



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	Obsolete
Core Processor	8051
Core Size	8-Bit
Speed	33MHz
Connectivity	UART/USART
Peripherals	POR, WDT
Number of I/O	32
Program Memory Size	64KB (64K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	512 x 8
Voltage - Supply (Vcc/Vdd)	4.5V ~ 5.5V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Surface Mount
Package / Case	44-LQFP
Supplier Device Package	44-LQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/p89c60x2bbd-00-557

Email: info@E-XFL.COM

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

## Product data

### P89C60X2/61X2

### **BLOCK DIAGRAM 1**



### P89C60X2/61X2

### LOGIC SYMBOL



### PLASTIC LEADED CHIP CARRIER PIN FUNCTIONS



#### 34 44 Ó 33 1 🖂 LQFP 11 🖵 \_\_\_\_\_ 23 12 22 Function Pin Pin Function Pin Function P0.6/AD6 P1 5 V<sub>SS</sub> NIC\* 31 1 16 P1.6 P0.5/AD5 2 17 32 P0.4/AD4 3 P1.7 P2.0/A8 18 33 P0.3/AD3 P2.1/A9 34 RST 4 19 P2.2/A10 35 P0.2/AD2 P3.0/RxD 20 5 P2.3/A11 36 P0.1/AD1 NIC\* 21 6 P3.1/TxD 37 P0.0/AD0 7 22 P2.4/A12 38 8 P3.2/INT0 23 P2.5/A13 V<sub>CC</sub> NIC\* 39 P3.3/INT1 P2.6/A14 9 24 25 P2.7/A15 40 P1.0/T2 10 P3 4/T0 P3.5/T1 PSEN 41 P1.1/T2EX 26 11 42 P1.2 12 P3.6/WR 27 ALE 43 P1.3 P3.7/RD 28 NIC\* 13 44 P1.4 EA/V<sub>PP</sub> XTAL2 14 29 P0.7/AD7 15 XTAL1 30 \* NO INTERNAL CONNECTION SU01487

## PLASTIC DUAL IN-LINE PACKAGE PIN FUNCTIONS

T2/P1.0 1		40 V <sub>CC</sub>
T2EX/P1.1 2		39 P0.0/AD0
P1.2 3		38 P0.1/AD1
P1.3 4		37 P0.2/AD2
P1.4 5		36 P0.3/AD3
P1.5 6		35 P0.4/AD4
P1.6 7		34 P0.5/AD5
P1.7 8		33 P0.6/AD6
RST 9		32 P0.7/AD7
RxD/P3.0 10	DUAL IN-LINE	31 EA/V <sub>PP</sub>
TxD/P3.1 11	PACKAGE	30 ALE/PROG
INT0/P3.2 12		29 PSEN
INT1/P3.3 13		28 P2.7/A15
T0/P3.4 14		27 P2.6/A14
T1/P3.5 15		26 P2.5/A13
WR/P3.6 16		25 P2.4/A12
RD/P3.7 17		24 P2.3/A11
XTAL2 18		23 P2.2/A10
XTAL1 19		22 P2.1/A9
V <sub>SS</sub> 20		21 P2.0/A8
		SU01780

### LOW PROFILE QUAD FLAT PACK PIN FUNCTIONS

### SPECIAL FUNCTION REGISTERS (see notes on next page)

SYMBOL	DESCRIPTION	DIRECT ADDRESS	MSB	IT ADDRE	ESS, SYM	BOL, OR	ALTERNA	TIVE PO	RT FUNC	TION	RESET VALUE
ACC*	Accumulator	E0H	E7	E6	E5	E4	E3	E2	E1	E0	00H
AUXR#	Auxiliary	8EH	-	-	-	-	-	-	EXTRAM	AO	xxxxxx00B
AUXR1#	Auxiliary 1	A2H	-	-	-	-	GF2	0	-	DPS	xxx000x0B
B*	B register	F0H	F7	F6	F5	F4	F3	F2	F1	F0	00H
CKCON	Clock Control Register	8FH	-	WDX2	-	-	-	-	-	X2	x0xxxxx0B
DPTR:	Data Pointer (2 bytes)					1					1
DPH	Data Pointer High	83H									00H
DPL	Data Pointer Low	82H									00H
			AF	AE	AD	AC	AB	AA	A9	A8	
IE*	Interrupt Enable	A8H	ĒĀ	-	ET2	ES	ET1	EX1	ET0	EX0	0x000000B
			BF	BE	BD	BC	BB	BA	B9	B8	1
IP*	Interrupt Priority	B8H	-	-	PT2	PS	PT1	PX1	PT0	PX0	xx000000B
IPH#	Interrupt Priority High	B7H	-	-	PT2H	PSH	PT1H	PX1H	PT0H	PX0H	xx000000B
			87	86	85	84	83	82	81	80	1
P0*	Port 0	80H	AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0	FFH
			97	96	95	94	93	92	91	90	1
P1*	Port 1	90H	-	-	-	-	-	-	T2EX	T2	FFH
			A7	A6	A5	A4	A3	A2	A1	A0	1
P2*	Port 2	A0H	AD15	AD14	AD13	AD12	AD11	AD10	AD9	AD8	FFH
			B7	B6	B5	B4	B3	B2	B1	B0	1
P3*	Port 3	B0H	RD	WR	T1	T0	INT1	INT0	TxD	RxD	FFH
											1
PCON#1	Power Control	87H	SMOD1	SMOD0	-	POF	GF1	GF0	PD	IDL	00xx0000B
			D7	D6	D5	D4	D3	D2	D1	D0	1
PSW*	Program Status Word	D0H	CY	AC	F0	RS1	RS0	OV	-	Р	000000x0B
RACAP2H#	Timer 2 Capture High	СВН				-		-		-	00H
RACAP2L#	Timer 2 Capture Low	CAH									00H
SADDR#	Slave Address	A9H									00H
SADEN#	Slave Address Mask	B9H									00H
SBUF	Serial Data Buffer	99H									xxxxxxxB
			9F	9E	9D	9C	9B	9A	99	98	
SCON*	Serial Control	98H	SM0/FE	SM1	SM2	REN	TB8	RB8	TI	RI	00H
SP	Stack Pointer	81H									07H
			8F	8E	8D	8C	8B	8A	89	88	
TCON*	Timer Control	88H	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0	00H
			CF	CE	CD	CC	CB	CA	C9	C8	
T2CON*	Timer 2 Control	C8H	TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/T2	CP/RL2	00H
T2MOD#	Timer 2 Mode Control	C9H	-	-	-	-	-	-	T2OE	DCEN	xxxxxx00B
TH0	Timer High 0	8CH									00H
TH1	Timer High 1	8DH									00H
TH2#	Timer High 2	CDH									00H
TL0	Timer Low 0	8AH									00H
TL1	Timer Low 1	8BH									00H
TL2#	Timer Low 2	ССН									00H
TMOD	Timer Mode	89H	GATE	C/T	M1	MO	GATE	C/T	M1	MO	00H
WDTRST	Watchdog Timer Reset	A6H									

### P89C60X2/61X2

### FLASH EPROM MEMORY

### **GENERAL DESCRIPTION**

The P89C60X2/61X2 Flash memory augments EPROM functionality with in-circuit electrical erasure and programming. The Flash can be read and written as bytes. The Chip Erase operation will erase the entire program memory. The Block Erase function can erase any Flash block. In-system programming (ISP) and standard parallel programming are both available. On-chip erase and write timing generation contribute to a user friendly programming interface.

The P89C60X2/61X2 Flash reliably stores memory contents even after 10,000 erase and program cycles. The cell is designed to optimize the erase and programming mechanisms. In addition, the combination of advanced tunnel oxide processing and low internal electric fields for erase and programming operations produces reliable cycling. The P89C60X2/61X2 uses a +5 V V<sub>PP</sub> supply to perform the Program/Erase algorithms (12 V tolerant).

### FEATURES

- Flash EPROM internal program memory with Block Erase.
- Internal 1-kbyte fixed BootROM, containing low-level in-system programming routines and a default serial loader.
- Loader in BootROM allows in-system programming via the serial port.
- Up to 64 kbytes external program memory if the internal program memory is disabled (EA = 0).
- Programming and erase voltage +5 V (+12 V tolerant).
- Read/Programming/Erase using ISP:
  - Byte Programming (8 µs).
  - Typical erase times:
     Block Erase (4 kbytes) in 3 seconds.
     Full-chip erase in 15 seconds.
- Parallel programming with 87C51 compatible hardware interface to programmer.

- Programmable security for the code in the Flash.
- 10,000 minimum erase/program cycles for each byte.
- 10-year minimum data retention.

### FLASH PROGRAMMING AND ERASURE

There are two methods of erasing or programming of the Flash memory that may be used. First, the on-chip ISP boot loader may be invoked. Second, the Flash may be programmed or erased using parallel method by using a commercially available EPROM programmer. The parallel programming method used by these devices is similar to that used by EPROM 87C51, but it is not identical, and the commercially available programmer will need to have support for these devices.

### FLASH MEMORY CHARACTERISTICS

#### Flash User Code Memory Organization

The P89C60X2/61X2 contains 64 kbytes Flash user code program memory organized into 4-kbyte blocks (see Figure 1).

#### **Boot ROM**

When the microcontroller programs its Flash memory during ISP, all of the low level details are handled by code that is contained in a 1 kbyte BootROM. BootROM operations include: erase block, program byte, verify byte, program security bit, etc.

### **Clock Mode**

The clock mode feature sets operating frequency to be 1/12 or 1/6 of the oscillator frequency. The clock mode configuration bit, FX2, is located in the Security Block (See Table 1). FX2, when programmed, will override the SFR clock mode bit (X2) in the CKCON register. If FX2 is erased, then the SFR bit (X2) may be used to select between 6-clock and 12-clock mode.

### P89C60X2/61X2

Product data

RECORD TYPE	COMMAND/DATA FUNCTION
03 (cont.)	Subfunction Code = 07 (Full Chip Erase) Erases all blocks, security bits, and sets status byte to default values ff = 07 ss = don't care dd = don't care Example: :0100000307F5 full chip erase
	Subfunction Code = 0C (Erase 4k blocks) ff = 0C ss = block code as shown below:
04	Display Device Data or Blank Check - Record type 04 causes the contents of the entire Flash array to be sent out the serial port in a formatted display. This display consists of an address and the contents of 16 bytes starting with that address. No display of the device contents will occur if security bit 2 has been programmed. Data to the serial port is initiated by the reception of any character and terminated by the reception of any character. General Format of Function 04 :05xxxx04sssseeeeffcc Where: 05 = number of bytes (hex) in record xxxx = required field, but value is a "don't care" 04 = "Display Device Data or Blank Check" function code ssss = starting address eeee = ending address ff = subfunction 00 = display data 01 = blank check 02 = display data in data block (valid addresses: 0001 ~ 0FFFH) cc = checksum Example 1: :050000440004FFF0069 display 4000-4FFF Example 2: :050000440000FFF02E7 display data in data block (the data at address 0000 is invalid)

### P89C60X2/61X2

### Security

The security feature protects against software piracy and prevents the contents of the FLASH from being read. The Security Lock bits are located in FLASH. The P89C60X2/61X2 has 3 programmable security lock bits that will provide different levels of protection for the on-chip code and data (see Table 3). Unlike the ROM and OTP versions, the security lock bits are independent. LB3 includes the security protection of LB1.

### Table 3.

SECURITY LOCK BITS <sup>1</sup>						
Level	PROTECTION DESCRIPTION					
LB1	MOVC instructions executed from external program memory are disabled from fetching code bytes from internal memory.					
LB2	Program verification is disabled					
LB3	External execution is disabled.					

NOTE:

1. The security lock bits are independent.







Figure 6. Timer/Counter 0/1 Mode 2: 8-Bit Auto-Reload

Product data

### Table 6. Timer 2 Operating Modes

RCLK + TCLK	CP/RL2	TR2	MODE
0	0	1	16-bit Auto-reload
0	1	1	16-bit Capture
1	Х	1	Baud rate generator
Х	Х	0	(off)

CON A Bit	ddress = C t Addressa	C8H Ible						F	Reset Value =	= 00H
		7	6	5	4	3	2	1	0	
		TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/T2	CP/RL2	
Symbol	Positio	n Nar	ne and Sig	nificance						
TF2	T2CON	.7 Tim whe	er 2 overflo en either RC	w flag set b LK or TCL	y a Timer 2 K = 1.	overflow and	d must be c	eared by so	oftware. TF2	will not be set
EXF2	T2CON	.6 Tim EXI inte cou	er 2 externa EN2 = 1. WI rrupt routine nter mode (	al flag set wi nen Timer 2 e. EXF2 mu DCEN = 1).	nen either a interrupt is st be cleare	capture or r enabled, EX d by softwar	reload is cau (F2 = 1 will re. EXF2 do	used by a no cause the C es not caus	egative trans PU to vector e an interrup	sition on T2EX an r to the Timer 2 st in up/down
RCLK	T2CON	.5 Rec in m	Receive clock flag. When set, causes the serial port to use Timer 2 overflow pulses for its receive clock in modes 1 and 3, RCLK = 0 causes Timer 1 overflow to be used for the receive clock.							
TCLK	T2CON	.4 Tra in m	nsmit clock nodes 1 and	flag. When 3. TCLK =	set, causes 0 causes T	the serial po imer 1 overfl	ort to use Ti ows to be u	mer 2 overfi sed for the	low pulses fo transmit cloo	or its transmit cloc k.
EXEN2	T2CON	.3 Tim tran igno	Timer 2 external enable flag. When set, allows a capture or reload to occur as a result of a negative transition on T2EX if Timer 2 is not being used to clock the serial port. EXEN2 = 0 causes Timer 2 to ignore events at T2EX							
TR2	T2CON	.2 Sta	rt/stop conti	ol for Timer	2. A logic 1	starts the ti	mer.			
C/T2	T2CON	.1 Tim	Timer or counter select. (Timer 2) 0 = Internal timer (OSC/12 in 12-clock mode or OSC/6 in 6-clock mode) 1 = External event counter (falling edge triggered).							
CP/RL2	T2CON	.0 Cap clea EXI	oture/Reload ared, auto-re EN2 = 1. WI	d flag. Wher bloads will c nen either R	n set, captur ccur either CLK = 1 or	res will occur with Timer 2 TCLK = 1, th	r on negativ overflows o nis bit is ign	e transitions or negative to ored and th	s at T2EX if I transitions at e timer is for	EXEN2 = 1. When T2EX when ced to auto-reloa
		on	limer 2 ove	rflow.						SU0163

Figure 8. Timer/Counter 2 (T2CON) Control Register

## P89C60X2/61X2



Figure 9. Timer 2 in Capture Mode

T2MOD	Addre	ess = 0C9H							Reset Va	lue = XXXX XX00B
	Not Bit	Addressab	le							
		7	6	5	4	3	2	1	0	
		_	_	_	_	_	_	T2OE	DCEN	
Symbol	Positi	ion	F	unction ot implemen	ted, reserve	ed for future	use.*			
T2OE	T2MO	D.1	Ti	mer 2 Outpu	ut Enable bi	t.				
DCEN * User softv In that ca: indetermi	T2MO ware sho se, the re nate.	DD.0 ould not writ eset or inac	D cc e 1s to reser tive value of	own Count E ounter. rved bits. Th the new bit	Enable bit. V ese bits ma will be 0, ar	When set, th by be used ir nd its active	is allows Ti n future 805 value will b	imer 2 to be o 51 family pro- be 1. The valu	configured a ducts to invo ue read fron	as an up/down oke new features. n a reserved bit is

Figure 10. Timer 2 Mode (T2MOD) Control Register



Figure 11. Timer 2 in Auto-Reload Mode (DCEN = 0)



Figure 12. Timer 2 Auto Reload Mode (DCEN = 1)

### FULL-DUPLEX ENHANCED UART

#### Standard UART operation

The serial port is full duplex, meaning it can transmit and receive simultaneously. It is also receive-buffered, meaning it can commence reception of a second byte before a previously received byte has been read from the register. (However, if the first byte still hasn't been read by the time reception of the second byte is complete, one of the bytes will be lost.) The serial port receive and transmit registers are both accessed at Special Function Register SBUF. Writing to SBUF loads the transmit register, and reading SBUF accesses a physically separate receive register.

The serial port can operate in 4 modes:

- Mode 0: Serial data enters and exits through RxD. TxD outputs the shift clock. 8 bits are transmitted/received (LSB first). The baud rate is fixed at 1/12 the oscillator frequency (in 12-clock mode) or 1/6 the oscillator frequency (in 6-clock mode).
- Mode 1: 10 bits are transmitted (through TxD) or received (through RxD): a start bit (0), 8 data bits (LSB first), and a stop bit (1). On receive, the stop bit goes into RB8 in Special Function Register SCON. The baud rate is variable.
- Mode 2: 11 bits are transmitted (through TxD) or received (through RxD): start bit (0), 8 data bits (LSB first), a programmable 9th data bit, and a stop bit (1). On Transmit, the 9th data bit (TB8 in SCON) can be assigned the value of 0 or 1. Or, for example, the parity bit (P, in the PSW) could be moved into TB8. On receive, the 9th data bit goes into RB8 in Special Function Register SCON, while the stop bit is ignored. The baud rate is programmable to either 1/32 or 1/64 the oscillator frequency (in 12-clock mode) or 1/16 or 1/32 the oscillator frequency (in 6-clock mode).
- Mode 3: 11 bits are transmitted (through TxD) or received (through RxD): a start bit (0), 8 data bits (LSB first), a programmable 9th data bit, and a stop bit (1). In fact, Mode 3 is the same as Mode 2 in all respects except baud rate. The baud rate in Mode 3 is variable.

In all four modes, transmission is initiated by any instruction that uses SBUF as a destination register. Reception is initiated in Mode 0 by the condition RI = 0 and REN = 1. Reception is initiated in the other modes by the incoming start bit if REN = 1.

#### **Multiprocessor Communications**

Modes 2 and 3 have a special provision for multiprocessor communications. In these modes, 9 data bits are received. The 9th one goes into RB8. Then comes a stop bit. The port can be programmed such that when the stop bit is received, the serial port interrupt will be activated only if RB8 = 1. This feature is enabled by setting bit SM2 in SCON. A way to use this feature in multiprocessor systems is as follows:

When the master processor wants to transmit a block of data to one of several slaves, it first sends out an address byte which identifies the target slave. An address byte differs from a data byte in that the 9th bit is 1 in an address byte and 0 in a data byte. With SM2 = 1, no slave will be interrupted by a data byte. An address byte, however, will interrupt all slaves, so that each slave can examine the received byte and see if it is being addressed. The addressed slave will clear its SM2 bit and prepare to receive the data bytes that will be coming. The slaves that weren't being addressed leave their SM2s set and go on about their business, ignoring the coming data bytes.

SM2 has no effect in Mode 0, and in Mode 1 can be used to check the validity of the stop bit. In a Mode 1 reception, if SM2 = 1, the receive interrupt will not be activated unless a valid stop bit is received.

#### Serial Port Control Register

The serial port control and status register is the Special Function Register SCON, shown in Figure 14. This register contains not only the mode selection bits, but also the 9th data bit for transmit and receive (TB8 and RB8), and the serial port interrupt bits (TI and RI).

#### **Baud Rates**

The baud rate in Mode 0 is fixed: Mode 0 Baud Rate = Oscillator Frequency / 12 (in 12-clock mode) or / 6 (in 6-clock mode). The baud rate in Mode 2 depends on the value of bit SMOD in Special Function Register PCON. If SMOD = 0 (which is the value on reset), and the port pins in 12-clock mode, the baud rate is 1/64 the oscillator frequency. If SMOD = 1, the baud rate is 1/32 the oscillator frequency. In 6-clock mode, the baud rate is 1/32 or 1/16 the oscillator frequency, respectively.

Mode 2 Baud Rate =

$$\frac{2^{\text{SMOD}}}{n} \times (\text{Oscillator Frequency})$$

Where:

n = 64 in 12-clock mode, 32 in 6-clock mode

The baud rates in Modes 1 and 3 are determined by the Timer 1 or Timer 2 overflow rate.

#### Using Timer 1 to Generate Baud Rates

When Timer 1 is used as the baud rate generator (T2CON.RCLK = 0, T2CON.TCLK = 0), the baud rates in Modes 1 and 3 are determined by the Timer 1 overflow rate and the value of SMOD as follows:

Mode 1, 3 Baud Rate =

$$\frac{2^{\text{SMOD}}}{n} \times$$
 (Timer 1 Overflow Rate)

Where:

n = 32 in 12-clock mode, 16 in 6-clock mode

The Timer 1 interrupt should be disabled in this application. The Timer itself can be configured for either "timer" or "counter" operation, and in any of its 3 running modes. In the most typical applications, it is configured for "timer" operation, in the auto-reload mode (high nibble of TMOD = 0010B). In that case the baud rate is given by the formula:

Mode 1, 3 Baud Rate =

$$\frac{2^{\text{SMOD}}}{n} \times \frac{\text{Oscillator Frequency}}{12 \times [256-(\text{TH1})]}$$

Where:

n = 32 in 12-clock mode, 16 in 6-clock mode

One can achieve very low baud rates with Timer 1 by leaving the Timer 1 interrupt enabled, and configuring the Timer to run as a 16-bit timer (high nibble of TMOD = 0001B), and using the Timer 1 interrupt to do a 16-bit software reload. Figure 15 lists various commonly used baud rates and how they can be obtained from Timer 1.

### P89C60X2/61X2

shifted to the left one position. The value that comes in from the right is the value that was sampled at the P3.0 pin at S5P2 of the same machine cycle.

As data bits come in from the right, 1s shift out to the left. When the 0 that was initially loaded into the rightmost position arrives at the leftmost position in the shift register, it flags the RX Control block to do one last shift and load SBUF. At S1P1 of the 10th machine cycle after the write to SCON that cleared RI, RECEIVE is cleared as RI is set.

#### More About Mode 1

Ten bits are transmitted (through TxD), or received (through RxD): a start bit (0), 8 data bits (LSB first), and a stop bit (1). On receive, the stop bit goes into RB8 in SCON. In the 80C51 the baud rate is determined by the Timer 1 or Timer 2 overflow rate.

Figure 17 shows a simplified functional diagram of the serial port in Mode 1, and associated timings for transmit receive.

Transmission is initiated by any instruction that uses SBUF as a destination register. The "write to SBUF" signal also loads a 1 into the 9th bit position of the transmit shift register and flags the TX Control unit that a transmission is requested. Transmission actually commences at S1P1 of the machine cycle following the next rollover in the divide-by-16 counter. (Thus, the bit times are synchronized to the divide-by-16 counter, not to the "write to SBUF" signal.)

The transmission begins with activation of SEND which puts the start bit at TxD. One bit time later, DATA is activated, which enables the output bit of the transmit shift register to TxD. The first shift pulse occurs one bit time after that.

As data bits shift out to the right, zeros are clocked in from the left. When the MSB of the data byte is at the output position of the shift register, then the 1 that was initially loaded into the 9th position is just to the left of the MSB, and all positions to the left of that contain zeros. This condition flags the TX Control unit to do one last shift and then deactivate SEND and set TI. This occurs at the 10th divide-by-16 rollover after "write to SBUF."

Reception is initiated by a detected 1-to-0 transition at RxD. For this purpose RxD is sampled at a rate of 16 times whatever baud rate has been established. When a transition is detected, the divide-by-16 counter is immediately reset, and 1FFH is written into the input shift register. Resetting the divide-by-16 counter aligns its rollovers with the boundaries of the incoming bit times.

The 16 states of the counter divide each bit time into 16ths. At the 7th, 8th, and 9th counter states of each bit time, the bit detector samples the value of RxD. The value accepted is the value that was seen in at least 2 of the 3 samples. This is done for noise rejection. If the value accepted during the first bit time is not 0, the receive circuits are reset and the unit goes back to looking for another 1-to-0 transition. This is to provide rejection of false start bits. If the start bit proves valid, it is shifted into the input shift register, and reception of the rest of the frame will proceed.

As data bits come in from the right, 1s shift out to the left. When the start bit arrives at the leftmost position in the shift register (which in mode 1 is a 9-bit register), it flags the RX Control block to do one last shift, load SBUF and RB8, and set RI. The signal to load SBUF and RB8, and to set RI, will be generated if, and only if, the following conditions are met at the time the final shift pulse is generated.: 1. R1 = 0, and

2. Either SM2 = 0, or the received stop bit = 1.

If either of these two conditions is not met, the received frame is irretrievably lost. If both conditions are met, the stop bit goes into RB8, the 8 data bits go into SBUF, and RI is activated. At this time,

whether the above conditions are met or not, the unit goes back to looking for a 1-to-0 transition in RxD.

#### More About Modes 2 and 3

Eleven bits are transmitted (through TxD), or received (through RxD): a start bit (0), 8 data bits (LSB first), a programmable 9th data bit, and a stop bit (1). On transmit, the 9th data bit (TB8) can be assigned the value of 0 or 1. On receive, the 9the data bit goes into RB8 in SCON. The baud rate is programmable to either 1/32 or 1/64 (12-clock mode) or 1/16 or 1/32 the oscillator frequency (6-clock mode) the oscillator frequency in Mode 2. Mode 3 may have a variable baud rate generated from Timer 1 or Timer 2.

Figures 18 and 19 show a functional diagram of the serial port in Modes 2 and 3. The receive portion is exactly the same as in Mode 1. The transmit portion differs from Mode 1 only in the 9th bit of the transmit shift register.

Transmission is initiated by any instruction that uses SBUF as a destination register. The "write to SBUF" signal also loads TB8 into the 9th bit position of the transmit shift register and flags the TX Control unit that a transmission is requested. Transmission commences at S1P1 of the machine cycle following the next rollover in the divide-by-16 counter. (Thus, the bit times are synchronized to the divide-by-16 counter, not to the "write to SBUF" signal.)

The transmission begins with activation of SEND, which puts the start bit at TxD. One bit time later, DATA is activated, which enables the output bit of the transmit shift register to TxD. The first shift pulse occurs one bit time after that. The first shift clocks a 1 (the stop bit) into the 9th bit position of the shift register. Thereafter, only zeros are clocked in. Thus, as data bits shift out to the right, zeros are clocked in from the left. When TB8 is at the output position of the shift register, then the stop bit is just to the left of TB8, and all positions to the left of that contain zeros. This condition flags the TX Control unit to do one last shift and then deactivate SEND and set TI. This occurs at the 11th divide-by-16 rollover after "write to SUBF."

Reception is initiated by a detected 1-to-0 transition at RxD. For this purpose RxD is sampled at a rate of 16 times whatever baud rate has been established. When a transition is detected, the divide-by-16 counter is immediately reset, and 1FFH is written to the input shift register.

At the 7th, 8th, and 9th counter states of each bit time, the bit detector samples the value of R-D. The value accepted is the value that was seen in at least 2 of the 3 samples. If the value accepted during the first bit time is not 0, the receive circuits are reset and the unit goes back to looking for another 1-to-0 transition. If the start bit proves valid, it is shifted into the input shift register, and reception of the rest of the frame will proceed.

As data bits come in from the right, 1s shift out to the left. When the start bit arrives at the leftmost position in the shift register (which in Modes 2 and 3 is a 9-bit register), it flags the RX Control block to do one last shift, load SBUF and RB8, and set RI.

The signal to load SBUF and RB8, and to set RI, will be generated if, and only if, the following conditions are met at the time the final shift pulse is generated.

1. RI = 0, and

2. Either SM2 = 0, or the received 9th data bit = 1.

If either of these conditions is not met, the received frame is irretrievably lost, and RI is not set. If both conditions are met, the received 9th data bit goes into RB8, and the first 8 data bits go into SBUF. One bit time later, whether the above conditions were met or not, the unit goes back to looking for a 1-to-0 transition at the RxD input.



Figure 17. Serial Port Mode 1

Product data



Figure 18. Serial Port Mode 2

#### Enhanced UART operation

In addition to the standard operation modes, the UART can perform framing error detect by looking for missing stop bits, and automatic address recognition. The UART also fully supports multiprocessor communication.

When used for framing error detect the UART looks for missing stop bits in the communication. A missing bit will set the FE bit in the SCON register. The FE bit shares the SCON.7 bit with SM0 and the function of SCON.7 is determined by PCON.6 (SMOD0) (see Figure 20). If SMOD0 is set then SCON.7 functions as FE. SCON.7 functions as SM0 when SMOD0 is cleared. When used as FE SCON.7 can only be cleared by software. Refer to Figure 21.

#### Automatic Address Recognition

Automatic Address Recognition is a feature which allows the UART to recognize certain addresses in the serial bit stream by using hardware to make the comparisons. This feature saves a great deal of software overhead by eliminating the need for the software to examine every serial address which passes by the serial port. This feature is enabled by setting the SM2 bit in SCON. In the 9 bit UART modes, mode 2 and mode 3, the Receive Interrupt flag (RI) will be automatically set when the received byte contains either the "Given" address or the "Broadcast" address. The 9 bit mode requires that the 9th information bit is a 1 to indicate that the received information is an address and not data. Automatic address recognition is shown in Figure 22.

The 8 bit mode is called Mode 1. In this mode the RI flag will be set if SM2 is enabled and the information received has a valid stop bit following the 8 address bits and the information is either a Given or Broadcast address.

Mode 0 is the Shift Register mode and SM2 is ignored.

Using the Automatic Address Recognition feature allows a master to selectively communicate with one or more slaves by invoking the Given slave address or addresses. All of the slaves may be contacted by using the Broadcast address. Two special Function Registers are used to define the slave's address, SADDR, and the address mask, SADEN. SADEN is used to define which bits in the SADDR are to be used and which bits are "don't care". The SADEN mask can be logically ANDed with the SADDR to create the "Given" address which the master will use for addressing each of the slaves. Use of the Given address allows multiple slaves to be recognized while excluding others. The following examples will help to show the versatility of this scheme:

Slave 0	SADDR	=	1100	0000
	SADEN	=	<u>1111</u>	1101
	Given	=	1100	00X0

Product	data

### P89C60X2/61X2

Slave 1	SADDR	=	1100 0000
	SADEN	=	<u>1111 1110</u>
	Given	=	1100 000X

In the above example SADDR is the same and the SADEN data is used to differentiate between the two slaves. Slave 0 requires a 0 in bit 0 and it ignores bit 1. Slave 1 requires a 0 in bit 1 and bit 0 is ignored. A unique address for Slave 0 would be 1100 0010 since slave 1 requires a 0 in bit 1. A unique address for slave 1 would be 1100 0001 since a 1 in bit 0 will exclude slave 0. Both slaves can be selected at the same time by an address which has bit 0 = 0 (for slave 0) and bit 1 = 0 (for slave 1). Thus, both could be addressed with 1100 0000.

In a more complex system the following could be used to select slaves 1 and 2 while excluding slave 0:

Slave 0	SADDR	=	1100	0000
	SADEN	=	<u>1111</u>	1001
	Given	=	1100	0XX0
Slave 1	SADDR	=	1110	0000
	SADEN	=	<u>1111</u>	1010
	Given	=	1110	0X0X
Slave 2	SADDR	=	1110	0000
	SADEN	=	<u>1111</u>	1100
	Given	=	1110	00XX

In the above example the differentiation among the 3 slaves is in the lower 3 address bits. Slave 0 requires that bit 0 = 0 and it can be uniquely addressed by 1110 0110. Slave 1 requires that bit 1 = 0 and it can be uniquely addressed by 1110 and 0101. Slave 2 requires that bit 2 = 0 and its unique address is 1110 0011. To select Slaves 0 and 1 and exclude Slave 2 use address 1110 0100, since it is necessary to make bit 2 = 1 to exclude slave 2.

The Broadcast Address for each slave is created by taking the logical OR of SADDR and SADEN. Zeros in this result are trended as don't-cares. In most cases, interpreting the don't-cares as ones, the broadcast address will be FF hexadecimal.

Upon reset SADDR (SFR address 0A9H) and SADEN (SFR address 0B9H) are leaded with 0s. This produces a given address of all "don't cares" as well as a Broadcast address of all "don't cares". This effectively disables the Automatic Addressing mode and allows the microcontroller to use standard 80C51 type UART drivers which do not make use of this feature.

### P89C60X2/61X2



#### Figure 27. Interrupt Response Timing Diagram

The polling cycle/LCALL sequence is illustrated in Figure 27.

Note that if an interrupt of higher priority level goes active prior to S5P2 of the machine cycle labeled C3 in Figure 27, then in accordance with the above rules it will be vectored to during C5 and C6, without any instruction of the lower priority routine having been executed.

Thus the processor acknowledges an interrupt request by executing a hardware-generated LCALL to the appropriate servicing routine. In some cases it also clears the flag that generated the interrupt, and in other cases it doesn't. It never clears the Serial Port flag. This has to be done in the user's software. It clears an external interrupt flag (IE0 or IE1) only if it was transition-activated. The

hardware-generated LCALL pushes the contents of the Program Counter on to the stack (but it does not save the PSW) and reloads the PC with an address that depends on the source of the interrupt being vectored to, as shown in Table 10.

Execution proceeds from that location until the RETI instruction is encountered. The RETI instruction informs the processor that this interrupt routine is no longer in progress, then pops the top two bytes from the stack and reloads the Program Counter. Execution of the interrupted program continues from where it left off.

Note that a simple RET instruction would also have returned execution to the interrupted program, but it would have left the interrupt control system thinking an interrupt was still in progress, making future interrupts impossible.

#### **External Interrupts**

The external sources can be programmed to be level-activated or transition-activated by setting or clearing bit IT1 or IT0 in Register TCON. If ITx = 0, external interrupt x is triggered by a detected low at the  $\overline{INTx}$  pin. If ITx = 1, external interrupt x is edge triggered. In this mode if successive samples of the  $\overline{INTx}$  pin show a high in one cycle and a low in the next cycle, interrupt request flag IEx in TCON is set. Flag bit IEx then requests the interrupt.

Since the external interrupt pins are sampled once each machine cycle, an input high or low should hold for at least 12 oscillator periods to ensure sampling. If the external interrupt is transition-activated, the external source has to hold the request pin high for at least one cycle, and then hold it low for at least one cycle. This is done to ensure that the transition is seen so that interrupt request flag IEx will be set. IEx will be automatically cleared by the CPU when the service routine is called.

If the external interrupt is level-activated, the external source has to hold the request active until the requested interrupt is actually generated. Then it has to deactivate the request before the interrupt service routine is completed, or else another interrupt will be generated.

#### **Response Time**

The INTO and INTT levels are inverted and latched into IEO and IE1 at S5P2 of every machine cycle. The values are not actually polled by the circuitry until the next machine cycle. If a request is active and conditions are right for it to be acknowledged, a hardware subroutine call to the requested service routine will be the next instruction to be executed. The call itself takes two cycles. Thus, a minimum of three complete machine cycles elapse between activation of an external interrupt request and the beginning of execution of the first instruction of the service routine. Figure 27 shows interrupt response timings.

A longer response time would result if the request is blocked by one of the 3 previously listed conditions. If an interrupt of equal or higher priority level is already in progress, the additional wait time obviously depends on the nature of the other interrupt's service routine. If the instruction in progress is not in its final cycle, the additional wait time cannot be more the 3 cycles, since the longest instructions (MUL and DIV) are only 4 cycles long, and if the instruction in progress is RETI or an access to IE or IP, the additional wait time cannot be more than 5 cycles (a maximum of one more cycle to complete the instruction in progress, plus 4 cycles to complete the next instruction if the instruction is MUL or DIV).

Thus, in a single-interrupt system, the response time is always more than 3 cycles and less than 9 cycles.

As previously mentioned, the derivatives described in this data sheet have a four-level interrupt structure. The corresponding registers are IE, IP and IPH. (See Figures 24, 25, and 26.) The IPH (Interrupt Priority High) register makes the four-level interrupt structure possible.

The function of the IPH SFR is simple and when combined with the IP SFR determines the priority of each interrupt. The priority of each interrupt is determined as shown in the following table:

PRIORITY BITS		
IPH.x	IP.x	
0	0	Level 0 (lowest priority)
0	1	Level 1
1	0	Level 2
1	1	Level 3 (highest priority)

### P89C60X2/61X2



Figure 30. Internal and External Data Memory Address Space with EXTRAM = 0

### HARDWARE WATCHDOG TIMER (ONE-TIME ENABLED WITH RESET-OUT FOR P89C51RA2/RB2/RC2/RD2xx)

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 disabled at reset. To enable the WDT, the user must write 01EH and 0E1H in sequence to the WDTRST, SFR location 0A6H. When the 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 the WDT overflows, it will drive an output reset HIGH pulse at the RST-pin (see the note below).

### Using the WDT

To enable the WDT, the user must write 01EH and 0E1H in sequence to the WDTRST, SFR location 0A6H. When the WDT is

enabled, the user needs to service it by writing 01EH and 0E1H to WDTRST to avoid a WDT overflow. The 14-bit counter overflows when it reaches 16383 (3FFFH) and this will reset the device. When the 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 cycles. 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 the WDT overflows, it will generate an output RESET pulse at the reset pin (see note below). The RESET pulse duration is 98 × T<sub>OSC</sub> (6-clock mode; 196 in 12-clock mode), where T<sub>OSC</sub> = 1/f<sub>OSC</sub>. 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.



Figure 33. External Data Memory Write Cycle



Figure 34. Shift Register Mode Timing



Figure 35. External Clock Drive



Figure 39. I<sub>CC</sub> Test Condition, Active Mode All other pins are disconnected

Figure 40. I<sub>CC</sub> Test Condition, Idle Mode All other pins are disconnected







Figure 42. I<sub>CC</sub> Test Condition, Power Down Mode All other pins are disconnected. V<sub>CC</sub> = 2 V to 5.5 V

### P89C60X2/61X2



### \_\_\_\_\_