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	16
Program Memory Size	4KB (4K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	256 x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 5.25V
Data Converters	A/D 16x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	20-TSSOP (0.173", 4.40mm Width)
Supplier Device Package	20-TSSOP
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051f534-c-it

1.1. Ordering Information

The following features are common to all devices in this family:

- 25 MHz system clock and 25 MIPS throughput (peak)
- 256 bytes of internal RAM
- Enhanced SPI peripheral
- Enhanced UART peripheral
- Three Timers
- Three Programmable Counter Array channels
- Internal 24.5 MHz oscillator
- Internal Voltage Regulator
- 12-bit, 200 ksps ADC
- Internal Voltage Reference and Temperature Sensor
- One Analog Comparator

Table 1.1 shows the features that differentiate the devices in this family.

Table 1.1. Product Selection Guide (Recommended for New Designs)

Ordering Part Number	Flash Memory (kB)	Port I/Os	LIN	Package	Ordering Part Number	Flash Memory (kB)	Port I/Os	LIN	Package
C8051F520-C-IM	8	6	✓	DFN-10	C8051F534-C-IM	4	16	—	QFN-20
C8051F521-C-IM	8	6	—	DFN-10	C8051F536-C-IM	2	16	✓	QFN-20
C8051F523-C-IM	4	6	✓	DFN-10	C8051F537-C-IM	2	16	—	QFN-20
C8051F524-C-IM	4	6	—	DFN-10	C8051F530-C-IT	8	16	✓	TSSOP-20
C8051F526-C-IM	2	6	✓	DFN-10	C8051F531-C-IT	8	16	—	TSSOP-20
C8051F527-C-IM	2	6	—	DFN-10	C8051F533-C-IT	4	16	✓	TSSOP-20
C8051F530-C-IM	8	16	✓	QFN-20	C8051F534-C-IT	4	16	—	TSSOP-20
C8051F531-C-IM	8	16	—	QFN-20	C8051F536-C-IT	2	16	✓	TSSOP-20
C8051F533-C-IM	4	16	✓	QFN-20	C8051F537-C-IT	2	16	—	TSSOP-20

All devices in Table 1.1 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 C8051F520-C-IM is the C8051F520-C-AM.

The -AM and -AT devices receive full automotive quality production status, including AEC-Q100 qualification (fault coverage report available upon request), 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 NDA and approved user account. The -AM and -AT devices enable high volume automotive OEM applications with their enhanced testing and processing. Please contact Silicon Labs sales for more information regarding -AM and -AT devices for your automotive project.

1.4. Operating Modes

The C8051F52x/F52xA/F53x/F53xA devices have four operating modes: Active (Normal), Idle, Suspend, and Stop. Active mode occurs during normal operation when the oscillator and peripherals are active. Idle mode halts the CPU while leaving the peripherals and internal clocks active. In Suspend and Stop mode, the CPU is halted, all interrupts and timers are inactive, and the internal oscillator is stopped. The various operating modes are described in Table 1.3 below:

Table 1.3. Operating Modes Summary

	Properties	Power Consumption	How Entered?	How Exited?
Active	■ SYSCLK active ■ CPU active (accessing Flash) ■ Peripherals active or inactive depending on user settings	Full	—	—
Idle	■ SYSCLK active ■ CPU inactive (not accessing Flash) ■ Peripherals active or inactive depending on user settings	Less than Full	IDLE (PCON.0)	Any enabled interrupt or device reset
Suspend	■ Internal oscillator inactive ■ If SYSCLK is derived from the internal oscillator, the peripherals and the CIP-51 will be stopped	Low	SUSPEND (OSCICN.5)	Port 0 event match Port 1 event match Comparator 0 enabled and output is logic 0
Stop	■ SYSCLK inactive ■ CPU inactive (not accessing Flash) ■ Digital peripherals inactive; analog peripherals active or inactive depending on user settings	Very low	STOP (PCON.1)	Device Reset

See Section “8.3. Power Management Modes” on page 89 for Idle and Stop mode details. See Section “14.1.1. Internal Oscillator Suspend Mode” on page 136 for more information on Suspend mode.

1.9. Port Input/Output

C8051F52x/F52xA/F53x/F53xA devices include up to 16 I/O pins. Port pins are organized as two byte-wide ports. The port pins behave like typical 8051 ports with a few enhancements. Each port pin can be configured as a digital or analog I/O pin. Pins selected as digital I/O can be configured for push-pull or open-drain operation. The “weak pullups” that are fixed on typical 8051 devices may be globally disabled to save power.

The Digital Crossbar allows mapping of internal digital system resources to port I/O pins. On-chip counter/timers, serial buses, hardware interrupts, and other digital signals can be configured to appear on the port pins using the Crossbar control registers. This allows the user to select the exact mix of general-purpose port I/O, digital, and analog resources needed for the application.

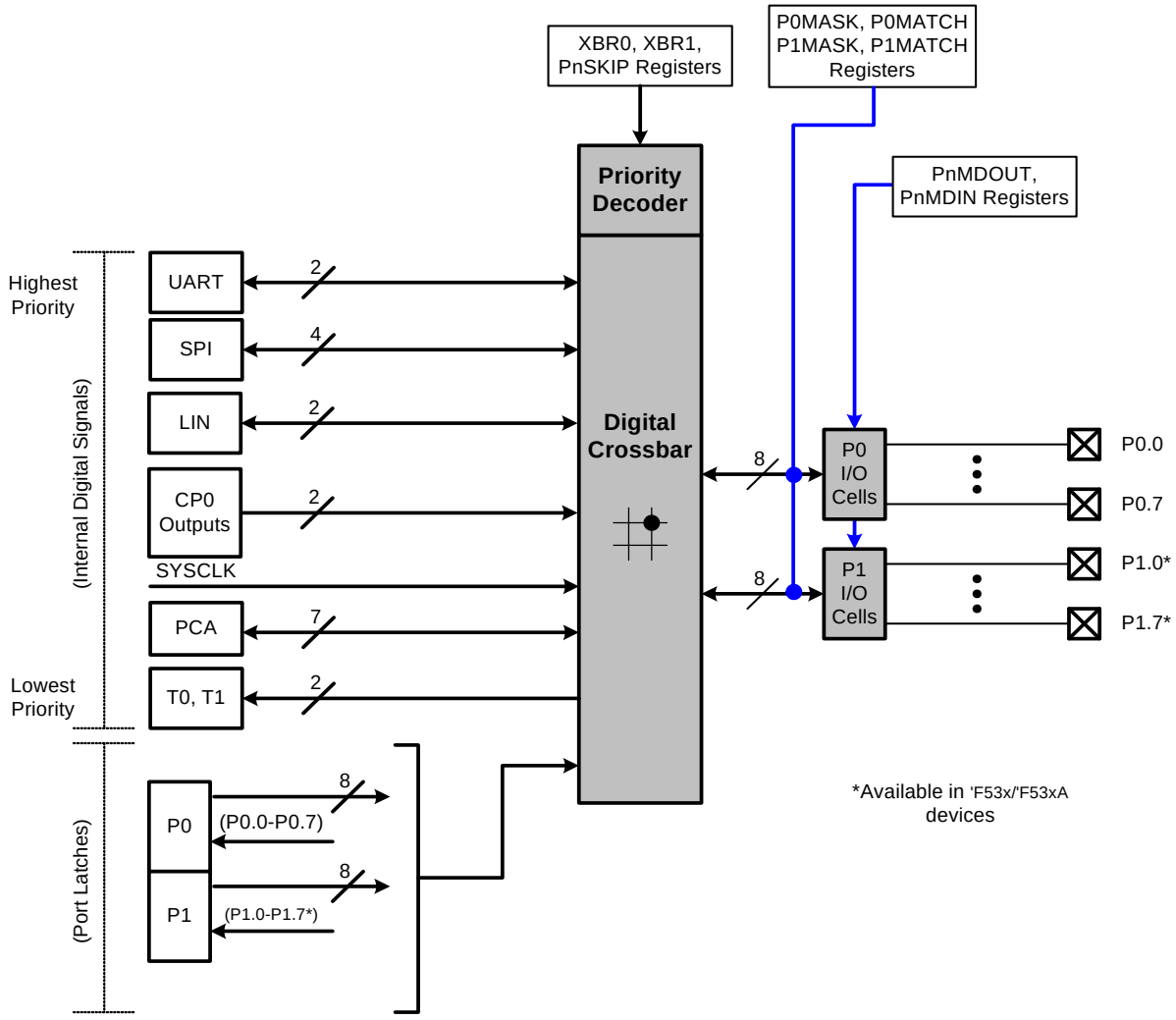


Figure 1.9. Port I/O Functional Block Diagram

Table 2.4. Temperature Sensor Electrical Characteristics

V_{DD} = 2.1 V, V_{REF} = 1.5 V (REFSL=0), –40 to +125 °C unless otherwise specified.

Parameter	Conditions	Min	Typ	Max	Units
Linearity[1]		—	0.1	—	°C
Gain[1]		—	3.33	—	mV/°C
Gain Error[2]		—	±100	—	µV/°C
Offset[1]	Temp = 0 °C	—	890	—	mV
Offset Error[2]	Temp = 0 °C	—	±15	—	mV
Tracking Time		12	—	—	µs
Power Supply Current		—	17	—	µA

Notes:
1. Includes ADC offset, gain, and linearity variations.
2. Represents one standard deviation from the mean.

Table 2.5. Voltage Reference Electrical Characteristics

V_{DD} = 2.1 V; –40 to +125 °C unless otherwise specified.

Parameter	Conditions	Min	Typ	Max	Units
Internal Reference (REFBE = 1)					
Output Voltage	I_{DD} ≈ 1 mA; No load on VREF pin and all other GPIO pins. 25 °C ambient (REFLV = 0) 25 °C ambient (REFLV = 1), V_{DD} = 2.6 V	 1.45 2.15	 1.5 2.2	 1.55 2.25	 V
V_{REF} Short-Circuit Current		—	2.5	—	mA
V_{REF} Temperature Coefficient		—	33	—	ppm/°C
Load Regulation	Load = 0 to 200 µA to GND	—	10	—	ppm/µA
V_{REF} Turn-on Time 1	4.7 µF, 0.1 µF bypass	—	21	—	ms
V_{REF} Turn-on Time 2	0.1 µF bypass	—	230	—	µs
Power Supply Rejection		—	2.1	—	mV/V
External Reference (REFBE = 0)					
Input Voltage Range		0	—	V_{DD}	V
Input Current	Sample Rate = 200 ksps; V_{REF} = 1.5 V	—	2.4	—	µA
Bias Generators					
ADC Bias Generator	BIASE = 1	—	22	—	µA
Power Consumption (Internal)		—	35	—	µA

Gain Register Definition 4.1. ADC0GNH: ADC0 Selectable Gain High Byte

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
GAINH[7:0]								11111100
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	Address:
								0x04

Bits7–0: High byte of Selectable Gain Word.

Gain Register Definition 4.2. ADC0GNL: ADC0 Selectable Gain Low Byte

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
GAINL[3:0]				Reserved	Reserved	Reserved	Reserved	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	Address:
								0x07

Bits7–4: Lower 4 bits of the Selectable Gain Word.
Bits3–0: **Reserved**. Must Write 0000b.

Gain Register Definition 4.3. ADC0GNA: ADC0 Additional Selectable Gain

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	GAINADD	00000001
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	Address:
								0x08

Bits7–1: **Reserved**. Must Write 0000000b.
Bit0: **GAINADD**: Additional Gain Bit.
Setting this bit adds 1/64 (0.016) gain to the gain value in the ADC0GNH and ADC0GNL registers.

4.5. Programmable Window Detector

The ADC Programmable Window Detector continuously compares the ADC0 output registers to user-programmed limits, and notifies the system when a desired condition is detected. This is especially effective in an interrupt-driven system, saving code space and CPU bandwidth while delivering faster system response times. The window detector interrupt flag (AD0WINT in register ADC0CN) can also be used in polled mode. The ADC0 Greater-Than (ADC0GTH, ADC0GTL) and Less-Than (ADC0LTH, ADC0LTL) registers hold the comparison values. The window detector flag can be programmed to indicate when measured data is inside or outside of the user-programmed limits, depending on the contents of the ADC0 Less-Than and ADC0 Greater-Than registers.

SFR Definition 4.10. ADC0GTH: ADC0 Greater-Than Data High Byte

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:
								0xC4

Bits7–0: High byte of ADC0 Greater-Than Data Word.

SFR Definition 4.11. ADC0GTL: ADC0 Greater-Than Data Low Byte

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:
								0xC3

Bits7–0: Low byte of ADC0 Greater-Than Data Word.

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 7.3. CPT0MD: Comparator0 Mode Selection

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
Reserved	—	CP0RIE	CP0FIE	—	—	CP0MD1	CP0MD0	00000010
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0x9D

Bit7: **RESERVED**. Read = 0b. Must write 0b.
Bit6: **UNUSED**. Read = 0b. Write = don't care.
Bit5: **CP0RIE**: Comparator Rising-Edge Interrupt Enable.
0: Comparator rising-edge interrupt disabled.
1: Comparator rising-edge interrupt enabled.
Bit4: **CP0FIE**: Comparator Falling-Edge Interrupt Enable.
0: Comparator falling-edge interrupt disabled.
1: Comparator falling-edge interrupt enabled.
Note: It is necessary to enable both CP0xIE and the correspondent ECPx bit located in EIE1 SFR.
Bits3–2: **UNUSED**. Read = 00b. Write = don't care.
Bits1–0: **CP0MD1–CP0MD0**: Comparator0 Mode Select
These bits select the response time for Comparator0.

Mode	CP0MD1	CP0MD0	CP0 Falling Edge Response Time (TYP)
0	0	0	Fastest Response Time
1	0	1	—
2	1	0	—
3	1	1	Lowest Power Consumption

Note: Rising Edge response times are approximately double the Falling Edge response times.

11.2. Power-Fail Reset / V_{DD} Monitors (VDDMON0 and VDDMON1)

C8051F52x-C/F53x-C devices include two V_{DD} monitors: a standard V_{DD} monitor (VDDMON0) and a level-sensitive V_{DD} monitor (VDDMON1). VDDMON0 is primarily intended for setting a higher threshold to allow safe erase or write of Flash memory from firmware. VDDMON1 is used to hold the device in a reset state during power-up and brownout conditions.

Note: VDDMON1 is not present in older silicon revisions A and B. Please refer to Section “20.4. VDD Monitors and VDD Ramp Time” on page 211 for more details.

When a power-down transition or power irregularity causes V_{DD} to drop below V_{RST}, the power supply monitors (VDDMON0 and VDDMON1) will drive the $\overline{\text{RST}}$ pin low and hold the CIP-51 in a reset state (see Figure 11.2). 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.

VDDMON0 is enabled and is selected as a reset source after power-on resets; however its defined state (enabled/disabled) is not altered by any other reset source. For example, if VDDMON0 is disabled by software, and a software reset is performed, VDDMON0 will still be disabled after that reset.

VDDMON1 is enabled and is selected as a reset source after power-on reset and any other type of reset. There is no register setting that can disable this level-sensitive VDD monitor as a reset source.

To protect the integrity of Flash contents, the V_{DD} monitor (VDDMON0) 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 higher setting, any erase or write performed on Flash memory will cause a Flash Error device reset.

Note: Please refer to Section “20.5. VDD Monitor (VDDMON0) High Threshold Setting” on page 212 for important notes related to the VDD Monitor high threshold setting in older silicon revisions A and B.

The V_{DD} monitor (VDDMON0) must be enabled before it is selected as a reset source. Selecting the VDDMON0 as a reset source before it is enabled and stabilized may cause a system reset. The procedure for re-enabling the V_{DD} monitor and configuring the V_{DD} monitor as a reset source is shown below:

1. Enable the V_{DD} monitor (VDMEN bit in VDDMON = 1).
2. Wait for the V_{DD} monitor to stabilize (see Table 2.8 on page 32 for the V_{DD} Monitor turn-on time). **Note: This delay should be omitted if software contains routines which write or erase Flash memory.**
3. Select the V_{DD} monitor as a reset source (PORSF bit in RSTSRC = 1).

See Figure 11.2 for V_{DD} monitor timing; note that the reset delay is not incurred after a V_{DD} monitor reset. See Table 2.8 on page 32 for complete electrical characteristics of the V_{DD} monitor.

Note: Software should take care not to inadvertently disable the V_{DD} Monitor (VDDMON0) as a reset source when writing to RSTSRC to enable other reset sources or to trigger a software reset. All writes to RSTSRC should explicitly set PORSF to '1' to keep the V_{DD} Monitor enabled as a reset source.

11.2.1. VDD Monitor Thresholds and Minimum VDD

The minimum operating digital supply voltage (V_{DD}) is specified as 2.0 V in Table 2.2 on page 26. The voltage at which the MCU is released from reset (V_{RST}) can be as low as 1.65 V based on the V_{DD} Monitor thresholds that are specified in Table 2.8 on page 32. This could allow code execution during the power-up

12.2. Flash Write and Erase Guidelines

Any system which contains routines which write or erase Flash memory from software involves some risk that the write or erase routines will execute unintentionally if the CPU is operating outside its specified operating range of V_{DD}, system clock frequency, or temperature. This accidental execution of Flash modifying code can result in alteration of Flash memory contents causing a system failure that is only recoverable by re-Flashing the code in the device.

The following guidelines are recommended for any system which contains routines which write or erase Flash from code.

12.2.1. V_{DD} Maintenance and the V_{DD} monitor

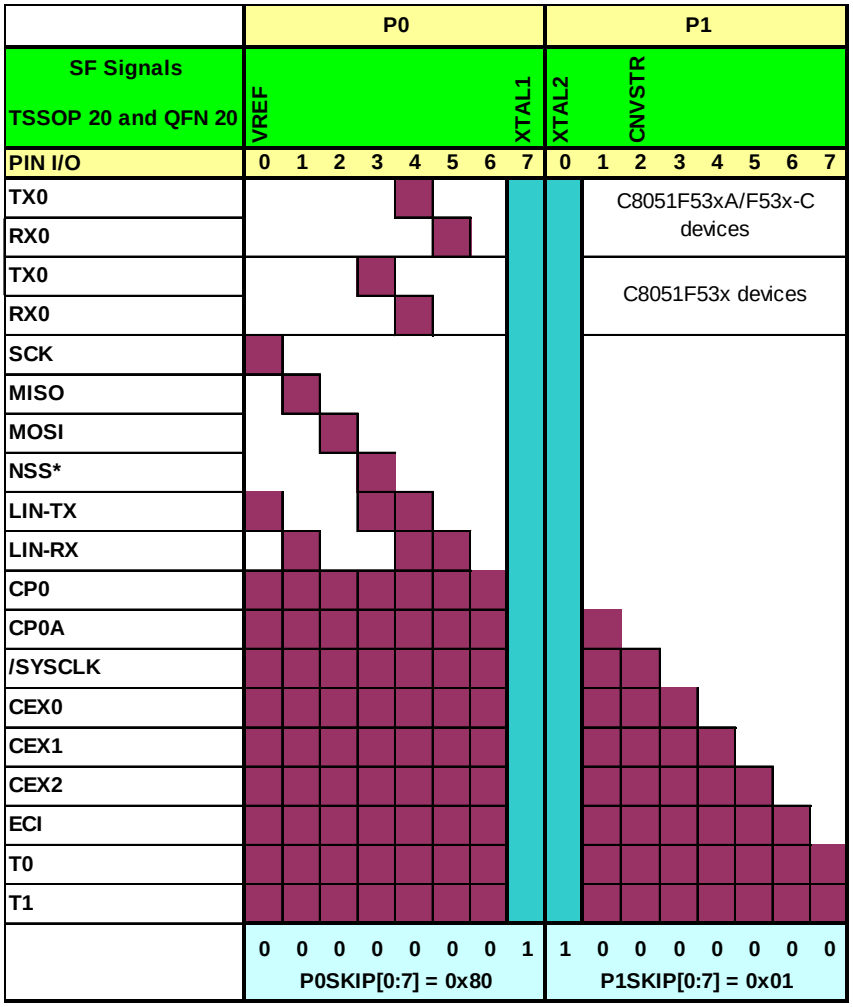
1. If the system power supply is subject to voltage or current "spikes," add sufficient transient protection devices to the power supply to ensure that the supply voltages listed in the Absolute Maximum Ratings table are not exceeded.
2. Make certain that the maximum V_{DD} ramp time specification (if applicable) is met. See Section 20.4 on page 211 for more details on V_{DD} ramp time. If the system cannot meet this ramp time specification, then add an external V_{DD} brownout circuit to the $\overline{RST}$ pin of the device that holds the device in reset until V_{DD} reaches the minimum specified V_{DD} and re-asserts $\overline{RST}$ if V_{DD} drops belowthat level. V_{DD} (min) is specified in Table 2.2 on page 26.
3. Enable the on-chip V_{DD} monitor (VDDMON0) and enable it as a reset source as early in code as possible. This should be the first set of instructions executed after the Reset Vector. For C-based systems, this will involve modifying the startup code added by the C compiler. See your compiler documentation for more details. Make certain that there are no delays in software between enabling the V_{DD} monitor (VDDMON0) and enabling it as a reset source. Code examples showing this can be found in “AN201: Writing to Flash from Firmware", available from the Silicon Laboratories web site.
4. As an added precaution, explicitly enable the V_{DD} monitor (VDDMON0) and enable the V_{DD} monitor as a reset source inside the functions that write and erase Flash memory. The V_{DD} monitor enable instructions should be placed just after the instruction to set PSWE to a 1, but before the Flash write or erase operation instruction.
5. Make certain that all writes to the RSTSRC (Reset Sources) register use direct assignment operators and explicitly DO NOT use the bit-wise operators (such as AND or OR). For example, "RSTSRC = 0x02" is correct. "RSTSRC |= 0x02" is incorrect.
6. Make certain that all writes to the RSTSRC register explicitly set the PORSF bit to a 1. Areas to check are initialization code which enables other reset sources, such as the Missing Clock Detector or Comparator, for example, and instructions which force a Software Reset. A global search on "RSTSRC" can quickly verify this.

12.2.2. PSWE Maintenance

1. Reduce the number of places in code where the PSWE bit (PSCTL.0) is set to a 1. There should be exactly one routine in code that sets PSWE to a 1 to write Flash bytes and one routine in code that sets PSWE and PSEE both to a 1 to erase Flash pages.
2. Minimize the number of variable accesses while PSWE is set to a 1. Handle pointer address updates and loop variable maintenance outside the "PSWE = 1;... PSWE = 0;" area. Code examples showing this can be found in “AN201: Writing to Flash from Firmware,” available from the Silicon Laboratories web site.
3. Disable interrupts prior to setting PSWE to a 1 and leave them disabled until after PSWE has been reset to '0'. Any interrupts posted during the Flash write or erase operation will be serviced in priority order after the Flash operation has been completed and interrupts have been re-enabled by software.
4. Make certain that the Flash write and erase pointer variables are not located in XRAM. See your compiler documentation for instructions regarding how to explicitly locate variables in different memory areas.

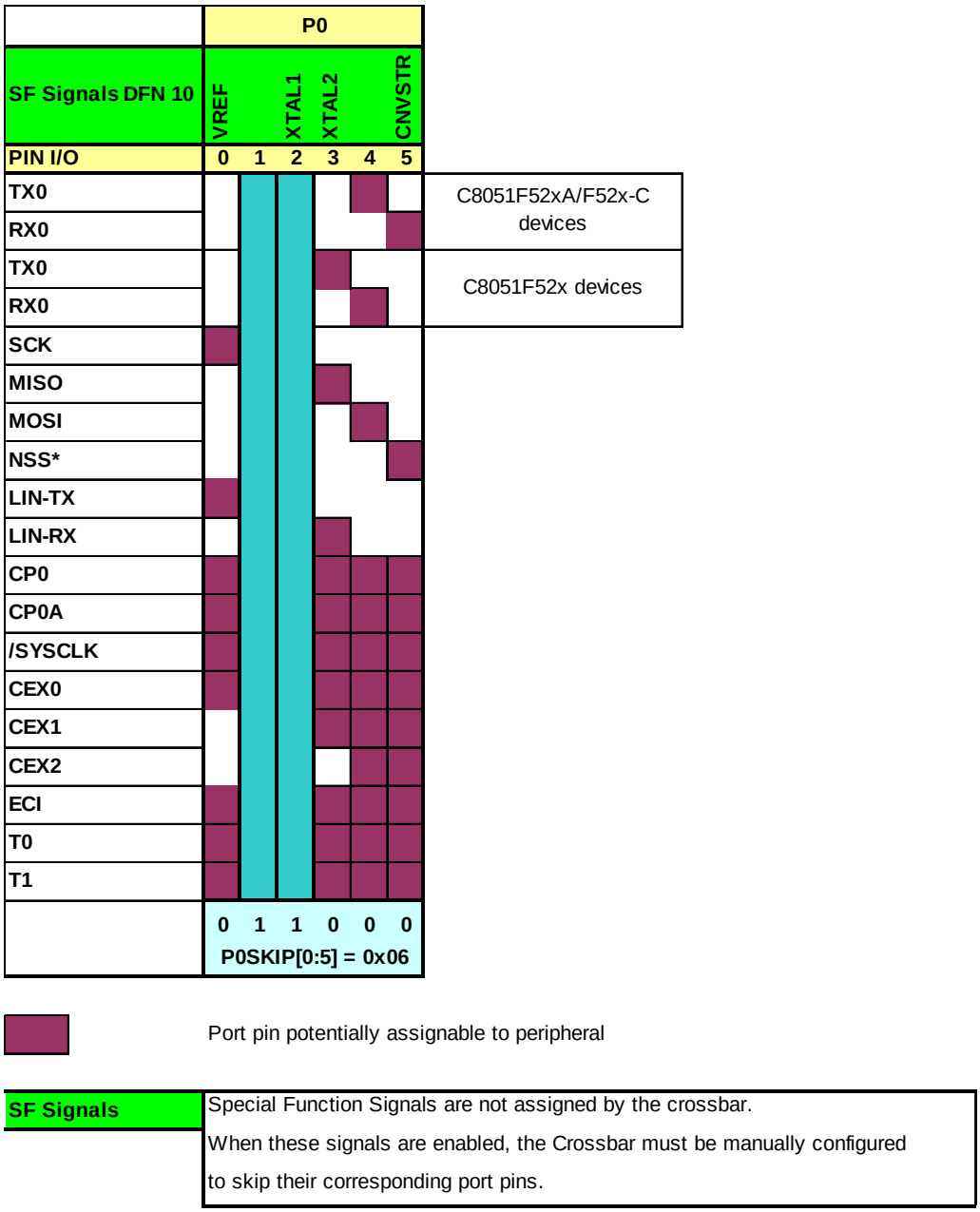
(F52x/F52xA) for the external CNVSTR signal, and any selected ADC or comparator inputs. The Crossbar skips selected pins as if they were already assigned, and moves to the next unassigned pin. Figure 13.3 shows the Crossbar Decoder priority with no Port pins skipped (P0SKIP, P1SKIP); Figure 13.4 shows the Crossbar Decoder priority with the XTAL1 (P1.0) and XTAL2 (P1.1) pins skipped (P1SKIP = 0x03).

Important Note on UART Pins: On C8051F52xA/F52x-C/F53xA/F53x-C devices, the UART pins must be skipped if the UART is enabled in order for peripherals to appear on port pins beyond the UART on the crossbar. For example, with the SPI and UART enabled on the crossbar with the SPI on P1.0-P1.3, the UART pins must be skipped using P0SKIP for the SPI pins to appear correctly.



Note: 4-Wire SPI Only.

Figure 13.4. Crossbar Priority Decoder with Crystal Pins Skipped (TSSOP 20 and QFN 20)



Note: 4-Wire SPI Only.

Figure 13.6. Crossbar Priority Decoder with Some Pins Skipped (DFN 10)

Registers XBR0 and XBR1 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); when the UART is selected, the Crossbar assigns both pins associated with the UART (TX and RX). UART0 pin assignments are fixed for bootloading purposes: UART TX0 is always assigned to P0.3 or P0.4*; UART RX0 is always assigned to P0.4 or P0.5*. Standard Port I/Os appear contiguously starting at P0.0 after prioritized functions and skipped pins are assigned.

Note: Refer to Section “20. Device Specific Behavior” on page 210.

SFR Definition 13.7. P0MAT: Port0 Match

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: 0xD7

Bits7–0: **P0MAT[7:0]**: Port0 Match Value.
These bits control the value that unmasked P0 Port pins are compared against. A Port Match event is generated if (P0 & P0MASK) does not equal (P0MAT & P0MASK).

SFR Definition 13.8. P0MASK: Port0 Mask

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: 0xC7

Bits7–0: **P0MASK[7:0]**: Port0 Mask Value.
These bits select which Port pins will be compared to the value stored in P0MAT.
0: Corresponding P0.n pin is ignored and cannot cause a Port Match event.
1: Corresponding P0.n pin is compared to the corresponding bit in P0MAT.

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

The equation for determining the load capacitance for two capacitors is:

$$C_L = \frac{C_A \times C_B}{C_A + C_B} + C_S$$

Where:
C_A and C_B are the capacitors connected to the crystal leads.
C_S is the total stray capacitance of the PCB.
The stray capacitance for a typical layout where the crystal is as close as possible to the pins is 2–5 pF per pin.

If C_A and C_B are the same (C), then the equation becomes:

$$C_L = \frac{C}{2} + C_S$$

For example, a tuning-fork crystal of 32 kHz with a recommended load capacitance of 12.5 pF should use the configuration shown in Figure 14.1, Option 1. With a stray capacitance of 3 pF per pin (6 pF total), the 13 pF capacitors yield an equivalent capacitance of 12.5 pF across the crystal, as shown in Figure 14.2.

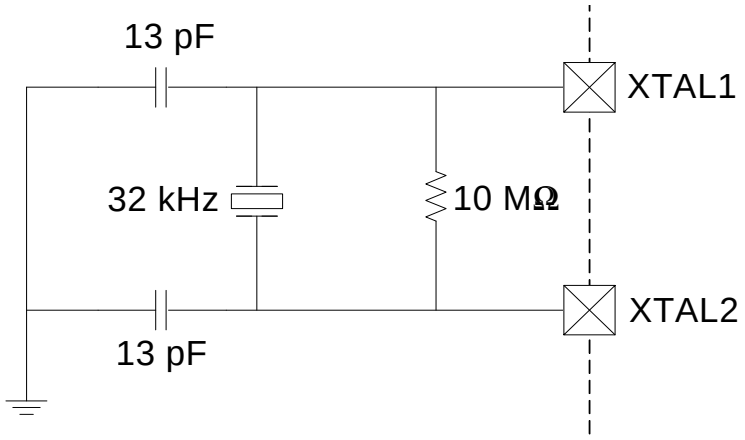


Figure 14.2. 32 kHz External Crystal Example

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.

15.1. Enhanced Baud Rate Generation

The UART0 baud rate is generated by Timer 1 in 8-bit auto-reload mode. The TX clock is generated by TL1; the RX clock is generated by a copy of TL1 (shown as RX Timer in Figure 15.2), which is not user-accessible. Both TX and RX Timer overflows are divided by two to generate the TX and RX baud rates. The RX Timer runs when Timer 1 is enabled, and uses the same reload value (TH1). However, an RX Timer reload is forced when a START condition is detected on the RX pin. This allows a receive to begin any time a START is detected, independent of the TX Timer state.

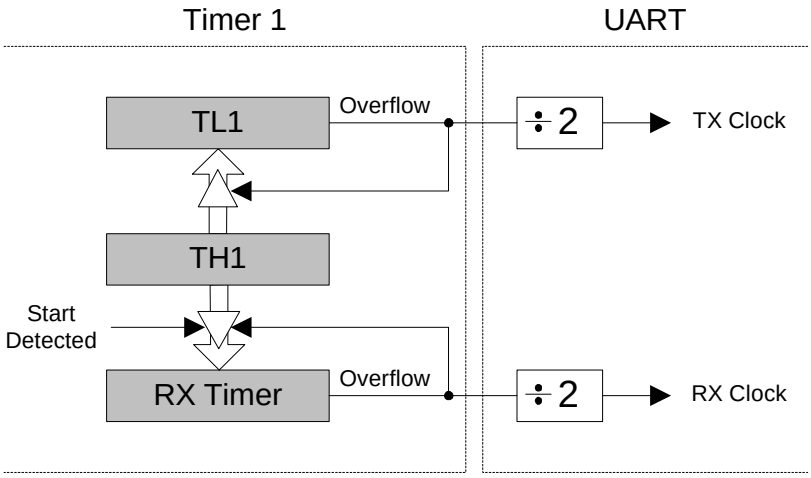


Figure 15.2. UART0 Baud Rate Logic

Timer 1 should be configured for Mode 2, 8-bit auto-reload (see Section “18.1.3. Mode 2: 8-bit Counter/Timer with Auto-Reload” on page 184). The Timer 1 reload value should be set so that overflows will occur at two times the desired UART baud rate frequency. Note that Timer 1 may be clocked by one of six sources: SYSCLK, SYSCLK / 4, SYSCLK / 12, SYSCLK / 48, the external oscillator clock / 8, or an external input T1. The UART0 baud rate is determined by Equation 15.1-A and Equation 15.1-B.

A) $$\text{UartBaudRate} = \frac{1}{2} \times \text{T1\_Overflow\_Rate}$$

B) $$\text{T1\_Overflow\_Rate} = \frac{\text{T1}_{CLK}}{256 - \text{TH1}}$$

Equation 15.1. UART0 Baud Rate

Where $T1_{CLK}$ is the frequency of the clock supplied to Timer 1, and *T1H* is the high byte of Timer 1 (8-bit auto-reload mode reload value). Timer 1 clock frequency is selected as described in Section “18. Timers” on page 182. A quick reference for typical baud rates and system clock frequencies is given in Table 15.1. Note that the internal oscillator may still generate the system clock when the external oscillator is driving Timer 1.

15.3. Multiprocessor Communications

9-Bit UART mode supports multiprocessor communication between a master processor and one or more slave processors by special use of the ninth data bit. When a master processor wants to transmit to one or more slaves, it first sends an address byte to select the target(s). An address byte differs from a data byte in that its ninth bit is logic 1; in a data byte, the ninth bit is always set to logic 0.

Setting the MCE0 bit (SCON0.5) of a slave processor configures its UART such that when a stop bit is received, the UART will generate an interrupt only if the ninth bit is logic 1 (RB80 = 1) signifying an address byte has been received. In the UART interrupt handler, software will compare the received address with the slave's own assigned 8-bit address. If the addresses match, the slave will clear its MCE0 bit to enable interrupts on the reception of the following data byte(s). Slaves that weren't addressed leave their MCE0 bits set and do not generate interrupts on the reception of the following data bytes, thereby ignoring the data. Once the entire message is received, the addressed slave resets its MCE0 bit to ignore all transmissions until it receives the next address byte.

Multiple addresses can be assigned to a single slave and/or a single address can be assigned to multiple slaves, thereby enabling "broadcast" transmissions to more than one slave simultaneously. The master processor can be configured to receive all transmissions or a protocol can be implemented such that the master/slave role is temporarily reversed to enable half-duplex transmission between the original master and slave(s).

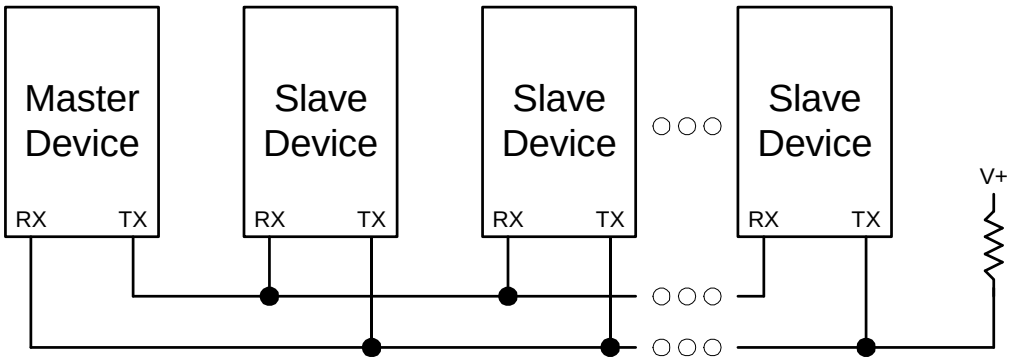


Figure 15.6. UART Multi-Processor Mode Interconnect Diagram

17.3. LIN Master Mode Operation

The master node is responsible for the scheduling of messages and sends the header of each frame, containing the SYNCH BREAK FIELD, SYNCH FIELD and IDENTIFIER FIELD. The steps to schedule a message transmission or reception are listed below.

1. Load the 6-bit Identifier into the LIN0ID register.
2. Load the data length into the LIN0SIZE register. Set the value to the number of data bytes or "1111b" if the data length should be decoded from the identifier. Also, set the checksum type, classic or enhanced, in the same LIN0SIZE register.
3. Set the data direction by setting the TXRX bit (LIN0CTRL.5). Set the bit to 1 to perform a master transmit operation, or set the bit to 0 to perform a master receive operation.
4. If performing a master transmit operation, load the data bytes to transmit into the data buffer (LIN0DT1 to LIN0DT8).
5. Set the STREQ bit (LIN0CTRL.0) to start the message transfer. The LIN peripheral will schedule the message frame and request an interrupt if the message transfer is successfully completed or if an error has occurred.

This code segment shows the procedure to schedule a message in a transmission operation:

```
LINADDR   = 0x08;// Point to LIN0CTRL
LINDATA  |=  0x20;// Select to transmit data
LINADDR   = 0x0E;// Point to LIN0ID
LINDATA   = 0x11;// Load the ID, in this example 0x11
LINADDR   = 0x0B;// Point to LIN0SIZE
LINDATA   = ( LINDATA & 0xF0 ) | 0x08; // Load the size with 8

LINADDR = 0x00;// Point to Data buffer first byte
for (i=0; i<8; i++)
{
   LINDATA = i + 0x41;// Load the buffer with ‘A’, ‘B’, ...
   LINADDR++;// Increment the address to the next buffer
}
LINADDR   = 0x08;// Point to LIN0CTRL
LINDATA   = 0x01;// Start Request
```

The application should perform the following steps when an interrupt is requested.

1. Check the DONE bit (LIN0ST.0) and the ERROR bit (LIN0ST.2).
2. If performing a master receive operation and the transfer was successful, read the received data from the data buffer.
3. If the transfer was not successful, check the error register to determine the kind of error. Further error handling has to be done by the application.
4. Set the RSTINT (LIN0CTRL.3) and RSTERR bits (LIN0CTRL.2) to reset the interrupt request and the error flags.

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.

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.

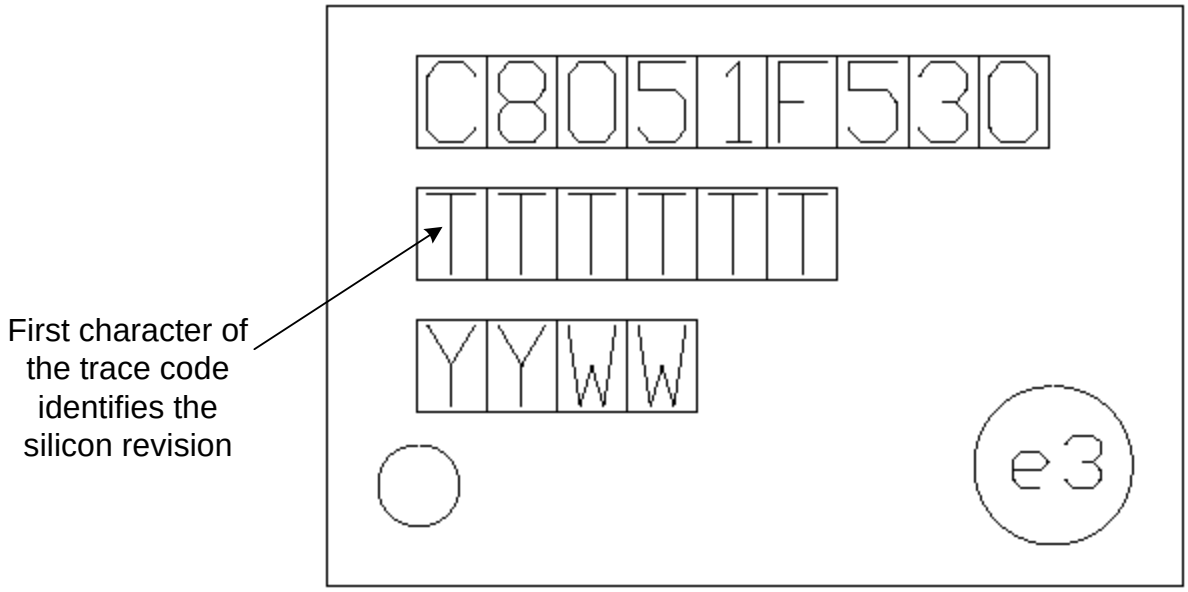


Figure 20.1. Device Package—TSSOP 20

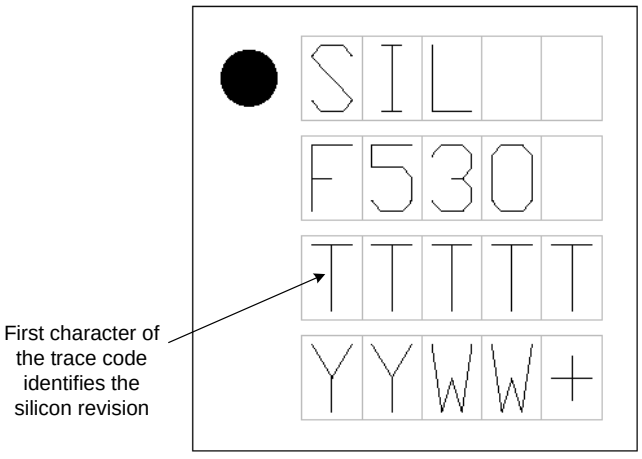


Figure 20.2. Device Package—QFN 20