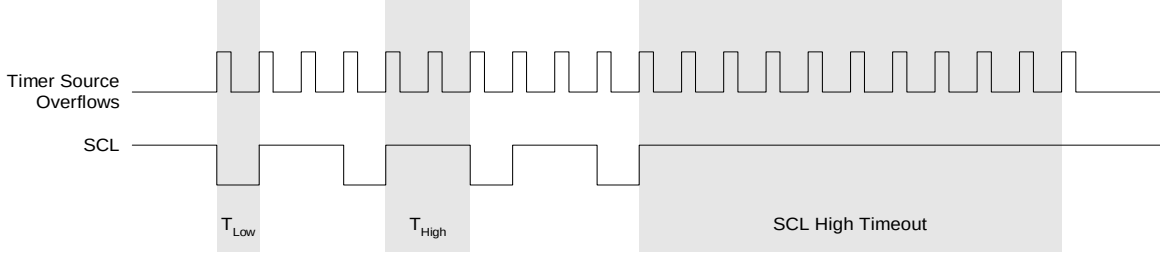

Figure 22.4. Typical SMBus SCL Generation

22.4.4. Data Register

The SMBus Data register SMB0DAT holds a byte of serial data to be transmitted or one that has just been received. Software may safely read or write to the data register when the SI flag is set. Software should not attempt to access the SMB0DAT register when the SMBus is enabled and the SI flag is cleared to logic 0, as the interface may be in the process of shifting a byte of data into or out of the register.

Data in SMB0DAT is always shifted out MSB first. After a byte has been received, the first bit of received data is located at the MSB of SMB0DAT. While data is being shifted out, data on the bus is simultaneously being shifted in. SMB0DAT always contains the last data byte present on the bus. In the event of lost arbitration, the transition from master transmitter to slave receiver is made with the correct data or address in SMB0DAT.

SFR Definition 22.5. SMB0DAT: SMBus Data

Bit	7	6	5	4	3	2	1	0
Name	SMB0DAT[7:0]							
Type	R/W							
Reset	0	0	0	0	0	0	0	0

SFR Page = 0x0; SFR Address = 0xC2

Bit	Name	Function
7:0	SMB0DAT[7:0]	**SMBus Data.** The SMB0DAT register contains a byte of data to be transmitted on the SMBus serial interface or a byte that has just been received on the SMBus serial interface. The CPU can read from or write to this register whenever the SI serial interrupt flag (SMB0CN.0) is set to logic 1. The serial data in the register remains stable as long as the SI flag is set. When the SI flag is not set, the system may be in the process of shifting data in/out and the CPU should not attempt to access this register.

22.5.3. Write Sequence (Slave)

During a write sequence, an SMBus master writes data to a slave device. The slave in this transfer will be a receiver during the address byte, and a receiver during all data bytes. When slave events are enabled (INH = 0), the interface enters Slave Receiver Mode when a START followed by a slave address and direction bit (WRITE in this case) is received. If hardware ACK generation is disabled, upon entering Slave Receiver Mode, an interrupt is generated and the ACKRQ bit is set. The software must respond to the received slave address with an ACK, or ignore the received slave address with a NACK. If hardware ACK generation is enabled, the hardware will apply the ACK for a slave address which matches the criteria set up by SMB0ADR and SMB0ADM. The interrupt will occur after the ACK cycle.

If the received slave address is ignored (by software or hardware), slave interrupts will be inhibited until the next START is detected. If the received slave address is acknowledged, zero or more data bytes are received.

If hardware ACK generation is disabled, the ACKRQ is set to 1 and an interrupt is generated after each received byte. Software must write the ACK bit at that time to ACK or NACK the received byte.

With hardware ACK generation enabled, the SMBus hardware will automatically generate the ACK/NACK, and then post the interrupt. **The appropriate ACK or NACK value should be set up by the software prior to receiving the byte when hardware ACK generation is enabled.**

The interface exits Slave Receiver Mode after receiving a STOP. Note that the interface will switch to Slave Transmitter Mode if SMB0DAT is written while an active Slave Receiver. Figure 22.7 shows a typical slave write sequence. Two received data bytes are shown, though any number of bytes may be received. Notice that the ‘data byte transferred’ interrupts occur at different places in the sequence, depending on whether hardware ACK generation is enabled. The interrupt occurs **before** the ACK with hardware ACK generation disabled, and **after** the ACK when hardware ACK generation is enabled.

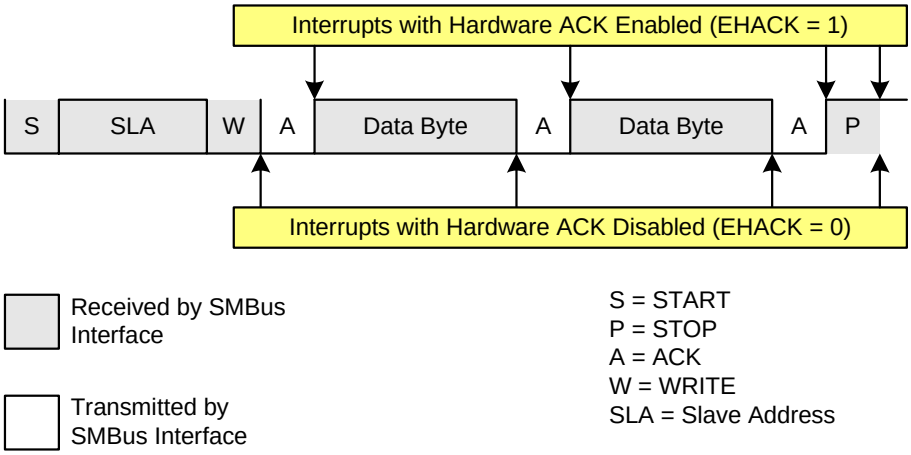

Figure 22.7. Typical Slave Write Sequence

Table 22.5. SMBus Status Decoding With Hardware ACK Generation Disabled (EHACK = 0)

Mode	Values Read				Current SMbus State	Typical Response Options	Values to Write			Next Status Vector Expected
	Status Vector	ACKRQ	ARBLOST	ACK			STA	STO	ACK	
Master Transmitter	1110	0	0	X	A master START was generated.	Load slave address + R/W into SMB0DAT.	0	0	X	1100
	1100	0	0	0	A master data or address byte was transmitted; NACK received.	Set STA to restart transfer.	1	0	X	1110
						Abort transfer.	0	1	X	-
		0	0	1	A master data or address byte was transmitted; ACK received.	Load next data byte into SMB0DAT.	0	0	X	1100
						End transfer with STOP.	0	1	X	-
						End transfer with STOP and start another transfer.	1	1	X	-
						Send repeated START.	1	0	X	1110
						Switch to Master Receiver Mode (clear SI without writing new data to SMB0DAT).	0	0	X	1000
Master Receiver	1000	1	0	X	A master data byte was received; ACK requested.	Acknowledge received byte; Read SMB0DAT.	0	0	1	1000
						Send NACK to indicate last byte, and send STOP.	0	1	0	-
						Send NACK to indicate last byte, and send STOP followed by START.	1	1	0	1110
						Send ACK followed by repeated START.	1	0	1	1110
						Send NACK to indicate last byte, and send repeated START.	1	0	0	1110
						Send ACK and switch to Master Transmitter Mode (write to SMB0DAT before clearing SI).	0	0	1	1100
						Send NACK and switch to Master Transmitter Mode (write to SMB0DAT before clearing SI).	0	0	0	1100

Table 22.6. SMBus Status Decoding With Hardware ACK Generation Enabled (EHACK = 1)

Mode	Values Read				Current SMbus State	Typical Response Options	Values to Write			Next Status Vector Expected
	Status Vector	ACKRQ	ARBLOST	ACK			STA	STO	ACK	
Master Transmitter	1110	0	0	X	A master START was generated.	Load slave address + R/W into SMB0DAT.	0	0	X	1100
	1100	0	0	0	A master data or address byte was transmitted; NACK received.	Set STA to restart transfer.	1	0	X	1110
						Abort transfer.	0	1	X	-
		0	0	1	A master data or address byte was transmitted; ACK received.	Load next data byte into SMB0DAT.	0	0	X	1100
						End transfer with STOP.	0	1	X	-
						End transfer with STOP and start another transfer.	1	1	X	-
						Send repeated START.	1	0	X	1110
						Switch to Master Receiver Mode (clear SI without writing new data to SMB0DAT). Set ACK for initial data byte.	0	0	1	1000
Master Receiver	1000	0	0	1	A master data byte was received; ACK sent.	Set ACK for next data byte; Read SMB0DAT.	0	0	1	1000
						Set NACK to indicate next data byte as the last data byte; Read SMB0DAT.	0	0	0	1000
						Initiate repeated START.	1	0	0	1110
						Switch to Master Transmitter Mode (write to SMB0DAT before clearing SI).	0	0	X	1100
		0	0	0	A master data byte was received; NACK sent (last byte).	Read SMB0DAT; send STOP.	0	1	0	-
						Read SMB0DAT; Send STOP followed by START.	1	1	0	1110
						Initiate repeated START.	1	0	0	1110
						Switch to Master Transmitter Mode (write to SMB0DAT before clearing SI).	0	0	X	1100

SFR Definition 24.4. SPInDAT: SPI Data

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

SFR Addresses: SPI0DAT = 0xA3, SPI1DAT = 0x86
SFR Pages: SPI0DAT = 0x0, SPI1DAT = 0x0

Bit	Name	Function
7:0	SPInDAT	**SPIn Transmit and Receive Data.** The SPInDAT register is used to transmit and receive SPIn data. Writing data to SPInDAT places the data into the transmit buffer and initiates a transfer when in Master Mode. A read of SPInDAT returns the contents of the receive buffer.

26.3.1. Edge-triggered Capture Mode

In this mode, a valid transition on the CEXn pin causes the PCA to capture the value of the PCA counter/timer and load it into the corresponding module's 16-bit capture/compare register (PCA0CPLn and PCA0CPHn). The CAPPn and CAPNn bits in the PCA0CPMn register are used to select the type of transition that triggers the capture: low-to-high transition (positive edge), high-to-low transition (negative edge), or either transition (positive or negative edge). When a capture occurs, the Capture/Compare Flag (CCFn) in PCA0CN is set to logic 1. An interrupt request is generated if the CCFn interrupt for that module is enabled. The CCFn bit is not automatically cleared by hardware when the CPU vectors to the interrupt service routine, and must be cleared by software. If both CAPPn and CAPNn bits are set to logic 1, then the state of the Port pin associated with CEXn can be read directly to determine whether a rising-edge or falling-edge caused the capture.

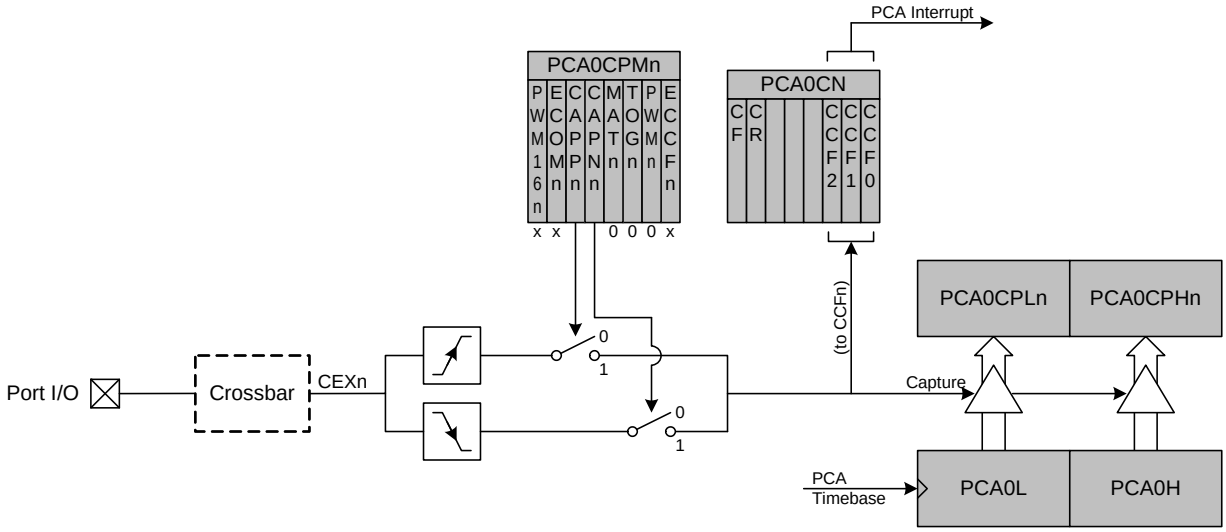

Figure 26.4. PCA Capture Mode Diagram

Note: The CEXn input signal must remain high or low for at least 2 system clock cycles to be recognized by the hardware.