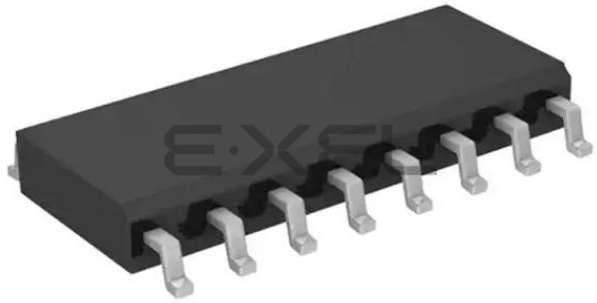

Welcome to E-XFL.COM

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	I²C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	13
Program Memory Size	2KB (2K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	256 x 8
Voltage - Supply (Vcc/Vdd)	2.2V ~ 3.6V
Data Converters	A/D 12x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	16-SOIC (0.154", 3.90mm Width)
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051f862-c-isr

Table 1.2. Power Consumption (Continued)

Parameter	Symbol	Test Condition	Min	Typ	Max	Unit
ADC0 Burst Mode, 10-bit single conversions, internal reference, Low power bias settings	I_{ADC}	200 ksps, V_{DD} = 3.0 V	—	490	—	µA
		100 ksps, V_{DD} = 3.0 V	—	245	—	µA
		10 ksps, V_{DD} = 3.0 V	—	23	—	µA
ADC0 Burst Mode, 12-bit single conversions, external reference	I_{ADC}	100 ksps, V_{DD} = 3.0 V	—	530	—	µA
		50 ksps, V_{DD} = 3.0 V	—	265	—	µA
		10 ksps, V_{DD} = 3.0 V	—	53	—	µA
ADC0 Burst Mode, 12-bit single conversions, internal reference	I_{ADC}	100 ksps, V_{DD} = 3.0 V, Normal bias	—	950	—	µA
		50 ksps, V_{DD} = 3.0 V, Low power bias	—	420	—	µA
		10 ksps, V_{DD} = 3.0 V, Low power bias	—	85	—	µA
Internal ADC0 Reference, Always-on[5]	I_{IREF}	Normal Power Mode	—	680	790	µA
		Low Power Mode	—	160	210	µA
Temperature Sensor	I_{TSENSE}		—	75	120	µA
Comparator 0 (CMP0), Comparator 1 (CMP1)	I_{CMP}	CPnMD = 11	—	0.5	—	µA
		CPnMD = 10	—	3	—	µA
		CPnMD = 01	—	10	—	µA
		CPnMD = 00	—	25	—	µA
Voltage Supply Monitor (VMON0)	I_{VMON}		—	15	20	µA

Notes:

1. Currents are additive. For example, where I_{DD} is specified and the mode is not mutually exclusive, enabling the functions increases supply current by the specified amount.
2. Includes supply current from internal regulator, supply monitor, and High Frequency Oscillator.
3. Includes supply current from internal regulator, supply monitor, and Low Frequency Oscillator.
4. ADC0 always-on power excludes internal reference supply current.
5. The internal reference is enabled as-needed when operating the ADC in burst mode to save power.

Table 1.3. Reset and Supply Monitor

Parameter	Symbol	Test Condition	Min	Typ	Max	Unit
V_{DD} Supply Monitor Threshold	V_{VDDM}		1.85	1.95	2.1	V
Power-On Reset (POR) Threshold	V_{POR}	Rising Voltage on V_{DD}	—	1.4	—	V
		Falling Voltage on V_{DD}	0.75	—	1.36	V
V_{DD} Ramp Time	t_{RMP}	Time to $V_{DD} \geq 2.2$ V	10	—	—	µs
Reset Delay from POR	t_{POR}	Relative to $V_{DD} \geq V_{POR}$	3	10	31	ms
Reset Delay from non-POR source	t_{RST}	Time between release of reset source and code execution	—	39	—	µs
$\overline{RST}$ Low Time to Generate Reset	t_{RSTL}		15	—	—	µs
Missing Clock Detector Response Time (final rising edge to reset)	t_{MCD}	$F_{SYSCLK} > 1$ MHz	—	0.625	1.2	ms
Missing Clock Detector Trigger Frequency	F_{MCD}		—	7.5	13.5	kHz
V_{DD} Supply Monitor Turn-On Time	t_{MON}		—	2	—	µs

Table 1.4. Flash Memory

Parameter	Symbol	Test Condition	Min	Typ	Max	Units
Write Time[1,2]	t_{WRITE}	One Byte, F_{SYSCLK} = 24.5 MHz	19	20	21	µs
Erase Time[1,2]	t_{ERASE}	One Page, F_{SYSCLK} = 24.5 MHz	5.2	5.35	5.5	ms
V_{DD} Voltage During Programming[3]	V_{PROG}		2.2	—	3.6	V
Endurance (Write/Erase Cycles)	N_{WE}		20k	100k	—	Cycles

Notes:

1. Does not include sequencing time before and after the write/erase operation, which may be multiple SYSCLK cycles.
2. The internal High-Frequency Oscillator has a programmable output frequency using the OSCICL register, which is factory programmed to 24.5 MHz. If user firmware adjusts the oscillator speed, it must be between 22 and 25 MHz during any flash write or erase operation. It is recommended to write the OSCICL register back to its reset value when writing or erasing flash.
3. Flash can be safely programmed at any voltage above the supply monitor threshold (V_{VDDM}).
4. Data Retention Information is published in the Quarterly Quality and Reliability Report.

2.7. Reset Sources

Reset circuitry allows the controller to be easily placed in a predefined default condition. On entry to this reset state, the following occur:

- The core halts program execution.
- Module registers are initialized to their defined reset values unless the bits reset only with a power-on reset.
- External port pins are forced to a known state.
- Interrupts and timers are disabled.

All registers are reset to the predefined values noted in the register descriptions unless the bits only reset with a power-on reset. The contents of RAM are unaffected during a reset; any previously stored data is preserved as long as power is not lost.

The Port I/O latches are reset to 1 in open-drain mode. Weak pullups are enabled during and after the reset. For VDD Supply Monitor and power-on resets, the $\overline{\text{RST}}$ pin is driven low until the device exits the reset state.

On exit from the reset state, the program counter (PC) is reset, and the system clock defaults to the internal low-power oscillator. The Watchdog Timer is enabled with the Low Frequency Oscillator (LFO0) as its clock source. Program execution begins at location 0x0000.

2.8. On-Chip Debugging

The C8051F85x/86x devices include an on-chip Silicon Labs 2-Wire (C2) debug interface to allow flash programming and in-system debugging with the production part installed in the end application. The C2 interface uses a clock signal (C2CK) and a bi-directional C2 data signal (C2D) to transfer information between the device and a host system. See the C2 Interface Specification for details on the C2 protocol.

Table 3.1. Pin Definitions for C8051F850/1/2/3/4/5-GU and C8051F850/1/2/3/4/5-IU

Pin Name	Type	Pin Numbers	Crossbar Capability	Additional Digital Functions	Analog Functions
N/C	No Connection	1 13 24			

Table 3.2. Pin Definitions for C8051F850/1/2/3/4/5-GM and C8051F850/1/2/3/4/5-IM

Pin Name	Type	Pin Numbers	Crossbar Capability	Additional Digital Functions	Analog Functions
P1.0	Standard I/O	14	Yes	P1MAT.0	ADC0.8 CP1P.0 CP1N.0
P1.1	Standard I/O	13	Yes	P1MAT.1	ADC0.9 CP1P.1 CP1N.1
P1.2	Standard I/O	11	Yes	P1MAT.2	ADC0.10 CP1P.2 CP1N.2
P1.3	Standard I/O	10	Yes	P1MAT.3	ADC0.11 CP1P.3 CP1N.3
P1.4	Standard I/O	9	Yes	P1MAT.4	ADC0.12 CP1P.4 CP1N.4
P1.5	Standard I/O	8	Yes	P1MAT.5	ADC0.13 CP1P.5 CP1N.5
P1.6	Standard I/O	7	Yes	P1MAT.6	ADC0.14 CP1P.6 CP1N.6
P2.0 / C2D	Standard I/O / C2 Debug Data	6			

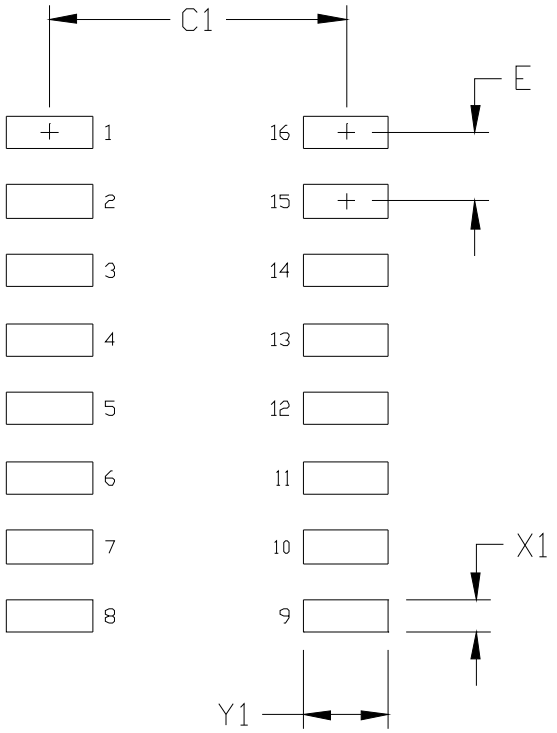

Figure 7.2. SOIC-16 PCB Land Pattern

Table 7.2. SOIC-16 PCB Land Pattern Dimensions

Dimension	Feature	(mm)
C1	Pad Column Spacing	5.40
E	Pad Row Pitch	1.27
X1	Pad Width	0.60
Y1	Pad Length	1.55

Notes:
General

1. All dimensions shown are in millimeters (mm) unless otherwise noted.
2. This Land Pattern Design is based on IPC-7351 pattern SOIC127P600X165-16N for Density Level B (Median Land Protrusion).
3. All feature sizes shown are at Maximum Material Condition (MMC) and a card fabrication tolerance of 0.05 mm is assumed.

11. Device Identification and Unique Identifier

The C8051F85x/86x has SFRs that identify the device family, derivative, and revision. These SFRs can be read by firmware at runtime to determine the capabilities of the MCU that is executing code. This allows the same firmware image to run on MCUs with different memory sizes and peripherals, and dynamically change functionality to suit the capabilities of that MCU.

In addition to the device identification registers, a 32-bit unique identifier (UID) is pre-programmed into all Revision C and later devices. The UID resides in the last four bytes of XRAM (C8051F850/1/3/4 and C8051F860/1/3/4) or RAM (C8051F852/5 and C8051F862/5). For devices with the UID in RAM, the UID can be read by firmware using indirect data accesses. For devices with the UID in XRAM, the UID can be read by firmware using MOVX instructions. The UID can also be read through the debug port for all devices.

Firmware can overwrite the UID during normal operation, and the bytes in memory will be automatically reinitialized with the UID value after any device reset. Firmware using this area of memory should always initialize the memory to a known value, as any previous data stored at these locations will be overwritten and not retained through a reset.

Table 11.1. UID Implementation Information

Device	Memory Segment	Addresses
C8051F850 C8051F851 C8051F853 C8051F854 C8051F860 C8051F861 C8051F863 C8051F864	XRAM	**(MSB)** 0x00FF, 0x00FE, 0x00FD, 0x00FC **(LSB)**
C8051F852 C8051F855 C8051F862 C8051F865	RAM (indirect)	**(MSB)** 0xFF, 0xFE, 0xFD, 0xFC **(LSB)**

cycles to complete the DIV instruction and 4 clock cycles to execute the LCALL to the ISR. If the CPU is executing an ISR for an interrupt with equal or higher priority, the new interrupt will not be serviced until the current ISR completes, including the RETI and following instruction. If more than one interrupt is pending when the CPU exits an ISR, the CPU will service the next highest priority interrupt that is pending.

13. Power Management and Internal Regulator

All internal circuitry on the C8051F85x/86x devices draws power from the VDD supply pin. Circuits with external connections (I/O pins, analog muxes) are powered directly from the VDD supply voltage, while most of the internal circuitry is supplied by an on-chip LDO regulator. The regulator output is fully internal to the device, and is available also as an ADC input or reference source for the comparators and ADC.

The devices support the standard 8051 power modes: idle and stop. For further power savings in stop mode, the internal LDO regulator may be disabled, shutting down the majority of the power nets on the device.

Although the C8051F85x/86x has idle and stop modes available, more control over the device power can be achieved by enabling/disabling individual peripherals as needed. Each analog peripheral can be disabled when not in use and placed in low power mode. Digital peripherals, such as timers and serial buses, have their clocks gated off and draw little power when they are not in use.

13.1. Power Modes

Idle mode halts the CPU while leaving the peripherals and clocks active. In stop mode, the CPU is halted, all interrupts and timers are inactive, and the internal oscillator is stopped (analog peripherals remain in their selected states; the external oscillator is not affected). Since clocks are running in Idle mode, power consumption is dependent upon the system clock frequency and the number of peripherals left in active mode before entering Idle. Stop mode consumes the least power because the majority of the device is shut down with no clocks active. The Power Control Register (PCON) is used to control the C8051F85x/86x's Stop and Idle power management modes.

13.1.1. Idle Mode

Setting the Idle Mode Select bit (PCON.0) causes the hardware to halt the CPU and enter idle mode as soon as the instruction that sets the bit completes execution. All internal registers and memory maintain their original data. All analog and digital peripherals can remain active during idle mode.

Idle mode is terminated when an enabled interrupt is asserted or a reset occurs. The assertion of an enabled interrupt will cause the Idle Mode Selection bit (PCON.0) to be cleared and the CPU to resume operation. The pending interrupt will be serviced and the next instruction to be executed after the return from interrupt (RETI) will be the instruction immediately following the one that set the Idle Mode Select bit. If idle mode is terminated by an internal or external reset, the CIP-51 performs a normal reset sequence and begins program execution at address 0x0000.

Note: If the instruction following the write of the IDLE bit is a single-byte instruction and an interrupt occurs during the execution phase of the instruction that sets the IDLE bit, the CPU may not wake from Idle mode when a future interrupt occurs. Therefore, instructions that set the IDLE bit should be followed by an instruction that has two or more opcode bytes, for example:

```
// in 'C':
PCON |= 0x01;                  // set IDLE bit
PCON = PCON;                   // ... followed by a 3-cycle dummy instruction

; in assembly:
ORL PCON, #01h                 ; set IDLE bit
MOV PCON, PCON                 ; ... followed by a 3-cycle dummy instruction
```

If enabled, the Watchdog Timer (WDT) will eventually cause an internal watchdog reset and thereby terminate the idle mode. This feature protects the system from an unintended permanent shutdown in the event of an inadvertent write to the PCON register. If this behavior is not desired, the WDT may be disabled by software prior to entering the Idle mode if the WDT was initially configured to allow this operation. This provides the opportunity for additional power savings, allowing the system to remain in the Idle mode indefinitely, waiting for an external stimulus to wake up the system.

14.2. ADC Operation

The ADC is clocked by an adjustable conversion clock (SARCLK). SARCLK is a divided version of the selected system clock when burst mode is disabled (ADBMEN = 0), or a divided version of the high-frequency oscillator when burst mode is enabled (ADBMEN = 1). The clock divide value is determined by the ADSC bits in the ADC0CF register. In most applications, SARCLK should be adjusted to operate as fast as possible, without exceeding the maximum electrical specifications. The SARCLK does not directly determine sampling times or sampling rates.

14.2.1. Starting a Conversion

A conversion can be initiated in many ways, depending on the programmed states of the ADC0 Start of Conversion Mode field (ADCM) in register ADC0CN0. Conversions may be initiated by one of the following:

1. Writing a 1 to the ADBUSY bit of register ADC0CN0 (software-triggered)
2. A timer overflow (see the ADC0CN0 register and the timer section for timer options)
3. A rising edge on the CNVSTR input signal (external pin-triggered)

Writing a 1 to ADBUSY provides software control of ADC0 whereby conversions are performed "on-demand". All other trigger sources occur autonomous to code execution. When the conversion is complete, the ADC posts the result to its output register and sets the ADC interrupt flag (ADINT). ADINT may be used to trigger a system interrupts, if enabled, or polled by firmware.

During conversion, the ADBUSY bit is set to logic 1 and reset to logic 0 when the conversion is complete. However, when polling for ADC conversion completions, the ADC0 interrupt flag (ADINT) should be used instead of the ADBUSY bit. Converted data is available in the ADC0 data registers, ADC0H:ADC0L, when the conversion is complete.

Important Note About Using CNVSTR: When the CNVSTR input is used as the ADC0 conversion source, the associated port pin should be skipped in the crossbar settings.

14.2.2. Tracking Modes

Each ADC0 conversion must be preceded by a minimum tracking time in order for the converted result to be accurate. The minimum tracking time is given in the electrical specifications tables. The ADTM bit in register ADC0CN0 controls the ADC0 track-and-hold mode. In its default state when Burst Mode is disabled, the ADC0 input is continuously tracked, except when a conversion is in progress. A conversion will begin immediately when the start-of-conversion trigger occurs.

When the ADTM bit is logic 1, each conversion is preceded by a tracking period of 4 SAR clocks (after the start-of-conversion signal) for any internal (non-CNVSTR) conversion trigger source. When the CNVSTR signal is used to initiate conversions with ADTM set to 1, ADC0 tracks only when CNVSTR is low; conversion begins on the rising edge of CNVSTR (see Figure 14.2). Setting ADTM to 1 is primarily useful when AMUX settings are frequently changed and conversions are started using the ADBUSY bit.

Register 14.8. ADC0L: ADC0 Data Word Low Byte

Bit	7	6	5	4	3	2	1	0
Name	ADC0L							
Type	RW							
Reset	0	0	0	0	0	0	0	0
SFR Address: 0xBD								

Table 14.11. ADC0L Register Bit Descriptions

Bit	Name	Function
7:0	ADC0L	**Data Word Low Byte.** When read, this register returns the least significant byte of the 16-bit ADC0 accumulator, formatted according to the settings in ADSJST. The register may also be written, to set the lower byte of the 16-bit ADC0 accumulator.
Note: If Accumulator shifting is enabled, the most significant bits of the value read will be zeros. This register should not be written when the SYNC bit is set to 1.		

17. Comparators (CMP0 and CMP1)

C8051F85x/86x devices include two on-chip programmable voltage comparators, CMP0 and CMP1. The two comparators are functionally identical, but have different connectivity within the device. A functional block diagram is shown in Figure 17.1.

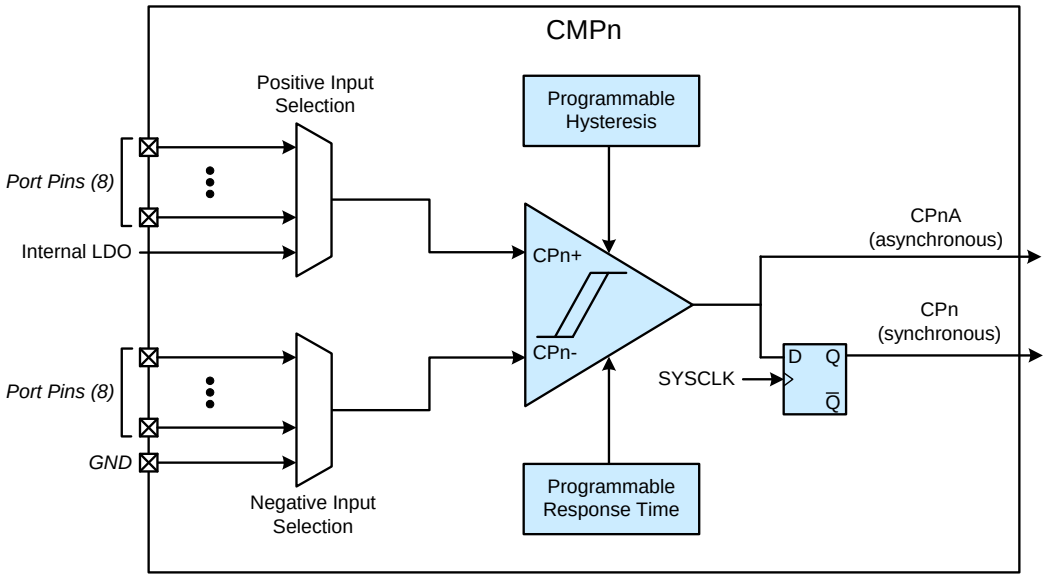

Figure 17.1. Comparator Functional Block Diagram

17.1. System Connectivity

Comparator inputs are routed to port I/O pins or internal signals using the comparator mux registers. The comparator's synchronous and asynchronous outputs can optionally be routed to port I/O pins through the port I/O crossbar. The output of either comparator may also be configured to generate a system interrupt. CMP0 may also be used as a reset source, or as a trigger to kill a PCA output channel.

The CMP0 inputs are selected in the CPT0MX register, while CPT1MX selects the CMP1 inputs. The CMXP field selects the comparator's positive input (CPnP.x); the CMXN field selects the comparator's negative input (CPnN.x). Table 17.1 through Table 17.4 detail the comparator input multiplexer options on the C8051F85x/86x family. See the port I/O crossbar sections for details on configuring comparator outputs via the digital crossbar. Comparator inputs can be externally driven from –0.25 V to (V_{DD}) + 0.25 V without damage or upset.

Important Note About Comparator Inputs: The port pins selected as comparator inputs should be configured as analog inputs in their associated port configuration register, and configured to be skipped by the crossbar.

Register 20.13. PCA0CPM2: PCA Capture/Compare Mode

Bit	7	6	5	4	3	2	1	0
Name	PWM16	ECOM	CAPP	CAPN	MAT	TOG	PWM	ECCF
Type	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0
SFR Address: 0xDC								

Table 20.15. PCA0CPM2 Register Bit Descriptions

Bit	Name	Function
7	PWM16	**16-bit Pulse Width Modulation Enable.** This bit enables 16-bit mode when Pulse Width Modulation mode is enabled. 0: 8 to 11-bit PWM selected. 1: 16-bit PWM selected.
6	ECOM	**Comparator Function Enable.** This bit enables the comparator function.
5	CAPP	**Capture Positive Function Enable.** This bit enables the positive edge capture capability.
4	CAPN	**Capture Negative Function Enable.** This bit enables the negative edge capture capability.
3	MAT	**Match Function Enable.** This bit enables the match function. When enabled, matches of the PCA counter with a module's capture/compare register cause the CCF2 bit in the PCA0MD register to be set to logic 1.
2	TOG	**Toggle Function Enable.** This bit enables the toggle function. When enabled, matches of the PCA counter with the capture/compare register cause the logic level on the CEX2 pin to toggle. If the PWM bit is also set to logic 1, the module operates in Frequency Output Mode.
1	PWM	**Pulse Width Modulation Mode Enable.** This bit enables the PWM function. When enabled, a pulse width modulated signal is output on the CEX2 pin. 8 to 11-bit PWM is used if PWM16 is cleared; 16-bit mode is used if PWM16 is set to logic 1. If the TOG bit is also set, the module operates in Frequency Output Mode.
0	ECCF	**Capture/Compare Flag Interrupt Enable.** This bit sets the masking of the Capture/Compare Flag (CCF2) interrupt. 0: Disable CCF2 interrupts. 1: Enable a Capture/Compare Flag interrupt request when CCF2 is set.

Register 20.15. PCA0CPH1: PCA Capture Module High Byte

Bit	7	6	5	4	3	2	1	0
Name	PCA0CPH1							
Type	RW							
Reset	0	0	0	0	0	0	0	0
SFR Address: 0xEA								

Table 20.17. PCA0CPH1 Register Bit Descriptions

Bit	Name	Function
7:0	PCA0CPH1	**PCA Capture Module High Byte.** The PCA0CPH1 register holds the high byte (MSB) of the 16-bit capture module.This register address also allows access to the high byte of the corresponding PCA channels auto-reload value for 9 to 11-bit PWM mode. The ARSEL bit in register PCA0PWM controls which register is accessed.
Note: A write to this register will set the modules ECOM bit to a 1.		

Register 21.11. P1MASK: Port 1 Mask

Bit	7	6	5	4	3	2	1	0
Name	P1MASK							
Type	RW							
Reset	0	0	0	0	0	0	0	0
SFR Address: 0xEE								

Table 21.14. P1MASK Register Bit Descriptions

Bit	Name	Function
7:0	P1MASK	**Port 1 Mask Value.** Selects P1 pins to be compared to the corresponding bits in P1MAT. 0: P1.x pin logic value is ignored and will cause a port mismatch event. 1: P1.x pin logic value is compared to P1MAT.x.
Note: Port 1 consists of 8 bits (P1.0-P1.7) on QSOP24 packages and 7 bits (P1.0-P1.6) on QFN20 packages and 4 bits (P1.0-P1.3) on SOIC16 packages.		

25.2.2. 8-bit Timers with Auto-Reload

When TnSPLIT is set, the timer operates as two 8-bit timers (TMRnH and TMRnL). Both 8-bit timers operate in auto-reload mode as shown in Figure 25.5. TMRnRLL holds the reload value for TMRnL; TMRnRLH holds the reload value for TMRnH. The TRn bit in TMRnCN handles the run control for TMRnH. TMRnL is always running when configured for 8-bit auto-reload mode.

Each 8-bit timer may be configured to clock from SYSCLK, SYSCLK divided by 12, or the external oscillator clock source divided by 8. The Clock Select bits (TnMH and TnML in CKCON) select either SYSCLK or the clock defined by the External Clock Select bit (TnXCLK in TMRnCN), as follows:

TnMH	TnXCLK	TMRnH Clock Source
0	0	SYSCLK / 12
0	1	External Clock / 8
1	X	SYSCLK

TnML	TnXCLK	TMRnL Clock Source
0	0	SYSCLK / 12
0	1	External Clock / 8
1	X	SYSCLK

The TFnH bit is set when TMRnH overflows from 0xFF to 0x00; the TFnL bit is set when TMRnL overflows from 0xFF to 0x00. When timer interrupts are enabled, an interrupt is generated each time TMRnH overflows. If timer interrupts are enabled and TFnLEN is set, an interrupt is generated each time either TMRnL or TMRnH overflows. When TFnLEN is enabled, software must check the TFnH and TFnL flags to determine the source of the timer interrupt. The TFnH and TFnL interrupt flags are not cleared by hardware and must be manually cleared by software.

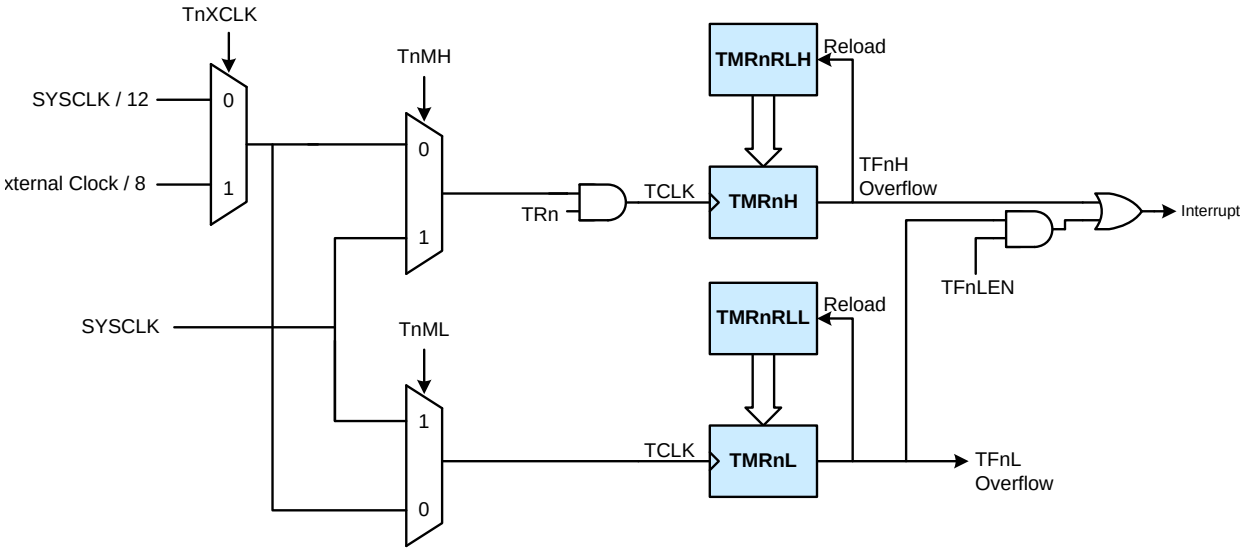

Figure 25.5. 8-Bit Mode Block Diagram

Register 25.8. TMR2CN: Timer 2 Control

Bit	7	6	5	4	3	2	1	0
Name	TF2H	TF2L	TF2LEN	TF2CEN	T2SPLIT	TR2	Reserved	T2XCLK
Type	RW	RW	RW	RW	RW	RW	R	RW
Reset	0	0	0	0	0	0	0	0
SFR Address: 0xC8 (bit-addressable)								

Table 25.10. TMR2CN Register Bit Descriptions

Bit	Name	Function
7	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. This bit is not automatically cleared by hardware.
6	TF2L	**Timer 2 Low Byte Overflow Flag.** Set by hardware when the Timer 2 low byte overflows from 0xFF to 0x00. TF2L will be set when the low byte overflows regardless of the Timer 2 mode. This bit is not automatically cleared by hardware.
5	TF2LEN	**Timer 2 Low Byte Interrupt Enable.** When set to 1, this bit enables Timer 2 Low Byte interrupts. If Timer 2 interrupts are also enabled, an interrupt will be generated when the low byte of Timer 2 overflows.
4	TF2CEN	**Timer 2 Capture Enable.** When set to 1, this bit enables Timer 2 Capture Mode. If TF2CEN is set and Timer 2 interrupts are enabled, an interrupt will be generated on a falling edge of the selected T2 input pin, and the current 16-bit timer value in TMR2H:TMR2L will be copied to TMR2RLH:TMR2RLL.
3	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.
2	TR2	**Timer 2 Run Control.** Timer 2 is enabled by setting this bit to 1. In 8-bit mode, this bit enables/disables TMR2H only; TMR2L is always enabled in split mode.
1	Reserved	Must write reset value.

Register 25.11. TMR2L: Timer 2 Low Byte

Bit	7	6	5	4	3	2	1	0
Name	TMR2L							
Type	RW							
Reset	0	0	0	0	0	0	0	0
SFR Address: 0xCC								

Table 25.13. TMR2L Register Bit Descriptions

Bit	Name	Function
7:0	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.

26.3. 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 MCE bit 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 (RB8 = 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 MCE bit to enable interrupts on the reception of the following data byte(s). Slaves that weren't addressed leave their MCE 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 MCE 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).

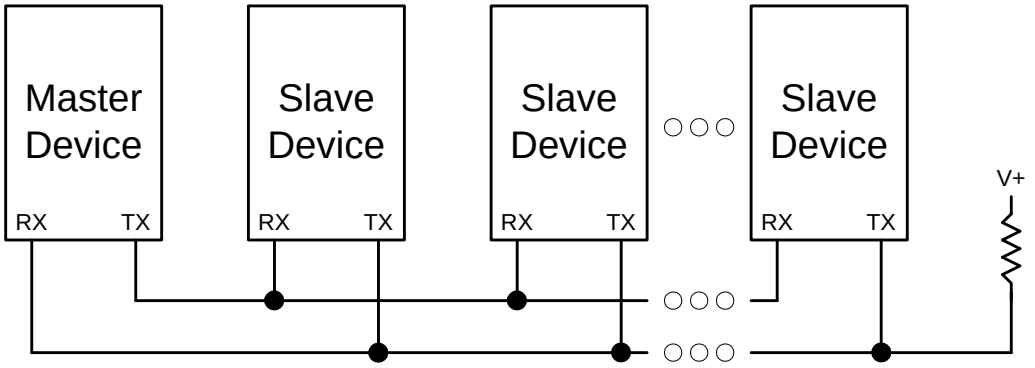

Figure 26.5. UART Multi-Processor Mode Interconnect Diagram

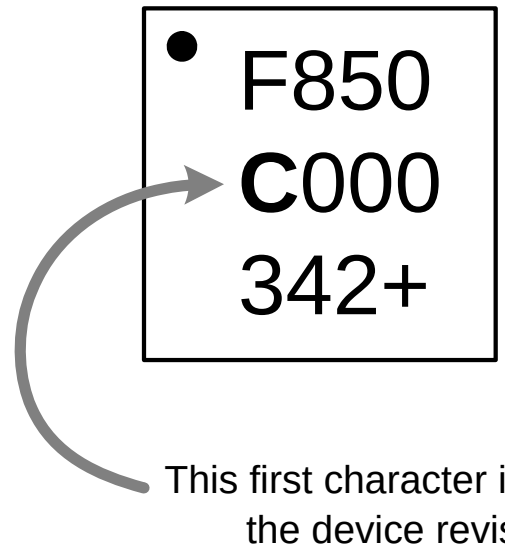

Figure 28.2. QFN-20 Package Revision Marking

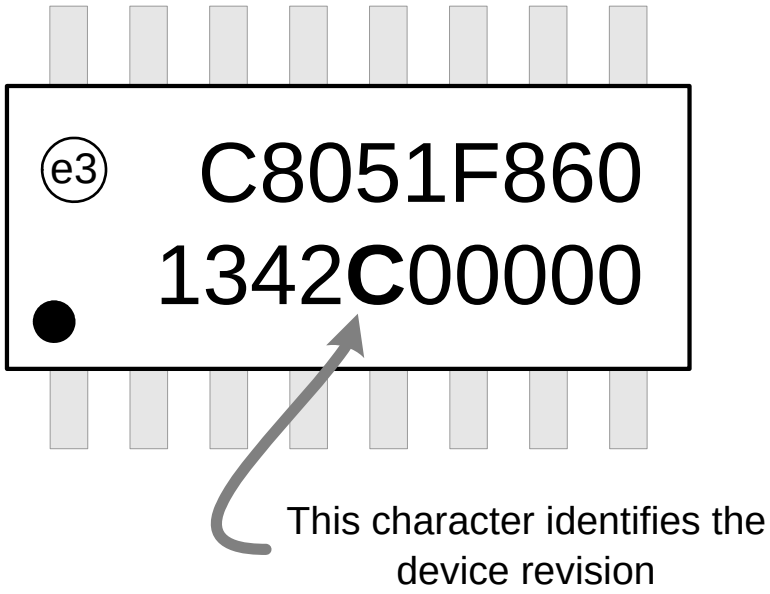

Figure 28.3. SOIC-16 Package Revision Marking