

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 "<u>Embedded -</u> <u>Microcontrollers</u>"

Details

Product Status	Active
Core Processor	80C51
Core Size	8-Bit
Speed	48MHz
Connectivity	I ² C, SPI, UART/USART, USB
Peripherals	LED, POR, PWM, WDT
Number of I/O	18
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	4K x 8
RAM Size	1.25K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 5.5V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	32-VQFN Exposed Pad
Supplier Device Package	32-QFN (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/at89c5131a-putum

Email: info@E-XFL.COM

Address: Room A, 16/F, Full Win Commercial Centre, 573 Nathan Road, Mongkok, Hong Kong

Signal Name	Туре	Description	Alternate Function
CEX[4:0]	I/O	Capture External Input Compare External Output	P1.3 P1.4 P1.5 P1.6 P1.7

Table 3-3. Serial I/O Signal Description

Signal Name	Туре	Description	Alternate Function
RxD	I	Serial Input Port	P3.0
TxD	0	Serial Output Port	P3.1

Table 3-4. Timer 0, Timer 1 and Timer 2 Signal Description

Signal Name	Туре	Description	Alternate Function
INTO	I	Timer 0 Gate InputINT0 serves as external run control for timer 0, when selected by GATE0bit in TCON register.External Interrupt 0INT0 input set IE0 in the TCON register. If bit IT0 in this register is set, bitsIE0 are set by a falling edge on INT0. If bit IT0 is cleared, bits IE0 is set by a low level on INT0.	P3.2
INT1	I	Timer 1 Gate InputINT1 serves as external run control for Timer 1, when selected by GATE1bit in TCON register.External Interrupt 1INT1 input set IE1 in the TCON register. If bit IT1 in this register is set, bitsIE1 are set by a falling edge on INT1. If bit IT1 is cleared, bits IE1 is set by a low level on INT1.	P3.3
то	I	Timer Counter 0 External Clock Input When Timer 0 operates as a counter, a falling edge on the T0 pin increments the count.	P3.4
T1	I	Timer/Counter 1 External Clock Input When Timer 1 operates as a counter, a falling edge on the T1 pin increments the count.	P3.5
T2	 0	Timer/Counter 2 External Clock Input Timer/Counter 2 Clock Output	P1.0
T2EX	Ι	Timer/Counter 2 Reload/Capture/Direction Control Input	P1.1





Table 3-5.LED Signal Description

Signal Name	Туре	Description	Alternate Function
LED[3:0]	0	Direct Drive LED Output These pins can be directly connected to the Cathode of standard LEDs without external current limiting resistors. The typical current of each output can be programmed by software to 2, 6 or 10 mA. Several outputs can be connected together to get higher drive capabilities.	P3.3 P3.5 P3.6 P3.7

Table 3-6.TWI Signal Description

Signal Name	Туре	Description	Alternate Function
SCL	I/O	SCL: TWI Serial Clock SCL output the serial clock to slave peripherals. SCL input the serial clock from master.	P4.0
SDA	I/O	SDA: TWI Serial Data SCL is the bidirectional TWI data line.	P4.1

Table 3-7.SPI Signal Description

Signal Name	Туре	Description	Alternate Function
SS	I/O	SS: SPI Slave Select	P1.1
MISO	I/O	MISO: SPI Master Input Slave Output line When SPI is in master mode, MISO receives data from the slave peripheral. When SPI is in slave mode, MISO outputs data to the master controller.	P1.5
SCK	I/O	SCK: SPI Serial Clock SCK outputs clock to the slave peripheral or receive clock from the master	P1.6
MOSI	I/O	MOSI: SPI Master Output Slave Input line When SPI is in master mode, MOSI outputs data to the slave peripheral. When SPI is in slave mode, MOSI receives data from the master controller	P1.7

4.2 PCB Recommandations

Figure 4-2. USB Pads



Isolate filter components with a ground wire

close to the microcontroller



Oscillator Frequency	R+1	N+1	PLLDIV
32 MHz	3	2	21h
40 MHz	12	10	B9h

5.4 Registers

Table 5-2.CKCON0 (S:8Fh)Clock Control Register 0

7	6	5	4	3	2	1	0	
TWIX2	WDX2	PCAX2	SIX2	T2X2	T1X2	T0X2	X2	
Bit Number	Bit Mnemonic	Description						
7	TWIX2	TWI Clock This control this bit has Clear to seled Set to select	bit is validat no effect. ct 6 clock perio 12 clock perio	ted when the iods per peripl ods per periph	CPU clock X neral clock cyc eral clock cyc	2 is set. Whe cle. le.	n X2 is low,	
6	WDX2	Watchdog C This control this bit has Clear to select Set to select	lock bit is validat no effect. ct 6 clock peri 12 clock perio	ted when the iods per peripl ods per periph	CPU clock X neral clock cyc eral clock cyc	2 is set. Whe cle. le.	n X2 is low,	
5	PCAX2	Programmal This control this bit has Clear to select Set to select	ble Counter A bit is validat no effect. ct 6 clock perion 12 clock perion	Array Clock ted when the iods per peripl ods per periph	CPU clock X neral clock cyc eral clock cyc	2 is set. Whe cle. le.	n X2 is low,	
4	SIX2	Enhanced U This control this bit has Clear to select Set to select	ART Clock (I bit is validat no effect. ct 6 clock peri 12 clock perio	Mode 0 and 2 ted when the iods per peripl ods per periph) CPU clock X neral clock cyc eral clock cyc	2 is set. Whe cle. le.	n X2 is low,	
3	T2X2	Timer2 Cloc This control this bit has Clear to select Set to select	k bit is validat no effect. ct 6 clock peri 12 clock perio	ted when the iods per peripl ods per periph	CPU clock X neral clock cyc eral clock cyc	2 is set. Whe cle. le.	n X2 is low,	
2	T1X2	Timer1 Clock This control bit is validated when the CPU clock X2 is set. When X2 is low, this bit has no effect. Clear to select 6 clock periods per peripheral clock cycle. Set to select 12 clock periods per peripheral clock cycle.						
1	T0X2	Timer0 Clock This control bit is validated when the CPU clock X2 is set. When X2 is low, this bit has no effect. Clear to select 6 clock periods per peripheral clock cycle. Set to select 12 clock periods per peripheral clock cycle.						
0	X2	System Cloc Clear to selec F _{OSC} /2). Set to select	ck Control bit ct 12 clock pe 6 clock period	t priods per mac ds per machin	hine cycle (S [−] e cycle (X2 m	ΓD mode, F _{CP} ode, F _{CPU =} F _F	_U = F _{PER =} P _{ER =} F _{OSC}).	



Bit Number	Bit Mnemonic	Description
7-4	R3:0	PLL R Divider Bits
3-0	N3:0	PLL N Divider Bits

Reset Value = 0000 0000





6. SFR Mapping

The Special Function Registers (SFRs) of the AT89C5130A/31A-M fall into the following categories:

- C51 core registers: ACC, B, DPH, DPL, PSW, SP
- I/O port registers: P0, P1, P2, P3, P4
- Timer registers: T2CON, T2MOD, TCON, TH0, TH1, TH2, TMOD, TL0, TL1, TL2, RCAP2L, RCAP2H
- Serial I/O port registers: SADDR, SADEN, SBUF, SCON
- PCA (Programmable Counter Array) registers: CCON, CMOD, CCAPMx, CL, CH, CCAPxH, CCAPxL (x: 0 to 4)
- Power and clock control registers: PCON
- Hardware Watchdog Timer registers: WDTRST, WDTPRG
- Interrupt system registers: IEN0, IPL0, IPH0, IEN1, IPL1, IPH1
- Keyboard Interface registers: KBE, KBF, KBLS
- LED register: LEDCON
- Two Wire Interface (TWI) registers: SSCON, SSCS, SSDAT, SSADR
- Serial Port Interface (SPI) registers: SPCON, SPSTA, SPDAT
- USB registers: Uxxx (17 registers)
- PLL registers: PLLCON, PLLDIV
- BRG (Baud Rate Generator) registers: BRL, BDRCON
- Flash register: FCON (FCON access is reserved for the Flash API and ISP software)
- EEPROM register: EECON
- Others: AUXR, AUXR1, CKCON0, CKCON1

Bit Number	Bit Mnemonic	Description
1	PWMn	Pulse Width Modulation Mode Cleared to disable the CEXn pin to be used as a pulse width modulated output. Set to enable the CEXn pin to be used as a pulse width modulated output.
0	ECCFn	Enable CCF Interrupt Cleared to disable compare/capture flag CCFn in the CCON register to generate an interrupt. Set to enable compare/capture flag CCFn in the CCON register to generate an interrupt.

Reset Value = X000 0000b

Not bit addressable

ECOMn	CAPPn	CAPNn	MATn	TOGn	PWMm	ECCFn	Module Function
0	0	0	0	0	0	0	No Operation
x	1	0	0	0	0	х	16-bit capture by a positive-edge trigger on CEXn
х	0	1	0	0	0	х	16-bit capture by a negative trigger on CEXn
x	1	1	0	0	0	х	16-bit capture by a transition on CEXn
1	0	0	1	0	0	х	16-bit Software Timer/Compare mode.
1	0	0	1	1	0	х	16-bit High Speed Output
1	0	0	0	0	1	0	8-bit PWM
1	0	0	1	Х	0	Х	Watchdog Timer (module 4 only)

Table 14-4. PCA Module Modes (CCAPMn Registers)

There are two additional registers associated with each of the PCA modules. They are CCAPnH and CCAPnL and these are the registers that store the 16-bit count when a capture occurs or a compare should occur. When a module is used in the PWM mode these registers are used to control the duty cycle of the output (see Table 14-5 and Table 14-6)

Table 14-5.CCAPnH Registers (n = 0-4)

CCAP0H - PCA Module 0 Compare/Capture Control Register High (0FAh)

CCAP1H - PCA Module 1 Compare/Capture Control Register High (0FBh)

- CCAP2H PCA Module 2 Compare/Capture Control Register High (0FCh)
- CCAP3H PCA Module 3 Compare/Capture Control Register High (0FDh)

CCAP4H - PCA Module 4 Compare/Capture Control Register High (0FEh)

7	6	5	4	3	2	1	0			
-	-	-	-	-	-	-	-			
Bit Number	Bit Mnemonic	Description	Description							
7 - 0	-	PCA Modul CCAPnH Va	e n Compare/(Ilue	Capture Contro	bl					



Figure 14-5. PCA High-speed Output Mode



Before enabling ECOM bit, CCAPnL and CCAPnH should be set with a non zero value, otherwise an unwanted match could happen.

Once ECOM set, writing CCAPnL will clear ECOM so that an unwanted match doesn't occur while modifying the compare value. Writing to CCAPnH will set ECOM. For this reason, user software should write CCAPnL first, and then CCAPnH. Of course, the ECOM bit can still be controlled by accessing to CCAPMn register.

14.4 Pulse Width Modulator Mode

All of the PCA modules can be used as PWM outputs. Figure 14-6 shows the PWM function. The frequency of the output depends on the source for the PCA timer. All of the modules will have the same frequency of output because they all share the PCA timer. The duty cycle of each module is independently variable using the module's capture register CCAPLn. When the value of the PCA CL SFR is less than the value in the module's CCAPLn SFR the output will be low, when it is equal to or greater than the output will be high. When CL overflows from FF to 00, CCAPLn is reloaded with the value in CCAPHn. This allows updating the PWM without glitches. The PWM and ECOM bits in the module's CCAPMn register must be set to enable the PWM mode.





SADEN - Slave Address Mask Register (B9h)

7	6	5	4	3	2	1	0

Reset Value = 0000 0000b Not bit addressable

SADDR - Slave Address Register (A9h)

7	6	5	4	3	2	1	0

Reset Value = 0000 0000b Not bit addressable

15.3 Baud Rate Selection for UART for Mode 1 and 3

The Baud Rate Generator for transmit and receive clocks can be selected separately via the T2CON and BDRCON registers.

Figure 15-4. Baud Rate Selection



Baud Bates	F _{osc} = 16	.384 MHz	F _{OSC} = 24 MHz		
Budu Hates	BRL Error (%)		BRL	Error (%)	
115200	247	1.23	243	0.16	
57600	238	1.23	230	0.16	
38400	229	1.23	217	0.16	
28800	220	1.23	204	0.16	
19200	203	0.63	178	0.16	
9600	149	0.31	100	0.16	
4800	43	1.23	-	-	

Example of computed value when X2 = 1, SMOD1 = 1, SPD = 1

Example of computed value when X2 = 0, SMOD1 = 0, SPD = 0

	F _{osc} = 16	.384 MHz	F _{osc} = 24 MHz		
Baud Rates	BRL Error (%)		BRL	Error (%)	
4800	247	1.23	243	0.16	
2400	238	1.23	230	0.16	
1200	220	1.23	202	3.55	
600	185	0.16	152	0.16	

The baud rate generator can be used for mode 1 or 3 (refer to Figure 15-4.), but also for mode 0 for UART, thanks to the bit SRC located in BDRCON register (Table 15-4.)

15.4 UART Registers

SADEN - Slave Address Mask Register for UART (B9h)

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-

Reset Value = 0000 0000b

SADDR - Slave Address Register for UART (A9h)

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-

Reset Value = 0000 0000b

SBUF - Serial Buffer Register for UART (99h)



Reset Value = XXXX XXXXb



AT89C5130A/31A-M

16.3 Interrupt Sources and Vector Addresses

Table 16-8.Vector Table

Number	Polling Priority	Interrupt Source	Interrupt Request	Vector Address
0	0	Reset		0000h
1	1	INT0	IE0	0003h
2	2	Timer 0	TF0	000Bh
3	3	INT1	IE1	0013h
4	4	Timer 1	IF1	001Bh
5	6	UART	RI+TI	0023h
6	7	Timer 2	TF2+EXF2	002Bh
7	5	PCA	CF + CCFn (n = 0-4)	0033h
8	8	Keyboard	KBDIT	003Bh
9	9	TWI	ТШІТ	0043h
10	10	SPI	SPIIT	004Bh
11	11			0053h
12	12			005Bh
13	13			0063h
14	14	USB	UEPINT + USBINT	006Bh
15	15			0073h





17. Keyboard Interface

17.1 Introduction

The AT89C5130A/31A-M implements a keyboard interface allowing the connection of a 8 x n matrix keyboard. It is based on 8 inputs with programmable interrupt capability on both high or low level. These inputs are available as an alternate function of P1 and allow to exit from idle and power down modes.

17.2 Description

The keyboard interface communicates with the C51 core through 3 special function registers: KBLS, the Keyboard Level Selection register (Table 17-3), KBE, The Keyboard interrupt Enable register (Table 17-2), and KBF, the Keyboard Flag register (Table 17-1).

17.2.1 Interrupt

The keyboard inputs are considered as 8 independent interrupt sources sharing the same interrupt vector. An interrupt enable bit (KBD in IE1) allows global enable or disable of the keyboard interrupt (see Figure 17-1). As detailed in Figure 17-2 each keyboard input has the capability to detect a programmable level according to KBLS.x bit value. Level detection is then reported in interrupt flags KBF.x that can be masked by software using KBE.x bits.

This structure allow keyboard arrangement from 1 by n to 8 by n matrix and allow usage of P1 inputs for other purpose.



Figure 17-1. Keyboard Interface Block Diagram







19.2.3 SPI Serial Clock (SCK)

This signal is used to synchronize the data movement both in and out the devices through their MOSI and MISO lines. It is driven by the Master for eight clock cycles which allows to exchange one byte on the serial lines.

19.2.4 Slave Select (SS)

Each Slave peripheral is selected by one Slave Select pin (\overline{SS}) . This signal must stay low for any message for a Slave. It is obvious that only one Master (\overline{SS} high level) can drive the network. The Master may select each Slave device by software through port pins (Figure 19-1). To prevent bus conflicts on the MISO line, only one slave should be selected at a time by the Master for a transmission.

In a Master configuration, the \overline{SS} line can be used in conjunction with the MODF flag in the SPI Status register (SPSTA) to prevent multiple masters from driving MOSI and SCK (see Section "Error Conditions", page 98).

A high level on the \overline{SS} pin puts the MISO line of a Slave SPI in a high-impedance state.

The SS pin could be used as a general-purpose if the following conditions are met:

- The device is configured as a Master and the SSDIS control bit in SPCON is set. This kind of configuration can be found when only one Master is driving the network and there is no way that the \overline{SS} pin could be pulled low. Therefore, the MODF flag in the SPSTA will never be set⁽¹⁾.
- The Device is configured as a Slave with CPHA and SSDIS control bits set⁽²⁾ This kind of configuration can happen when the system comprises one Master and one Slave only. Therefore, the device should always be selected and there is no reason that the Master uses the SS pin to select the communicating Slave device.
- Notes: 1. Clearing SSDIS control bit does not clear MODF.
 - 2. Special care should be taken not to set SSDIS control bit when CPHA ='0' because in this mode, the SS is used to start the transmission.

19.2.5 Baud Rate

In Master mode, the baud rate can be selected from a baud rate generator which is controlled by three bits in the SPCON register: SPR2, SPR1 and SPR0. The Master clock is chosen from one of seven clock rates resulting from the division of the internal clock by 4, 8, 16, 32, 64 or 128.

Table 19-1 gives the different clock rates selected by SPR2:SPR1:SPR0:

SPR2	SPR1	SPR0	Clock Rate	Baud Rate Divisor (BD)
0	0	0	Don't Use	No BRG
0	0	1	F _{CLK PERIPH} /4	4
0	1	0	F _{CLK PERIPH} /8	8
0	1	1	F _{CLK PERIPH} /16	16
1	0	0	F _{CLK PERIPH} /32	32
1	0	1	F _{CLK PERIPH} /64	64
1	1	0	F _{CLK PERIPH} /128	128

Table 19-1. SPI Master Baud Rate Selection

Figure 20-6. Format and State in the Slave Receiver Mode





21.4.4 Bulk/Interrupt IN Transactions in Ping-pong Mode



Figure 21-10. Bulk/Interrupt IN Transactions in Ping-pong Mode

An endpoint will be first enabled and configured before being able to send Bulk or Interrupt packets.

The firmware will fill the FIFO bank 0 with the data to be sent and set the TXRDY bit in the UEP-STAX register to allow the USB controller to send the data stored in FIFO at the next IN request concerning the endpoint. The FIFO banks are automatically switched, and the firmware can immediately write into the endpoint FIFO bank 1.

When the IN packet concerning the bank 0 has been sent and acknowledged by the Host, the TXCMPL bit is set by the USB controller. This triggers a USB interrupt if enabled. The firmware will clear the TXCMPL bit before filling the endpoint FIFO bank 0 with new data. The FIFO banks are then automatically switched.

When the IN packet concerning the bank 1 has been sent and acknowledged by the Host, the TXCMPL bit is set by the USB controller. This triggers a USB interrupt if enabled. The firmware will clear the TXCMPL bit before filling the endpoint FIFO bank 1 with new data.

The bank switch is performed by the USB controller each time the TXRDY bit is set by the firmware. Until the TXRDY bit has been set by the firmware for an endpoint bank, the USB controller will answer a NAK handshake for each IN requests concerning this bank.

Note that in the example above, the firmware clears the Transmit Complete bit (TXCMPL) before setting the Transmit Ready bit (TXRDY). This is done in order to avoid the firmware to clear at the same time the TXCMPL bit for bank 0 and the bank 1.





The firmware has to clear one of these two bits after having read all the data FIFO to allow a new packet to be stored in the corresponding bank.

If the Host sends more bytes than supported by the endpoint FIFO, the overflow data won't be stored, but the USB controller will consider that the packet is valid if the CRC is correct.

21.6.3 Isochronous IN Transactions in Standard Mode

An endpoint will be first enabled and configured before being able to send Isochronous packets.

The firmware will fill the FIFO with the data to be sent and set the TXRDY bit in the UEPSTAX register to allow the USB controller to send the data stored in FIFO at the next IN request concerning this endpoint.

If the TXRDY bit is not set when the IN request occurs, nothing will be sent by the USB controller.

When the IN packet has been sent, the TXCMPL bit in the UEPSTAX register is set by the USB controller. This triggers a USB interrupt if enabled. The firmware will clear the TXCMPL bit before filling the endpoint FIFO with new data.

The firmware will never write more bytes than supported by the endpoint FIFO

21.6.4 Isochronous IN Transactions in Ping-pong Mode

An endpoint will be first enabled and configured before being able to send Isochronous packets.

The firmware will fill the FIFO bank 0 with the data to be sent and set the TXRDY bit in the UEP-STAX register to allow the USB controller to send the data stored in FIFO at the next IN request concerning the endpoint. The FIFO banks are automatically switched, and the firmware can immediately write into the endpoint FIFO bank 1.

If the TXRDY bit is not set when the IN request occurs, nothing will be sent by the USB controller.

When the IN packet concerning the bank 0 has been sent, the TXCMPL bit is set by the USB controller. This triggers a USB interrupt if enabled. The firmware will clear the TXCMPL bit before filling the endpoint FIFO bank 0 with new data. The FIFO banks are then automatically switched.

When the IN packet concerning the bank 1 has been sent, the TXCMPL bit is set by the USB controller. This triggers a USB interrupt if enabled. The firmware will clear the TXCMPL bit before filling the endpoint FIFO bank 1 with new data.

The bank switch is performed by the USB controller each time the TXRDY bit is set by the firmware. Until the TXRDY bit has been set by the firmware for an endpoint bank, the USB controller won't send anything at each IN requests concerning this bank.

The firmware will never write more bytes than supported by the endpoint FIFO.

21.7 Miscellaneous

21.7.1 USB Reset

The EORINT bit in the USBINT register is set by hardware when a End Of Reset has been detected on the USB bus. This triggers a USB interrupt if enabled. The USB controller is still enabled, but all the USB registers are reset by hardware. The firmware will clear the EORINT bit to allow the next USB reset detection.



Table 21-16.UFNUMH RegisterUFNUMH (S:BBh, read-only)USB Frame Number High Register

7	6	5	4	3	2	1	0			
-	-	CRCOK	CRCERR	-	FNUM10	FNUM9	FNUM8			
Bit Number	Bit Mnemonic	Description	Description							
5	CRCOK	Frame Number CRC OK This bit is set by hardware when a new Frame Number in Start of Frame Packet is received without CRC error. This bit is updated after every Start of Frame packet receipt. Important note: the Start of Frame interrupt is generated just after the PID receipt.								
4	CRCERR	Frame Number CRC Error This bit is set by hardware when a corrupted Frame Number in Start of Frame packet is received. This bit is updated after every Start of Frame packet receipt. Important note: the Start of Frame interrupt is generated just after the PID receipt.								
3	-	Reserved The value read	d from this bit is	s always 0. Do	not set this bit.					
2-0	FNUM[10:8]	Frame Number FNUM[10:8] at UFNUML (S:B provided in the (S:BEh) USB (corrupted SOF	Frame Number FNUM[10:8] are the upper 3 bits of the 11-bit Frame Number (see the "UFNUML Register UFNUML (S:BAh, read-only) USB Frame Number Low Register" on page 150). It is provided in the last received SOF packet (see SOFINT in the "USBIEN Register USBIEN (S:BEh) USB Global Interrupt Enable Register" on page 141). FNUM is updated if a corrupted SOF is received.							

Reset Value = 00h

 Table 21-17.
 UFNUML Register

 UFNUML (S:BAh, read-only)
 USB Frame Number Low Register

7	6	5	4	3	2	1	0		
FNUM7	FNUM6	FNUM5	FNUM4	FNUM3	FNUM2	FNUM1	FNUM0		
Bit Number	Bit Mnemonic	Description							
7 - 0	FNUM[7:0]	Frame Number FNUM[7:0] are the lower 8 bits of the 11-bit Frame Number (See "UFNUMH Register UFNUMH (S:BBh, read-only) USB Frame Number High Register" on page 150.).							

Reset Value = 00h

24. Power Management

24.1 Idle Mode

An instruction that sets PCON.0 indicates that it is the last instruction to be executed before going into the Idle mode. In the Idle mode, the internal clock signal is gated off to the CPU, but not to the interrupt, Timer, and Serial Port functions. The CPU status is preserved in its entirety: the Stack Pointer, Program Counter, Program Status Word, Accumulator and all other registers maintain their data during Idle. The port pins hold the logical states they had at the time Idle was activated. ALE and PSEN hold at logic high level.

There are two ways to terminate the Idle mode. Activation of any enabled interrupt will cause PCON.0 to be cleared by hardware, terminating the Idle mode. The interrupt will be serviced, and following RETI the next instruction to be executed will be the one following the instruction that put the device into idle.

The flag bits GF0 and GF1 can be used to give an indication if an interrupt occurred during normal operation or during an Idle. For example, an instruction that activates Idle can also set one or both flag bits. When Idle is terminated by an interrupt, the interrupt service routine can examine the flag bits.

The other way of terminating the Idle mode is with a hardware reset. Since the clock oscillator is still running, the hardware reset needs to be held active for only two machine cycles (24 oscillator periods) to complete the reset.

24.2 Power-down Mode

To save maximum power, a power-down mode can be invoked by software (refer to Table 13, PCON register).

In power-down mode, the oscillator is stopped and the instruction that invoked power-down mode is the last instruction executed. The internal RAM and SFRs retain their value until the power-down mode is terminated. V_{CC} can be lowered to save further power. Either a hardware reset or an external interrupt can cause an exit from power-down. To properly terminate power-down, the reset or external interrupt should not be executed before V_{CC} is restored to its normal operating level and must be held active long enough for the oscillator to restart and stabilize.

Only:

- external interrupt INTO,
- external interrupt INT1,
- · Keyboard interrupt and
- USB Interrupt

are useful to exit from power-down. For that, interrupt must be enabled and configured as level or edge sensitive interrupt input. When Keyboard Interrupt occurs after a power down mode, 1024 clocks are necessary to exit to power-down mode and enter in operating mode.

Holding the pin low restarts the oscillator but bringing the pin high completes the exit as detailed in Figure 24-1. When both interrupts are enabled, the oscillator restarts as soon as one of the two inputs is held low and power-down exit will be completed when the first input is released. In this case, the higher priority interrupt service routine is executed. Once the interrupt is serviced, the next instruction to be executed after RETI will be the one following the instruction that put AT89C5130A/31A-M into power-down mode.





25. Hardware Watchdog Timer

The WDT is intended as a recovery method in situations where the CPU may be subjected to software upset. The WDT consists of a 14-bit counter and the WatchDog Timer ReSeT (WDTRST) SFR. The WDT is by default disabled from exiting reset. To enable the WDT, user must write 01EH and 0E1H in sequence to the WDTRST, SFR location 0A6H. When WDT is enabled, it will increment every machine cycle while the oscillator is running and there is no way to disable the WDT except through reset (either hardware reset or WDT overflow reset). When WDT overflows, it will drive an output RESET LOW pulse at the RST-pin.

25.1 Using the WDT

To enable the WDT, user must write 01EH and 0E1H in sequence to the WDTRST, SFR location 0A6H. When WDT is enabled, the user needs to service it by writing to 01EH and 0E1H to WDTRST to avoid WDT overflow. The 14-bit counter overflows when it reaches 16383 (3FFFH) and this will reset the device. When WDT is enabled, it will increment every machine cycle while the oscillator is running. This means the user must reset the WDT at least every 16383 machine cycle. To reset the WDT the user must write 01EH and 0E1H to WDTRST. WDTRST is a write only register. The WDT counter cannot be read or written. When WDT overflows, it will generate an output RESET pulse at the RST-pin. The RESET pulse duration is 96 x T_{CLK PERIPH}, where T_{CLK PERIPH} = 1/F_{CLK PERIPH}. To make the best use of the WDT, it should be serviced in those sections of code that will periodically be executed within the time required to prevent a WDT reset.

To have a more powerful WDT, a 2^7 counter has been added to extend the Time-out capability, ranking from 16 ms to 2s at F_{OSCA} = 12 MHz. To manage this feature, refer to WDTPRG register description, Table 25-2.

Table 25-1.	WDTRS1 WDTRS1	WDTRST Register WDTRST - Watchdog Reset Register (0A6h)								
7	6	5	4	3	2	1	0			
-	-	-	-	-	-	-	-			
Deeet Value										

Reset Value = XXXX XXXXb

Write only, this SFR is used to reset/enable the WDT by writing 01EH then 0E1H in sequence.



19	Serial Peripheral Interface (SPI)	
	19.1	Features
	19.2	Signal Description
	19.3	Functional Description95
20	Two Wire Interface (TWI)	
	20.1	Description104
	20.2	Notes107
	20.3	Registers118
21	USB Controller	
	21.1	Description120
	21.2	Configuration122
	21.3	Read/Write Data FIFO124
	21.4	Bulk/Interrupt Transactions125
	21.5	Control Transactions
	21.6	Isochronous Transactions
	21.7	Miscellaneous
	21.8	Suspend/Resume Management133
	21.9	Detach Simulation
	21.10	USB Interrupt System
	21.11	USB Registers139
22	Reset	
	22.1	Introduction151
	22.2	Reset Input151
	22.3	Reset Output151
23	Power Monitor1	
	23.1	Description153
24	4 Power Management	
	24.1	Idle Mode155
	24.2	Power-down Mode155
	24.3	Registers157
25	Hardware Watchdog Timer	
	25.1	Using the WDT158
	25.2	WDT During Power-down and Idle159
26	Reduced EMI Mode160	

III AT89C5130A/31A-M