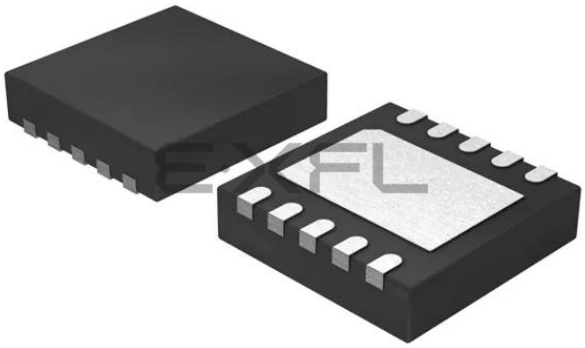

Welcome to E-XFL.COM

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	25MHz
Connectivity	SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, Temp Sensor, WDT
Number of I/O	6
Program Memory Size	8KB (8K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	256 x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 5.25V
Data Converters	A/D 6x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	10-VFDFN Exposed Pad
Supplier Device Package	10-DFN (3x3)
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051f521-c-imr

Table 3.1. Pin Definitions for the C8051F52x and C8051F52xA (DFN 10) (Continued)

Name	Pin Numbers		Type	Description
	‘F52xA ‘F52x-C	‘F52x		
P0.3/TX*/	—	8	D I/O or A In	Port 0.3. See Port I/O Section for a complete description.
XTAL2			D I/O	External Clock Output. For an external crystal or resonator, this pin is the excitation driver. This pin is the external clock input for CMOS, capacitor, or RC oscillator configurations. See Section “14. Oscillators” on page 135.
P0.2	9	9	D I/O or	Port 0.2. See Port I/O Section for a complete description.
XTAL1			A In	External Clock Input. This pin is the external oscillator return for a crystal or resonator. Section “14. Oscillators” on page 135.
P0.1/	10	10	D I/O or A In	Port 0.1. See Port I/O Section for a complete description.
C2D			D I/O	Bi-directional data signal for the C2 Debug Interface
Note: Please refer to Section “20. Device Specific Behavior” on page 210.				

Table 3.9. QFN-20 Landing Diagram Dimensions

Symbol	Min	Max
C1	3.90	4.00
C2	3.90	4.00
E	0.50 BSC.	
X1	0.20	0.30
X2	2.75	2.85
Y1	0.65	0.75
Y2	2.75	2.85

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 60 µm 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.
7. A 2x2 array of 1.10 x 1.10 mm openings on 1.30 mm pitch should be used for the center ground pad.

Card Assembly

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

SFR Definition 7.2. CPT0MX: Comparator0 MUX Selection

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
CMX0N3	CMX0N2	CMX0N1	CMX0N0	CMX0P3	CMX0P2	CMX0P1	CMX0P0	01110111
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0x9F

Bits7–4: **CMX0N3–CMX0N0**: Comparator0 Negative Input MUX Select.
These bits select which Port pin is used as the Comparator0 negative input.

CMX0N3	CMX0N2	CMX0N1	CMX0N0	Negative Input
0	0	0	0	P0.1
0	0	0	1	P0.3
0	0	1	0	P0.5
0	0	1	1	P0.7*
0	1	0	0	P1.1*
0	1	0	1	P1.3*
0	1	1	0	P1.5*
0	1	1	1	P1.7*

***Note:** Available only on the C8051F53x/53xA devices

Bits1–0: **CMX0P3–CMX0P0**: Comparator0 Positive Input MUX Select.
These bits select which Port pin is used as the Comparator0 positive input.

CMX0P3	CMX0P2	CMX0P1	CMX0P0	Positive Input
0	0	0	0	P0.0
0	0	0	1	P0.2
0	0	1	0	P0.4
0	0	1	1	P0.6*
0	1	0	0	P1.0*
0	1	0	1	P1.2*
0	1	1	0	P1.4*
0	1	1	1	P1.6*

***Note:** Available only on the C8051F53x/53xA devices.

SFR Definition 8.4. PSW: Program Status Word

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	Reset Value
CY	AC	F0	RS1	RS0	OV	F1	PARITY	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	Bit Addressable

SFR Address: 0xD0

Bit7: **CY**: Carry Flag.
This bit is set when the last arithmetic operation resulted in a carry (addition) or a borrow (subtraction). It is cleared to 0 by all other arithmetic operations.

Bit6: **AC**: Auxiliary Carry Flag
This bit is set when the last arithmetic operation resulted in a carry into (addition) or a borrow from (subtraction) the high order nibble. It is cleared to 0 by all other arithmetic operations.

Bit5: **F0**: User Flag 0.
This is a bit-addressable, general purpose flag for use under software control.

Bits4–3: **RS1–RS0**: Register Bank Select.
These bits select which register bank is used during register accesses.

RS1	RS0	Register Bank	Address
0	0	0	0x00–0x07
0	1	1	0x08–0x0F
1	0	2	0x10–0x17
1	1	3	0x18–0x1F

Bit2: **OV**: Overflow Flag.
This bit is set to 1 under the following circumstances:
- An ADD, ADDC, or SUBB instruction causes a sign-change overflow.
- A MUL instruction results in an overflow (result is greater than 255).
- A DIV instruction causes a divide-by-zero condition.

The OV bit is cleared to 0 by the ADD, ADDC, SUBB, MUL, and DIV instructions in all other cases.

Bit1: **F1**: User Flag 1.
This is a bit-addressable, general purpose flag for use under software control.

Bit0: **PARITY**: Parity Flag.
This bit is set to 1 if the sum of the eight bits in the accumulator is odd and cleared if the sum is even.

9. Memory Organization and SFRs

The memory organization of the C8051F52x/F52xA/F53x/F53xA is similar to that of a standard 8051. There are two separate memory spaces: program memory and data memory. Program and data memory share the same address space but are accessed via different instruction types. The memory map is shown in Figure 9.1.

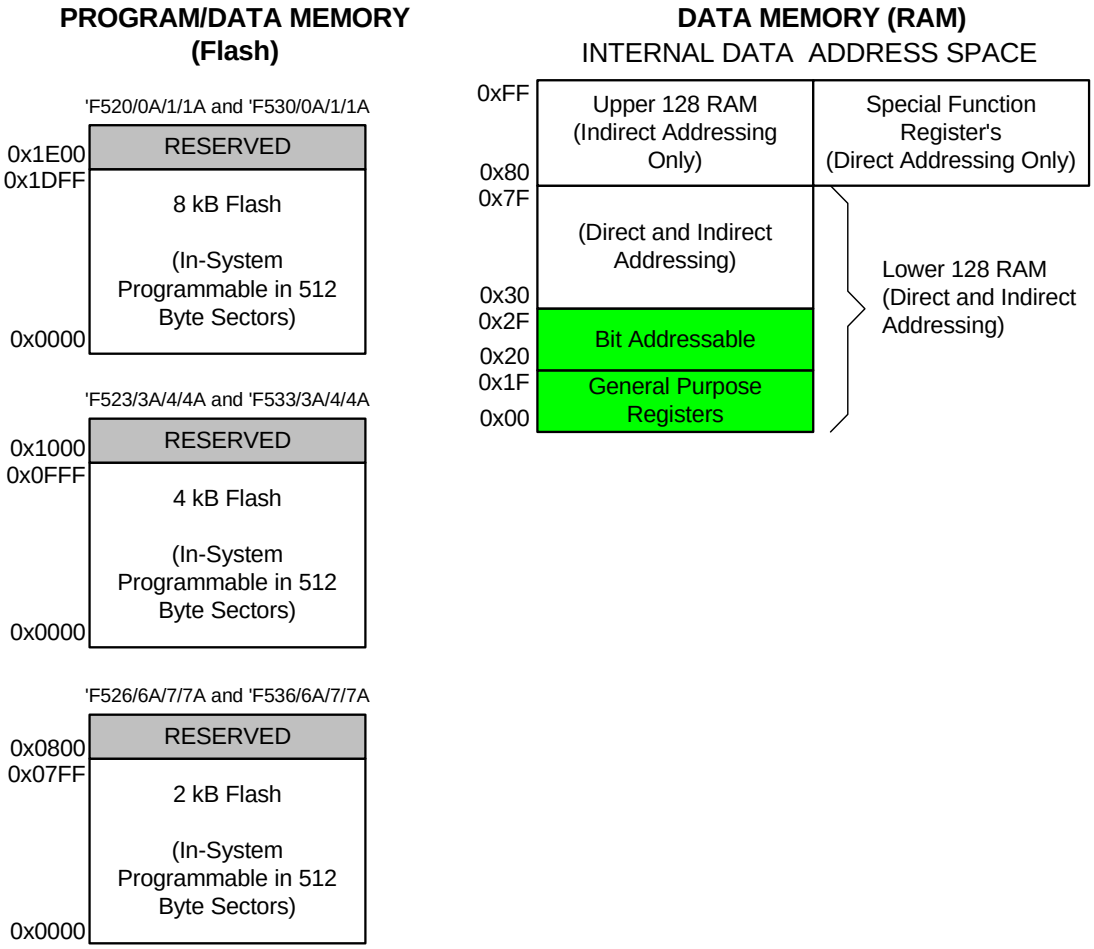

Figure 9.1. Memory Map

9.1. Program Memory

The CIP-51 core has a 64 kB program memory space. The C8051F520/0A/1/1A and C8051F530/0A/1/1A implement 8 kB of this program memory space as in-system, re-programmable Flash memory, organized in a contiguous block from addresses 0x0000 to 0x1FFF. Addresses above 0x1DFF are reserved on the 8 kB devices. The C8051F523/3A/4/4A and C8051F533/3A/4/4A implement 4 kB of Flash from addresses 0x0000 to 0x0FFF. The C8051F526/6A/7/7A and C8051F536/6A/7/7A implement 2 kB of Flash from addresses 0x0000 to 0x07FF.

Program memory is normally assumed to be read-only. However, the C8051F52x/F52xA/F53x/F53xA can write to program memory by setting the Program Store Write Enable bit (PSCTL.0) and using the MOVX write instruction. This feature provides a mechanism for updates to program code and use of the program memory space for non-volatile data storage. Refer to Section “12. Flash Memory” on page 113 for further details.

SFR Definition 13.9. P1: Port1

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
P1.7	P1.6	P1.5	P1.4	P1.3	P1.2	P1.1	P1.0	11111111
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	Bit Addressable
							SFR Address:	0x90

Bits7–0: **P1.[7:0]**
Write - Output appears on I/O pins per Crossbar Registers.
0: Logic Low Output.
1: Logic High Output (high impedance if corresponding P1MDOUT.n bit = 0).
Read - Always reads 0 if selected as analog input in register P1MDIN. Directly reads Port pin when configured as digital input.
0: P1.n pin is logic low.
1: P1.n pin is logic high.

SFR Definition 13.10. P1MDIN: Port1 Input Mode

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								11111111
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
							SFR Address:	0xF2

Bits7–0: Analog Input Configuration Bits for P1.7–P1.0 (respectively).
Port pins configured as analog inputs have their weak pullup, digital driver, and digital receiver disabled.
0: Corresponding P1.n pin is configured as an analog input.
1: Corresponding P1.n pin is not configured as an analog input.

SFR Definition 13.11. P1MDOUT: Port1 Output Mode

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

SFR Address: 0xA5

Bits7–0: Output Configuration Bits for P1.7–P1.0 (respectively): ignored if 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: When SDA and SCL appear on any of the Port I/O, each are open-drain regardless of the value of P0MDOUT.

SFR Definition 13.12. P1SKIP: Port1 Skip

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

SFR Address: 0xD5

Bits7–0: **P1SKIP[7:0]**: Port1 Crossbar Skip Enable Bits.
These bits select Port pins to be skipped by the Crossbar Decoder. Port pins used as analog inputs (for ADC or Comparator) or used as special functions (V_{REF} input, external oscillator circuit, CNVSTR input) should be skipped by the Crossbar.
0: Corresponding P1.n pin is not skipped by the Crossbar.
1: Corresponding P1.n pin is skipped by the Crossbar.

16.2. SPI0 Master Mode Operation

A SPI master device initiates all data transfers on a SPI bus. SPI0 is placed in master mode by setting the Master Enable flag (MSTEN, SPI0CN.6). Writing a byte of data to the SPI0 data register (SPI0DAT) when in master mode writes to the transmit buffer. If the SPI shift register is empty, the byte in the transmit buffer is moved to the shift register, and a data transfer begins. The SPI0 master immediately shifts out the data serially on the MOSI line while providing the serial clock on SCK. The SPIF (SPI0CN.7) flag is set to logic 1 at the end of the transfer. If interrupts are enabled, an interrupt request is generated when the SPIF flag is set. While the SPI0 master transfers data to a slave on the MOSI line, the addressed SPI slave device simultaneously transfers data to the SPI master on the MISO line in a full-duplex operation. Therefore, the SPIF flag serves as both a transmit-complete and receive-data-ready flag. The data byte received from the slave is transferred MSB-first into the master's shift register. When a byte is fully shifted into the register, it is moved to the receive buffer where it can be read by the processor by reading SPI0DAT.

When configured as a master, SPI0 can operate in one of three different modes: multi-master mode, 3-wire single-master mode, and 4-wire single-master mode. The default, multi-master mode is active when NSSMD1 (SPI0CN.3) = 0 and NSSMD0 (SPI0CN.2) = 1. In this mode, NSS is an input to the device, and is used to disable the master SPI0 when another master is accessing the bus. When NSS is pulled low in this mode, MSTEN (SPI0CN.6) and SPIEN (SPI0CN.0) are set to 0 to disable the SPI master device, and a Mode Fault is generated (MODF, SPI0CN.5 = 1). Mode Fault will generate an interrupt if enabled. SPI0 must be manually re-enabled in software under these circumstances. In multi-master systems, devices will typically default to being slave devices while they are not acting as the system master device. In multi-master mode, slave devices can be addressed individually (if needed) using general-purpose I/O pins. Figure 16.2 shows a connection diagram between two master devices in multiple-master mode.

3-wire single-master mode is active when NSSMD1 (SPI0CN.3) = 0 and NSSMD0 (SPI0CN.2) = 0. In this mode, NSS is not used and is not mapped to an external port pin through the crossbar. Any slave devices that must be addressed in this mode should be selected using general-purpose I/O pins. Figure 16.3 shows a connection diagram between a master device in 3-wire master mode and a slave device.

4-wire single-master mode is active when NSSMD1 (SPI0CN.3) = 1. In this mode, NSS is configured as an output pin and can be used as a slave-select signal for a single SPI device. In this mode, the output value of NSS is controlled (in software) with the bit NSSMD0 (SPI0CN.2). Additional slave devices can be addressed using general-purpose I/O pins. Figure 16.4 shows a connection diagram for a master device in 4-wire master mode and two slave devices.

SFR Definition 16.4. SPI0DAT: SPI0 Data

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

SFR Address: 0xA3

Bits7–0: **SPI0DAT**: SPI0 Transmit and Receive Data.
The SPI0DAT register is used to transmit and receive SPI0 data. Writing data to SPI0DAT places the data into the transmit buffer and initiates a transfer when in Master Mode. A read of SPI0DAT returns the contents of the receive buffer.

17.4. LIN Slave Mode Operation

When the device is configured for slave mode operation, it must wait for a command from a master node. Access from the firmware to data buffer and ID registers of the LIN peripheral is only possible when a data request is pending (DTREQ bit (LIN0ST.4) is 1) and also when the LIN bus is not active (ACTIVE bit (LIN0ST.7) is set to 0).

The LIN peripheral in slave mode detects the header of the message frame sent by the LIN master. If slave synchronization is enabled (autobaud), the slave synchronizes its internal bit time to the master bit time.

The LIN peripheral configured for slave mode will generated an interrupt in one of three situations:

1. After the reception of the IDENTIFIER FIELD.
2. When an error is detected.
3. When the message transfer is completed.

The application should perform the following steps when an interrupt is detected:

1. Check the status of the DTREQ bit (LIN0ST.4). This bit is set when the IDENTIFIER FIELD has been received.
2. If DTREQ (LIN0ST.4) is set, read the identifier from LIN0ID and process it. If DTREQ (LIN0ST.4) is not set, continue to step 7.
3. Set the TXRX bit (LIN0CTRL.5) to 1 if the current frame is a transmit operation for the slave and set to 0 if the current frame is a receive operation for the slave.
4. Load the data length into LIN0SIZE.
5. For a slave transmit operation, load the data to transmit into the data buffer.
6. Set the DTACK bit (LIN0CTRL.4). Continue to step 10.
7. If DTREQ (LIN0ST.4) is not set, check the DONE bit (LIN0ST.0). The transmission was successful if the DONE bit is set.
8. If the transmission was successful and the current frame was a receive operation for the slave, load the received data bytes from the data buffer.
9. If the transmission was not successful, check LIN0ERR to determine the nature of the error. Further error handling has to be done by the application.
10. Set the RSTINT (LIN0CTRL.3) and RSTERR bits (LIN0CTRL.2) to reset the interrupt request and the error flags.

In addition to these steps, the application should be aware of the following:

1. If the current frame is a transmit operation for the slave, steps 1 through 5 must be completed during the IN-FRAME RESPONSE SPACE. If it is not completed in time, a timeout will be detected by the master.
2. If the current frame is a receive operation for the slave, steps 1 through 5 have to be finished until the reception of the first byte after the IDENTIFIER FIELD. Otherwise, the internal receive buffer of the LIN peripheral will be overwritten and a timeout error will be detected in the LIN peripheral.
3. The LIN module does not directly support LIN Version 1.3 Extended Frames. If the application detects an unknown identifier (e.g. extended identifier), it has to write a 1 to the STOP bit (LIN0CTRL.7) instead of setting the DTACK (LIN0CTRL.4) bit. At that time, steps 2 through 5 can then be skipped. In this situation, the LIN peripheral stops the processing of the LIN communication until the next SYNC BREAK is received.
4. Changing the configuration of the checksum during a transaction will cause the interface to reset and the transaction to be lost. To prevent this, the checksum should not be configured while a transaction is

SFR Definition 17.13. LIN0ST: LIN0 STATUS Register

R	R	R	R	R/W	R	R	R	Reset Value
ACTIVE	IDLTOUT	ABORT	DTREQ	LININT	ERROR	WAKEUP	DONE	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

Address: 0x09 (indirect)

Bit7: **ACTIVE**: LIN Bus Activity Bit.
0: No transmission activity detected on the LIN bus.
1: Transmission activity detected on the LIN bus.

Bit6: **IDLTOUT**: Bus Idle Timeout Bit (**slave mode only**).
0: The bus has not been idle for four seconds.
1: No bus activity has been detected for four seconds, but the bus is not yet in Sleep mode.

Bit5: **ABORT**: Aborted transmission signal (**slave mode only**).
0: The current transmission has not been interrupted or stopped. This bit is reset to 0 after receiving a SYNCH BREAK that does not interrupt a pending transmission.
1: New SYNCH BREAK detected before the end of the last transmission or the STOP bit (LIN0CTRL.7) has been set.

Bit4: **DTREQ**: Data Request bit (**slave mode only**).
0: Data identifier has not been received.
1: Data identifier has been received.

Bit3: **LININT**: Interrupt Request bit.
0: An interrupt is not pending. This bit is cleared by setting RSTINT (LIN0CTRL.3)
1: There is a pending LIN0 interrupt.

Bit2: **ERROR**: Communication Error Bit.
0: No error has been detected. This bit is cleared by setting RSTERR (LIN0CTRL.2)
1: An error has been detected.

Bit1: **WAKEUP**: Wakeup Bit.
0: A wakeup signal is not being transmitted and has not been received.
1: A wakeup signal is being transmitted or has been received.

Bit0: **DONE**: Transmission Complete Bit.
0: A transmission is not in progress or has not been started. This bit is cleared at the start of a transmission.
1: The current transmission is complete.

18. Timers

Each MCU includes three counter/timers: two are 16-bit counter/timers compatible with those found in the standard 8051, and one is a 16-bit auto-reload timer for use with other device peripherals or for general purpose use. These timers can be used to measure time intervals, count external events and generate periodic interrupt requests. Timer 0 and Timer 1 are nearly identical and have four primary modes of operation. Timer 2 offer 16-bit and split 8-bit timer functionality with auto-reload.

Timer 0 and Timer 1 Modes	Timer 2 Modes
13-bit counter/timer	16-bit timer with auto-reload
16-bit counter/timer	
8-bit counter/timer with auto-reload	Two 8-bit timers with auto-reload
Two 8-bit counter/timers (Timer 0 only)	

Timers 0 and 1 may be clocked by one of five sources, determined by the Timer Mode Select bits (T1M–T0M) and the Clock Scale bits (SCA1–SCA0). The Clock Scale bits define a pre-scaled clock from which Timer 0 and/or Timer 1 may be clocked (See SFR Definition 18.3 for pre-scaled clock selection).

Timer 0/1 may then be configured to use this pre-scaled clock signal or the system clock. Timer 2 may be clocked by the system clock, the system clock divided by 12, or the external oscillator clock source divided by 8.

Timer 0 and Timer 1 may also be operated as counters. When functioning as a counter, a counter/timer register is incremented on each high-to-low transition at the selected input pin (T0 or T1). Events with a frequency of up to one-fourth the system clock's frequency can be counted. The input signal need not be periodic, but it must be held at a given level for at least two full system clock cycles to ensure the level is properly sampled.

18.1. Timer 0 and Timer 1

Each timer is implemented as a 16-bit register accessed as two separate bytes: a low byte (TL0 or TL1) and a high byte (TH0 or TH1). The Counter/Timer Control register (TCON) is used to enable Timer 0 and Timer 1 as well as indicate status. Timer 0 interrupts can be enabled by setting the ET0 bit in the IE register (Section “10.4. Interrupt Register Descriptions” on page 100); Timer 1 interrupts can be enabled by setting the ET1 bit in the IE register (Section 10.4). Both counter/timers operate in one of four primary modes selected by setting the Mode Select bits T1M1–T0M0 in the Counter/Timer Mode register (TMOD). Each timer can be configured independently. Each operating mode is described below.

18.1.1. Mode 0: 13-bit Counter/Timer

Timer 0 and Timer 1 operate as 13-bit counter/timers in Mode 0. The following describes the configuration and operation of Timer 0. However, both timers operate identically, and Timer 1 is configured in the same manner as described for Timer 0.

The TH0 register holds the eight MSBs of the 13-bit counter/timer. TL0 holds the five LSBs in bit positions TL0.4–TL0.0. The three upper bits of TL0 (TL0.7–TL0.5) are indeterminate and should be masked out or ignored when reading. As the 13-bit timer register increments and overflows from 0x1FFF (all ones) to 0x0000, the timer overflow flag TF0 (TCON.5) is set and an interrupt will occur if Timer 0 interrupts are enabled.

The C/T0 bit (TMOD.2) selects the counter/timer's clock source. When C/T0 is set to logic 1, high-to-low transitions at the selected Timer 0 input pin (T0) increment the timer register (Refer to Section “13.1. Priority Crossbar Decoder” on page 122 for information on selecting and configuring external I/O pins). Clearing C/T selects the clock defined by the T0M bit (CKCON.3). When T0M is set, Timer 0 is

18.1.4. Mode 3: Two 8-bit Counter/Timers (Timer 0 Only)

In Mode 3, Timer 0 is configured as two separate 8-bit counter/timers held in TL0 and TH0. The counter/timer in TL0 is controlled using the Timer 0 control/status bits in TCON and TMOD: TR0, C/T0, GATE0 and TF0. TL0 can use either the system clock or an external input signal as its timebase. The TH0 register is restricted to a timer function sourced by the system clock or prescaled clock. TH0 is enabled using the Timer 1 run control bit TR1. TH0 sets the Timer 1 overflow flag TF1 on overflow and thus controls the Timer 1 interrupt.

Timer 1 is inactive in Mode 3. When Timer 0 is operating in Mode 3, Timer 1 can be operated in Modes 0, 1 or 2, but cannot be clocked by external signals nor set the TF1 flag and generate an interrupt. However, the Timer 1 overflow can be used to generate baud rates for the SMBus and UART. While Timer 0 is operating in Mode 3, Timer 1 run control is handled through its mode settings. To run Timer 1 while Timer 0 is in Mode 3, set the Timer 1 Mode as 0, 1, or 2. To disable Timer 1, configure it for Mode 3.

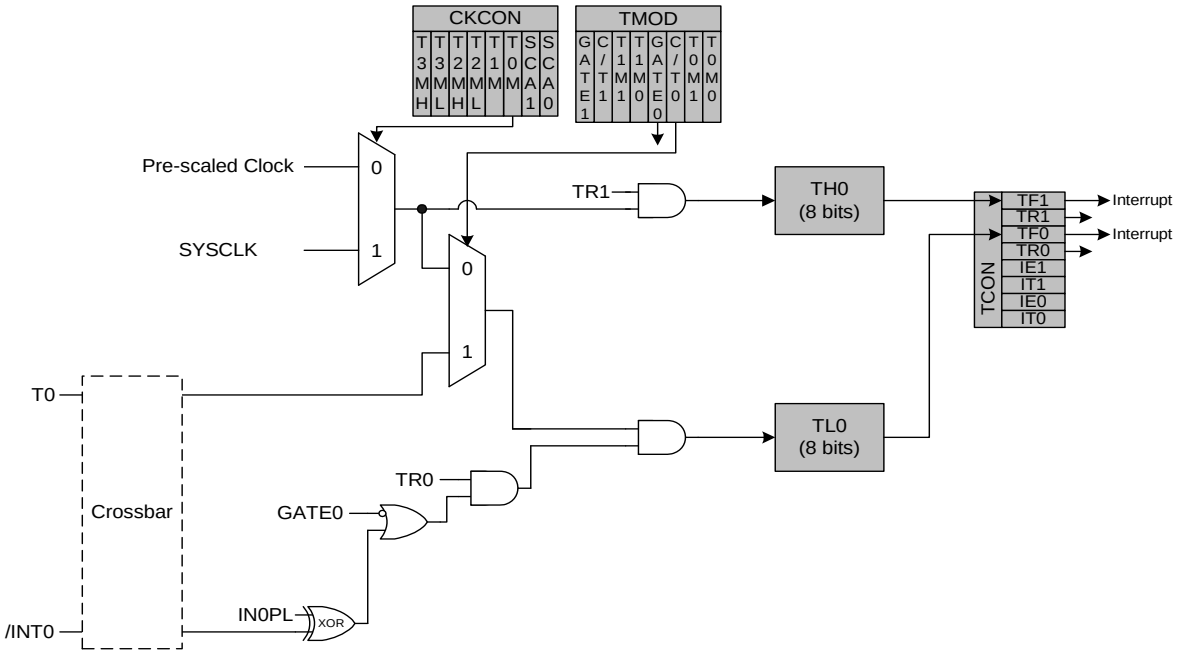

Figure 18.3. T0 Mode 3 Block Diagram

SFR Definition 18.1. TCON: Timer Control

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	Bit Addressable
							SFR Address:	0x88

Bit7: **TF1**: Timer 1 Overflow Flag.
Set by hardware when Timer 1 overflows. This flag can be cleared by software but is automatically cleared when the CPU vectors to the Timer 1 interrupt service routine.
0: No Timer 1 overflow detected.
1: Timer 1 has overflowed.

Bit6: **TR1**: Timer 1 Run Control.
0: Timer 1 disabled.
1: Timer 1 enabled.

Bit5: **TF0**: Timer 0 Overflow Flag.
Set by hardware when Timer 0 overflows. This flag can be cleared by software but is automatically cleared when the CPU vectors to the Timer 0 interrupt service routine.
0: No Timer 0 overflow detected.
1: Timer 0 has overflowed.

Bit4: **TR0**: Timer 0 Run Control.
0: Timer 0 disabled.
1: Timer 0 enabled.

Bit3: **IE1**: External Interrupt 1.
This flag is set by hardware when an edge/level of type defined by IT1 is detected. It can be cleared by software but is automatically cleared when the CPU vectors to the External Interrupt 1 service routine if IT1 = 1. When IT1 = 0, this flag is set to 1 when $\overline{INT0}$ is active as defined by bit IN1PL in register IT01CF (see SFR Definition 10.5. "IT01CF: INT0/INT1 Configuration" on page 105).

Bit2: **IT1**: Interrupt 1 Type Select.
This bit selects whether the configured $\overline{INT0}$ interrupt will be edge or level sensitive. $\overline{INT0}$ is configured active low or high by the IN1PL bit in the IT01CF register (see SFR Definition 10.5. "IT01CF: INT0/INT1 Configuration" on page 105).
0: $\overline{INT0}$ is level triggered.
1: $\overline{INT0}$ is edge triggered.

Bit1: **IE0**: External Interrupt 0.
This flag is set by hardware when an edge/level of type defined by IT0 is detected. It can be cleared by software but is automatically cleared when the CPU vectors to the External Interrupt 0 service routine if IT0 = 1. When IT0 = 0, this flag is set to 1 when $\overline{INT0}$ is active as defined by bit IN0PL in register IT01CF (see SFR Definition 10.5. "IT01CF: INT0/INT1 Configuration" on page 105).

Bit0: **IT0**: Interrupt 0 Type Select.
This bit selects whether the configured $\overline{INT0}$ interrupt will be edge or level sensitive. $\overline{INT0}$ is configured active low or high by the IN0PL bit in register IT01CF (see SFR Definition 10.5. "IT01CF: INT0/INT1 Configuration" on page 105).
0: $\overline{INT0}$ is level triggered.
1: $\overline{INT0}$ is edge triggered.

18.2.3. External Capture Mode

Capture Mode allows the external oscillator to be measured against the system clock. Timer 2 can be clocked from the system clock, or the system clock divided by 12, depending on the T2ML (CKCON.4) and T2XCLK bits. When a capture event is generated, the contents of Timer 2 (TMR2H:TMR2L) are loaded into the Timer 2 reload registers (TMR2RLH:TMR2RLL) and the TF2H flag is set. A capture event is generated by the falling edge of the clock source being measured, which is the external oscillator/8. By recording the difference between two successive timer capture values, the external oscillator frequency can be determined with respect to the Timer 2 clock. The Timer 2 clock should be much faster than the capture clock to achieve an accurate reading. Timer 2 should be in 16-bit auto-reload mode when using Capture Mode.

For example, if T2ML = 1b and TF2CEN = 1b, Timer 2 will clock every SYSCLK and capture every external clock divided by 8. If the SYSCLK is 24.5 MHz and the difference between two successive captures is 5984, then the external clock frequency is:

$$\frac{24.5\ \text{MHz}}{(5984/8)} = 0.032754\ \text{MHz or } 32.754\ \text{kHz}$$

This mode allows software to determine the external oscillator frequency when an RC network or capacitor is used to generate the clock source.

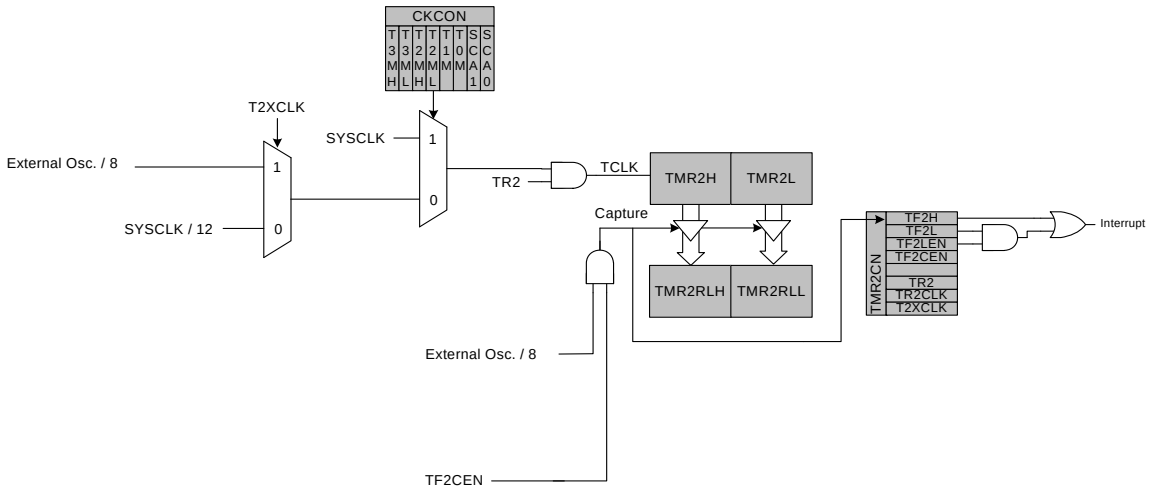

Figure 18.6. Timer 2 Capture Mode Block Diagram

SFR Definition 18.8. TMR2CN: Timer 2 Control

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
TF2H	TF2L	TF2LEN	TF2CEN	T2SPLIT	TR2	—	T2XCLK	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	Bit Addressable
							SFR Address:	0xC8

Bit7: **TF2H**: Timer 2 High Byte Overflow Flag.
Set by hardware when the Timer 2 high byte overflows from 0xFF to 0x00. In 16 bit mode, this will occur when Timer 2 overflows from 0xFFFF to 0x0000. When the Timer 2 interrupt is enabled, setting this bit causes the CPU to vector to the Timer 2 interrupt service routine. TF2H is not automatically cleared by hardware and must be cleared by software.

Bit6: **TF2L**: Timer 2 Low Byte Overflow Flag.
Set by hardware when the Timer 2 low byte overflows from 0xFF to 0x00. When this bit is set, an interrupt will be generated if TF2LEN is set and Timer 2 interrupts are enabled. TF2L will set when the low byte overflows regardless of the Timer 2 mode. This bit is not automatically cleared by hardware.

Bit5: **TF2LEN**: Timer 2 Low Byte Interrupt Enable.
This bit enables/disables Timer 2 Low Byte interrupts. If TF2LEN is set and Timer 2 interrupts are enabled, an interrupt will be generated when the low byte of Timer 2 overflows. This bit should be cleared when operating Timer 2 in 16-bit mode.
0: Timer 2 Low Byte interrupts disabled.
1: Timer 2 Low Byte interrupts enabled.

Bit4: **TF2CEN**. Timer 2 Capture Enable.
0: Timer 2 capture mode disabled.
1: Timer 2 capture mode enabled.

Bit3: **T2SPLIT**: Timer 2 Split Mode Enable.
When this bit is set, Timer 2 operates as two 8-bit timers with auto-reload.
0: Timer 2 operates in 16-bit auto-reload mode.
1: Timer 2 operates as two 8-bit auto-reload timers.

Bit2: **TR2**: Timer 2 Run Control.
This bit enables/disables Timer 2. In 8-bit mode, this bit enables/disables TMR2H only; TMR2L is always enabled in this mode.
0: Timer 2 disabled.
1: Timer 2 enabled.

Bit1: **Unused**. Read = 0b. Write = don't care.

Bit0: **T2XCLK**: Timer 2 External Clock Select.
This bit selects the external clock source for Timer 2. If Timer 2 is in 8-bit mode, this bit selects the external oscillator clock source for both timer bytes. However, the Timer 2 Clock Select bits (T2MH and T2ML in register CKCON) may still be used to select between the external clock and the system clock for either timer.
0: Timer 2 external clock selection is the system clock divided by 12.
1: Timer 2 external clock selection is the external clock divided by 8.

SFR Definition 18.9. TMR2RLL: Timer 2 Reload Register Low Byte

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

SFR Address: 0xCA

Bits7–0: **TMR2RLL**: Timer 2 Reload Register Low Byte.
TMR2RLL holds the low byte of the reload value for Timer 2.

SFR Definition 18.10. TMR2RLH: Timer 2 Reload Register High Byte

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

SFR Address: 0xCB

Bits7–0: **TMR2RLH**: Timer 2 Reload Register High Byte.
The TMR2RLH holds the high byte of the reload value for Timer 2.

SFR Definition 18.11. TMR2L: Timer 2 Low Byte

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

SFR Address: 0xCC

Bits7–0: **TMR2L**: Timer 2 Low Byte.
In 16-bit mode, the TMR2L register contains the low byte of the 16-bit Timer 2. In 8-bit mode, TMR2L contains the 8-bit low byte timer value.

SFR Definition 18.12. TMR2H Timer 2 High Byte

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

SFR Address: 0xCD

Bits7–0: **TMR2H**: Timer 2 High Byte.
In 16-bit mode, the TMR2H register contains the high byte of the 16-bit Timer 2. In 8-bit mode, TMR2H contains the 8-bit high byte timer value.

19.2.4. Frequency Output Mode

Frequency Output Mode produces a programmable-frequency square wave on the module's associated CEXn pin. The capture/compare module high byte holds the number of PCA clocks to count before the output is toggled. The frequency of the square wave is then defined by Equation 19.1.

$$F_{CEXn} = \frac{F_{PCA}}{2 \times PCA0CPHn}$$

Note: A value of 0x00 in the PCA0CPHn register is equal to 256 for this equation.

Equation 19.1. Square Wave Frequency Output

Where F_{PCA} is the frequency of the clock selected by the CPS2-0 bits in the PCA mode register, PCA0MD. The lower byte of the capture/compare module is compared to the PCA counter low byte; on a match, CEXn is toggled and the offset held in the high byte is added to the matched value in PCA0CPLn. Frequency Output Mode is enabled by setting the ECOMn, TOGn, and PWMn bits in the PCA0CPMn register.

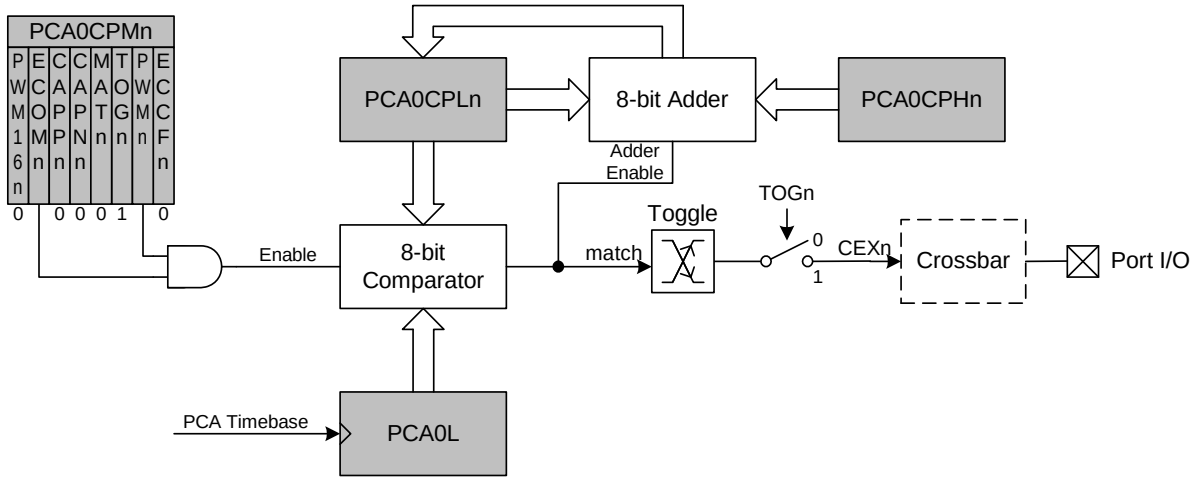

Figure 19.7. PCA Frequency Output Mode

20. Device Specific Behavior

This chapter contains behavioral differences between the silicon revisions of C8051F52x/52xA/F53x/53xA devices.

These differences do not affect the functionality or performance of most systems and are described below.

20.1. Device Identification

The Part Number Identifier on the top side of the device package can be used for decoding device information. The first character of the trace code identifies the silicon revision. On C8051F52x-C/53x-C devices, the trace code (second line on the TSSOP-20 and DFN-10 packages; third line on the QFN-20 package) will begin with the letter "C". The "A" suffix at the end of the part number such as "C8051F530A" is only present on Revision B devices. All other revisions do not include this suffix. Figures 20.1, 20.2, and 20.3 show how to find the part number on the top side of the device package.

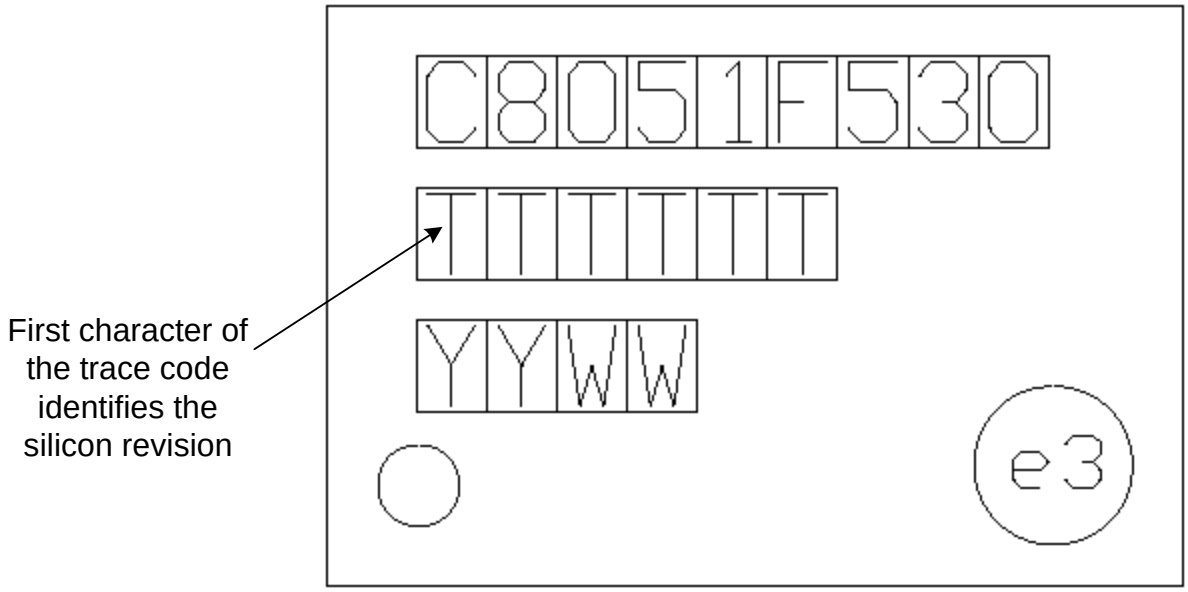

Figure 20.1. Device Package—TSSOP 20

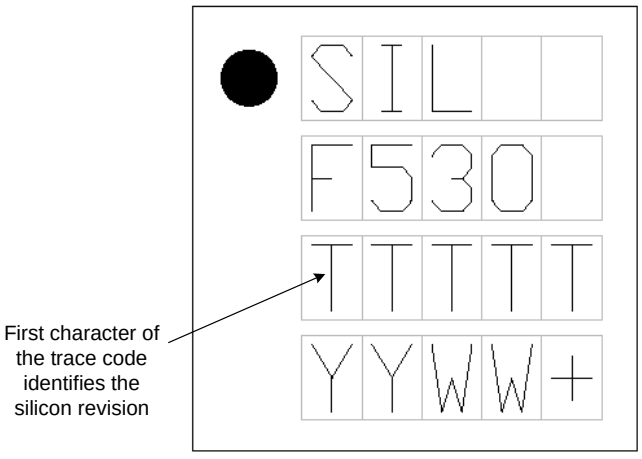

Figure 20.2. Device Package—QFN 20