Welcome to E-XFL.COM

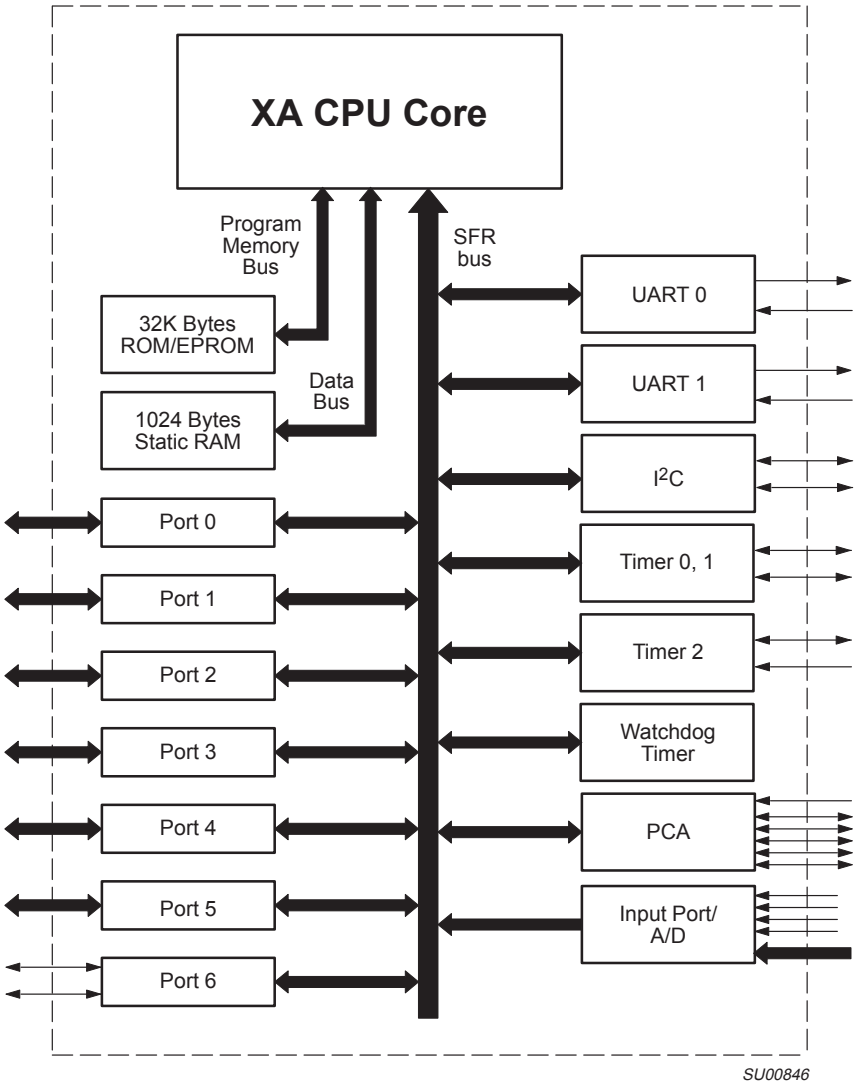
What is "Embedded - Microcontrollers"?

"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	Obsolete
Core Processor	XA
Core Size	16-Bit
Speed	30MHz
Connectivity	EBI/EMI, I²C, UART/USART
Peripherals	PWM, WDT
Number of I/O	50
Program Memory Size	-
Program Memory Type	ROMless
EEPROM Size	-
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 5.5V
Data Converters	A/D 8x8b
Oscillator Type	External
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Surface Mount
Package / Case	80-LQFP
Supplier Device Package	80-LQFP (12x12)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/pxas30kbbe-557

BLOCK DIAGRAM



NAME	DESCRIPTION	SFR Address	BIT FUNCTIONS AND ADDRESSES MSB							LSB	Reset Value
			3AF	3AE	3AD	3AC	3AB	3AA	3A9	3A8	
P5#*	Port 5	435	AD7/SDA	AD6/SCL	AD5	AD4	AD3	AD2	AD1	AD0	FFh
									3B1	3B0	
P6#*	Port 6	436	–	–	–	–	–	–	A23	A22	FFh
P0CFGA	Port 0 configuration A	470									Note 5
P1CFGA	Port 1 configuration A	471									Note 5
P2CFGA	Port 2 configuration A	472									Note 5
P3CFGA	Port 3 configuration A	473									Note 5
P4CFGA#	Port 4 configuration A	474									Note 5
P5CFGA#	Port 5 configuration A	475									Note 5
P6CFGA#	Port 6 configuration A	476	–	–	–	–	–	–			Note 5
P0CFGB	Port 0 configuration B	4F0									Note 5
P1CFGB	Port 1 configuration B	4F1									Note 5
P2CFGB	Port 2 configuration B	4F2									Note 5
P3CFGB	Port 3 configuration B	4F3									Note 5
P4CFGB#	Port 4 configuration B	4F4									Note 5
P5CFGB#	Port 5 configuration B	4F5									Note 5
P6CFGB#	Port 6 configuration B	4F6	–	–	–	–	–	–			Note 5
			227	226	225	224	223	222	221	220	
PCON*	Power control register	404	–	–	–	–	–	–	PD	IDL	00h
			20F	20E	20D	20C	20B	20A	209	208	
PSWH*	Program status word (high byte)	401	SM	TM	RS1	RS0	IM3	IM2	IM1	IM0	Note 2
			207	206	205	204	203	202	201	200	
PSWL*	Program status word (low byte)	400	C	AC	–	–	–	V	N	Z	Note 2
			217	216	215	214	213	212	211	210	
PSW51*	80C51 compatible PSW	402	C	AC	F0	RS1	RS0	V	F1	P	Note 3
RSTSRC#	Reset source register	463	–	–	–	–	–	R_WD	R_CMD	R_EXT	Note 7
RTH0	Timer 0 reload register, high byte	455									00h
RTH1	Timer 1 reload register, high byte	457									00h
RTL0	Timer 0 reload register, low byte	454									00h
RTL1	Timer 1 reload register, low byte	456									00h
			307	306	305	304	303	302	301	300	
S0CON*	Serial port 0 control register	420	SM0_0	SM1_0	SM2_0	REN_0	TB8_0	RB8_0	TI_0	RI_0	00h
			30F	30E	30D	30C	30B	30A	309	308	
S0STAT#*	Serial port 0 extended status	421	–	–	–	ERR0	FE0	BR0	OE0	STINT0	00h
S0BUF	Serial port 0 data buffer register	460									xx
S0ADDR	Serial port 0 address register	461									00h
S0ADEN	Serial port 0 address enable	462									00h
			327	326	325	324	323	322	321	320	
S1CON*	Serial port 1 control register	424	SM0_1	SM1_1	SM2_1	REN_1	TB8_1	RB8_1	TI_1	RI_1	00H
			32F	32E	32D	32C	32B	32A	329	328	
S1STAT#*	Serial port 1 extended status	425	–	–	–	ERR1	FE1	BR1	OE1	STINT1	00h
S1BUF	Serial port 1 data buffer register	464									xx
S1ADDR	Serial port 1 address register	465									00h

NAME	DESCRIPTION	SFR Address	BIT FUNCTIONS AND ADDRESSES MSB							LSB	Reset Value
S1ADEN	Serial port 1 address enable	466									00h
SCR	System configuration register	440	–	–	–	–	PT1	PT0	CM	PZ	00h
			21F	21E	21D	21C	21B	21A	219	218	
SSEL*	Segment selection register	403	ESWEN	R6SEG	R5SEG	R4SEG	R3SEG	R2SEG	R1SEG	R0SEG	00h
SWE	Software interrupt enable	47A	–	SWE7	SWE6	SWE5	SWE4	SWE3	SWE2	SWE1	00h
			357	356	355	354	353	352	351	350	
SWR*	Software interrupt request	42A	–	SWR7	SWR6	SWR5	SWR4	SWR3	SWR2	SWR1	00h
			2C7	2C6	2C5	2C4	2C3	2C2	2C1	2C0	
T2CON*	Timer 2 control register	418	TF2	EXF2	RCLK0	TCLK0	EXEN2	TR2	C/T2	CP/RL2	00h
			2CF	2CE	2CD	2CC	2CB	2CA	2C9	2C8	
T2MOD*	Timer 2 mode control	419	–	–	RCLK1	TCLK1	–	–	T2OE	DCEN	00h
TH2	Timer 2 high byte	459									00h
TL2	Timer 2 low byte	458									00h
T2CAPH	Timer 2 capture, high byte	45B									00h
T2CAPL	Timer 2 capture, low byte	45A									00h
			287	286	285	284	283	282	281	280	
TCON*	Timer 0 and 1 control register	410	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0	00h
TH0	Timer 0 high byte	451									00h
TH1	Timer 1 high byte	453									00h
TL0	Timer 0 low byte	450									00h
TL1	Timer 1 low byte	452									00h
TMOD	Timer 0 and 1 mode control	45C	GATE	C/T	M1	M0	GATE	C/T	M1	M0	00h
			28F	28E	28D	28C	28B	28A	289	288	
TSTAT*	Timer 0 and 1 extended status	411	–	–	–	–	–	T1OE	–	T0OE	00h
			2FF	2FE	2FD	2FC	2FB	2FA	2F9	2F8	
WDCON*	Watchdog control register	41F	PEW2	PRE1	PRE0	–	–	WDRUN	WDTOF	–	Note 6
WDL	Watchdog timer reload	45F									00h
WFEED1	Watchdog feed 1	45D									xx
WFEED2	Watchdog feed 2	45E									xx

NOTES:

* SFRs are bit addressable.

\# SFRs are modified from or added to XA-G3 SFRs.

1. At reset, the BCR is loaded with the binary value 00000a11, where “a’ is the value on the BUSW pin. This defaults the address bus size to 24 bits.
2. SFR is loaded from the reset vector.
3. All bits except F1, F0, and P are loaded from the reset vector. Those bits are all 0.
4. Unimplemented bits in SFRs are X (unknown) at all times. Ones should not be written to these bits since they may be used forother purposes in future XA derivatives. The reset value shown for these bits is 0.
5. Port configurations default to quasi-bidirectional when the XA begins execution from internal code memory after reset, basedon the condition found on the $\overline{EA}$ pin. Thus, all PnCFGA registers will contain FF, and PnCFGB register will contain 00 when the XA begins execution using internal code memory. When the XA begins execution using external code memory, the default configuration for pins that are associated with the external bus will be push-pull. The PnCFGA and PnCFGB register contents will reflect this difference.
6. The WDCON reset value is E6 for a Watchdog reset, E4 for all other reset causes.
7. The RSTSRC register reflects the cause of the last XA-S3 reset. One bit will be set to 1, the others will be cleared to 0.
8. The XA guards writes to certain bits (typically interrupt flags) that may be altered directly by a peripheral function. Thisprevents loss of an interrupt or other status if a bit was written directly by a peripheral action during the time between the read and write portions of an instruction that performs a read-modify-write operation. Examples of such instructions are:

```
and     s0con,#$fb
clr     tr0
setb    ti_0
```

   XA-S3 SFR bits that are guarded in this manner are: ADINT (in ADCON); CF, CCF4, CCF3, CCF2, CCF1, and CCF0 (in CCON); SI (in I2CON); TI_0 and RI_0 (in S0CON); TI_1 and RI_1 (in S1CON); FE0, BR0, and OE0 (in S0STAT); FE1, BR1, and OE1 (in S1STAT); TF2 (in T2CON); TF1, TF0, IE1, and IE0 (in TCON); and WDTOF (in WDCON).
9. The XA-S3 implements an 8-bit SFR bus, as stated in *Chapter 8* of the *XA User Guide.* All SFR accesses must be 8-bit operations. Attempts to write 16 bits to an SFR will actually write only the lower 8 bits. Sixteen bit SFR reads will return undefined data in the upper byte.

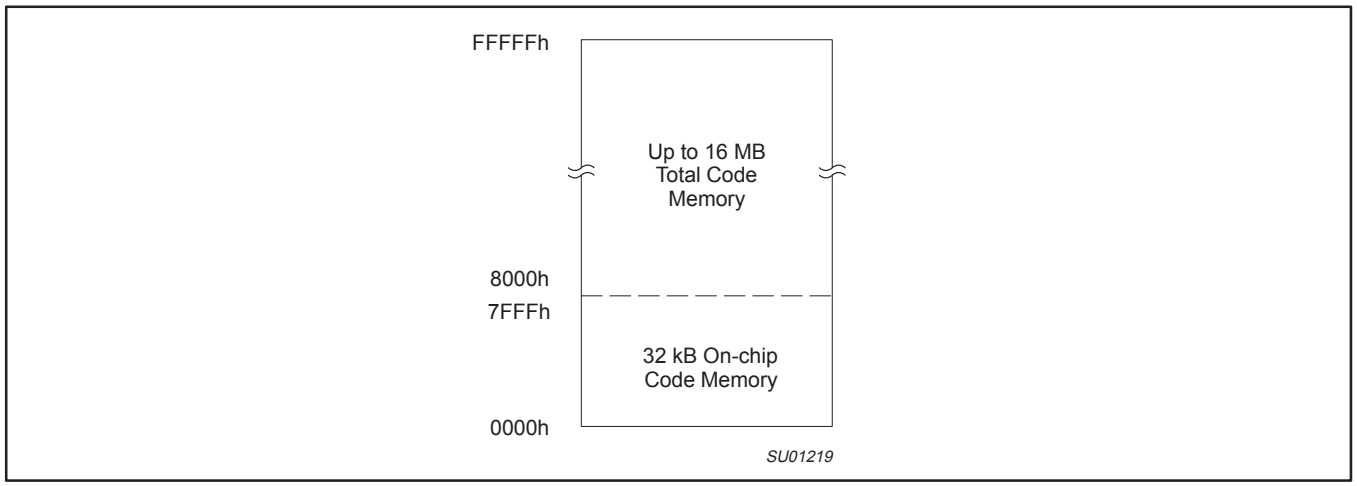


Figure 1. XA-S3 program memory map

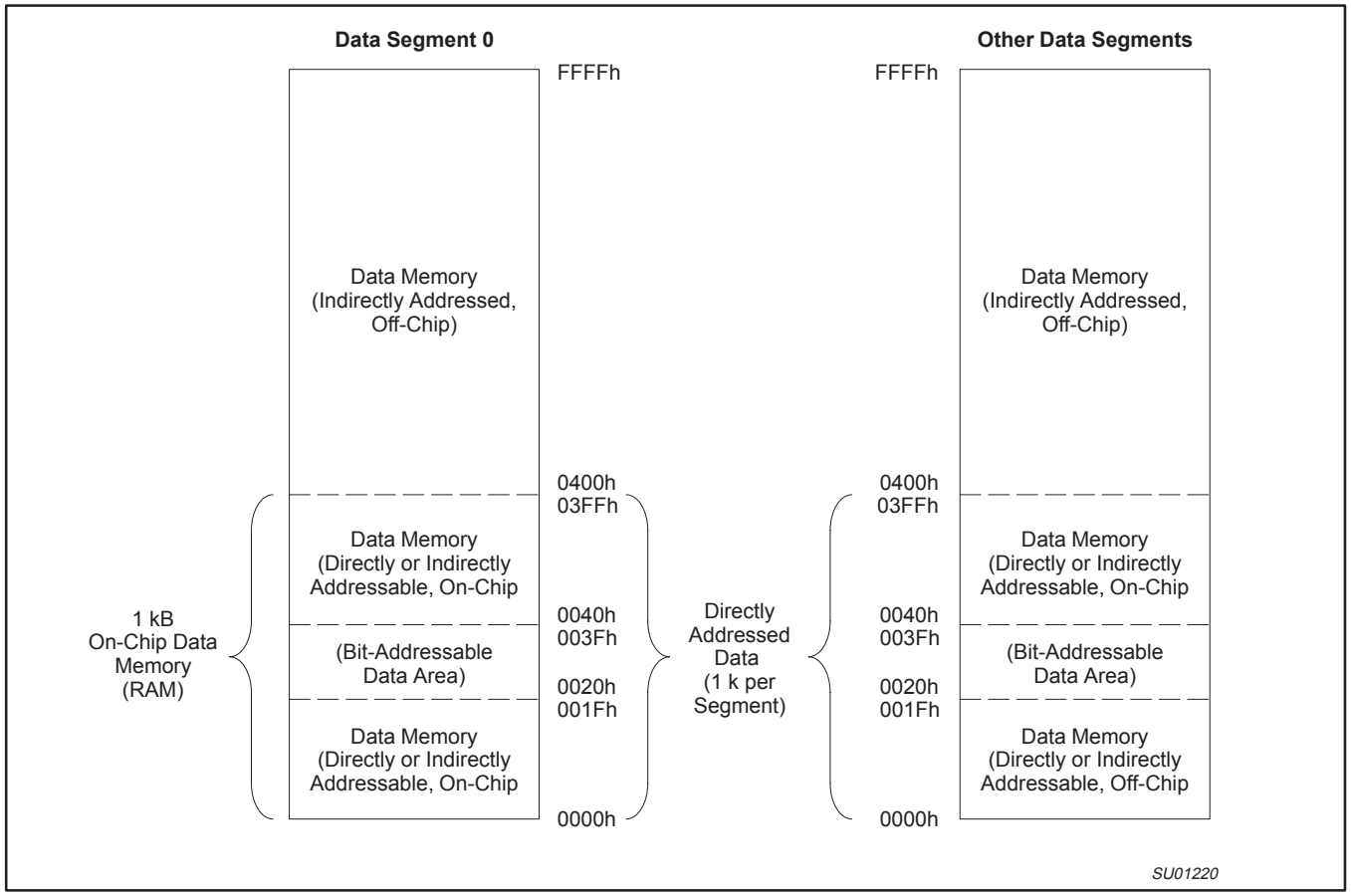


Figure 2. XA-S3 data memory map

ADCS Address:43Fh
Bit Addressable
Reset Value: 00h

MSB							LSB
ADCS7	ADCS6	ADCS5	ADCS4	ADCS3	ADCS2	ADCS1	ADCS0

BIT	SYMBOL	FUNCTION
ADCS.7	ADCS7	A/D channel 7 select bit.
ADCS.6	ADCS6	A/D channel 6 select bit.
ADCS.5	ADCS5	A/D channel 5 select bit.
ADCS.4	ADCS4	A/D channel 4 select bit.
ADCS.3	ADCS3	A/D channel 3 select bit.
ADCS.2	ADCS2	A/D channel 2 select bit.
ADCS.1	ADCS1	A/D channel 1 select bit.
ADCS.0	ADCS0	A/D channel 0 select bit.

SU00939

Figure 2. A/D Channel Select Register (ADCS)

ADCFG Address:4B9h
Not bit Addressable
Reset Value: 00h

MSB					LSB
—	—	—	—	A/D Timing Configuration	

BIT	SYMBOL	FUNCTION
ADCFG.7	—	Reserved for future use. Should not be set to 1 by user programs.
ADCFG.6	—	Reserved for future use. Should not be set to 1 by user programs.
ADCFG.5	—	Reserved for future use. Should not be set to 1 by user programs.
ADCFG.4	—	Reserved for future use. Should not be set to 1 by user programs.
ADCFG.3–0	ADCFG	A/D timing configuration (see text and table).

SU00940

Figure 3. A/D Timing Configuration Register (ADCFG)

Table 2. A/D Timing Configuration

ADCFG.3–0	Max. Oscillator Frequency (MHz)	Conversion Time		Sampling Time (Osc. Clocks)
		Osc. Clocks	µsec at max. Osc.	
0h (0000)	6.66	72	10.81	4
1h (0001)	10	76	7.6	6
2h (0010)	11.11	80	7.2	8
3h (0011)	13.33	96	7.2	8
4h (0100)	16.66	100	6.0	10
5h (0101)	20	104	5.2	12
6h (0110)[1]	20	116	5.8	24
7h (0111)	22.2	108	4.86	14
8h (1000)	23.3	124	5.32	14
9h (1001)	26.6	128	4.81	16
Ah (1010)	30	132	4.4	18
Bh (1011)[1]	30	146	4.87	32
Ch (1100)	–	136	4.25	20
Dh (1101)	–	152	4.56	20
Eh (1110)	–	172	4.7	22
Fh (1111)	–	176	4.4	24

NOTE:
1. These settings provide additional A/D input sampling time, in order to allow accurate readings with a higher external sourceimpedance.

A/D Accuracy

The XA-S3 A/D in 10 -bit mode is specified with 16 samples averaged in order to factor out on-chip noise. In an application where averaging 16 samples is not practical, the accuracy specifications may be de-rated according to the number of samples that are actually taken. The graph in Figure 5 shows the relationship of additional A/D error to the number of samples that are averaged. For example, if a single A/D reading is used with no averaging, the A/D accuracy should be de-rated by ±1.25 LSB.

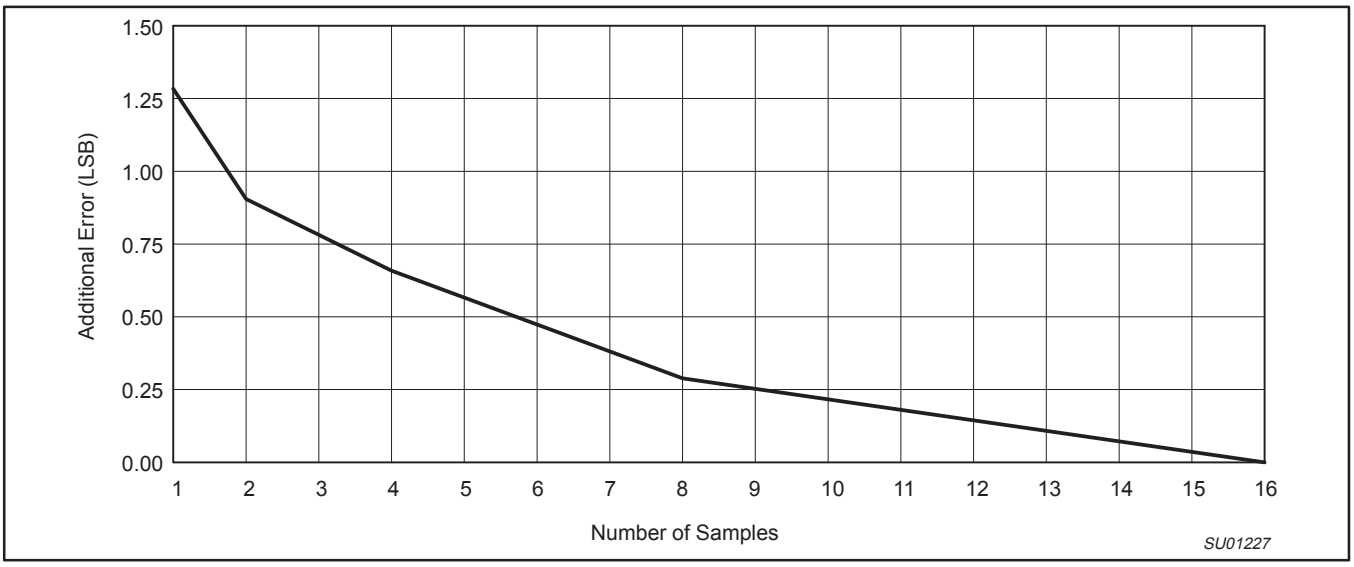


Figure 5. A/D accuracy by number of averaging samples
(Pertains to 10-bit mode only. Note that 10-bit mode is only specified up to f_C = 20 MHz.)

I2CON Address:42Ch
Bit Addressable
Reset Value: 00h

MSB							LSB
CR2	ENA	STA	STO	SI	AA	CR1	CR0

BIT	SYMBOL	FUNCTION
I2CON.7	CR2	I2C Rate Control, with CR1 and CR0. See text and table.
I2CON.6	ENA	Enable I2C port. When ENA = 1, the I2C port is enabled.
I2CON.5	STA	Start flag. Setting STA to 1 causes the I2C interface to attempt to gain mastership of the bus by generating a Start condition.
I2CON.4	STO	Stop flag. Setting STO to 1 causes the I2C interface to attempt to generate a Stop condition.
I2CON.3	SI	Serial Interrupt. SI is set by the I2C hardware when a new I2C state is entered, indicating that software needs to respond. SI causes an I2C interrupt if enabled and of sufficient priority.
I2CON.2	AA	Assert Acknowledge. Setting AA to 1 causes the I2C hardware to automatically generate acknowledge pulses for various conditions (see text).
I2CON.1	CR1	I2C Rate Control, with CR2 and CR0. See text and table.
I2CON.0	CR0	I2C Rate Control, with CR2 and CR1. See text and table.

SU00941

Figure 6. I2C Control Register (I2CON)

I2C Interface

The I2C interface on the XA-S3 is identical to the standard byte-style I2C interface found on devices such as the 8xC552 except for the rate selection. ***The I2C interface conforms to the 100 kHz I2C specification, but may be used at rates up to 400 kHz (non-conforming).***

Important: Before the I2C interface may be used, the port pins P5.6 and 5.7, which correspond to the I2C functions SCL and SDA respectively, must be set to the open drain mode.

The processor interfaces to the I2C logic via the following four special function registers: I2CON (I2C control register), I2STA (I2C status register), I2DAT (I2C data register), and I2ADR (I2C slave address register). The I2C control logic interfaces to the external I2C bus via two port 5 pins: P5.6/SCL (serial clock line) and P5.7/SDA (serial data line).

The Control Register, I2CON

This register is shown in Figure 6. Two bits are affected by the I2C hardware: the SI bit is set when a serial interrupt is requested, and the STO bit is cleared when a STOP condition is present on the I2C bus. The STO bit is also cleared when ENA = “0”.

ENA, the I2C Enable Bit

ENA = 0: When ENA is “0”, the SDA and SCL outputs are not driven. SDA and SCL input signals are ignored, SIO1 is in the “not addressed” slave state, and the STO bit in I2CON is forced to “0”. No other bits are affected. P5.6 and P5.7 may be used as open drain I/O ports.

ENA = 1: When ENA is “1”, SIO1 is enabled. The P5.6 and P5.7 port latches must be set to logic 1.

ENA should not be used to temporarily release the I2C-bus since, when ENA is reset, the I2C-bus status is lost. The AA flag should be used instead (see description of the AA flag in the following text).

In the following text, it is assumed the ENA = “1”.

STA, the START flag

STA = 1: When the STA bit is set to enter a master mode, the I2C hardware checks the status of the I2C bus and generates a START condition if the bus is free. If the bus is not free, the I2C interface waits for a STOP condition (which will free the bus) and generates a START condition after a delay of a half clock period of the internal serial clock generator.

If STA is set while the I2C interface is already in a master mode and one or more bytes are transmitted or received, the hardware transmits a repeated START condition. STA may be set at any time. STA may also be set when the I2C interface is an addressed slave.

STA = 0: When the STA bit is reset, no START condition or repeated START condition will be generated.

STO, the STOP flag

STO = 1: When the STO bit is set while the I2C interface is in a master mode, a STOP condition is transmitted to the I2C bus. When the STOP condition is detected on the bus, the hardware clears the STO flag. In a slave mode, the STO flag may be set to recover from an error condition. In this case, no STOP condition is transmitted to the I2C bus. However, the hardware behaves as if a STOP condition has been received and switches to the defined “not addressed” slave receiver mode. The STO flag is automatically cleared by hardware.

If the STA and STO bits are both set, then a STOP condition is transmitted to the I2C bus if the interface is in a master mode (in a slave mode, the hardware generates an internal STOP condition which is not transmitted). The I2C interface then transmits a START condition.

STO = 0: When the STO bit is reset, no STOP condition will be generated.

SI, the Serial Interrupt flag

SI = 1: When the SI flag is set, and the EA (interrupt system enable) and EI2 (I2C interrupt enable) bits are also set, an I2C interrupt is requested. SI is set by hardware when one of 25 of the 26 possible I2C interface states is entered. The only state that does not cause SI to be set is state F8H, which indicates that no relevant state information is available.

While SI is set, the low period of the serial clock on the SCL line is stretched, and the serial transfer is suspended. A high level on the SCL line is unaffected by the serial interrupt flag. SI must be reset by software.

SI = 0: When the SI flag is reset, no serial interrupt is requested, and there is no stretching of the serial clock on the SCL line.

AA, the Assert Acknowledge flag

AA = 1: If the AA flag is set, an acknowledge (low level to SDA) will be returned during the acknowledge clock pulse on the SCL line when:

- The “own slave address” has been received.
- The general call address has been received while the general call bit (GC) in I2ADR is set.
- A data byte has been received while the I2C interface is in the master receiver mode.
- A data byte has been received while the I2C interface is in the addressed slave receiver mode.

AA = 0: If the AA flag is reset, a not acknowledge (high level to SDA) will be returned during the acknowledge clock pulse on the SCL line when:

- A data byte has been received while the I2C interface is in the master receiver mode.
- A data byte has been received while the I2C interface is in the addressed slave receiver mode.

When the I2C interface is in the addressed slave transmitter mode, state C8H will be entered after the last serial data byte is transmitted. When SI is cleared, the I2C interface leaves state C8H, enters the not addressed slave receiver mode, and the SDA line remains at a high level. In state C8H, the AA flag can be set again for future address recognition.

When the I2C interface is in the not addressed slave mode, its own slave address and the general call address are ignored. Consequently, no acknowledge is returned, and a serial interrupt is not requested. Thus, the hardware can be temporarily released from the I2C bus while the bus status is monitored. While the hardware is released from the bus, START and STOP conditions are detected, and serial data is shifted in. Address recognition can be resumed at any time by setting the AA flag. If the AA flag is set when the part’s own slave address or the general call address has been partly received, the address will be recognized at the end of the byte transmission.

CCAPMn Address CCAPM0 491H
CCAPM1 492H
CCAPM2 493H
CCAPM3 494H
CCAPM4 495H

Reset Value = 00H

Not Bit Addressable

–	ECOMn	CAPPn	CAPNn	MATn	TOGn	PWMn	ECCFn
7	6	5	4	3	2	1	0

Bit:

Symbol	Function
–	Not implemented, reserved for future use*.
ECOMn	Enable Comparator. ECOMn = 1 enables the comparator function.
CAPPn	Capture Positive, CAPPn = 1 enables positive edge capture.
CAPNn	Capture Negative, CAPNn = 1 enables negative edge capture.
MATn	Match. When MATn = 1, a match of the PCA counter with this module's compare/capture register causes the CCFn bit in CCON to be set, flagging an interrupt.
TOGn	Toggle. When TOGn = 1, a match of the PCA counter with this module's compare/capture register causes the CEXn pin to toggle.
PWMn	Pulse Width Modulation Mode. PWMn = 1 enables the CEXn pin to be used as a pulse width modulated output.
ECCFn	Enable CCF interrupt. Enables compare/capture flag CCFn in the CCON register to generate an interrupt.

NOTE:
*User software should not write 1s to reserved bits. These bits may be used in future products to invoke new features. In that case, the reset or inactive value of the new bit will be 0, and its active value will be 1. The value read from a reserved bit is indeterminate.

SU01308

Figure 12. CCAPMn: PCA Modules Compare/Capture Registers

–	ECOMn	CAPPn	CAPNn	MATn	TOGn	PWMn	ECCFn	MODULE FUNCTION
X	0	0	0	0	0	0	0	No operation
X	X	1	0	0	0	0	X	16-bit capture by a positive-edge trigger on CEXn
X	X	0	1	0	0	0	X	16-bit capture by a negative trigger on CEXn
X	X	1	1	0	0	0	X	16-bit capture by a transition on CEXn
X	1	0	0	1	0	0	X	16-bit Software Timer
X	1	0	0	1	1	0	X	16-bit High Speed Output
X	1	0	0	0	0	1	0	8-bit PWM
X	1	0	0	1	X	0	X	Watchdog Timer

Figure 13. PCA Module Modes (CCAPMn Register)

PCA Capture Mode
To use one of the PCA modules in the capture mode either one or both of the CCAPM bits CAPN and CAPP for that module must be set. The external CEX input for the module (on port 1) is sampled for a transition. When a valid transition occurs the PCA hardware loads the value of the PCA counter registers (CH and CL) into the module's capture registers (CCAPnL and CCAPnH). If the CCFn bit for the module in the CCON SFR and the ECCFn bit in the CCAPMn SFR are set then an interrupt will be generated. Refer to Figure 14.

16-bit Software Timer Mode
The PCA modules can be used as software timers by setting both the ECOM and MAT bits in the modules CCAPMn register. The PCA timer will be compared to the module's capture registers and when a match occurs an interrupt will occur if the CCFn (CCON SFR) and the ECCFn (CCAPMn SFR) bits for the module are both set (see Figure 15).

High Speed Output Mode
In this mode the CEX output (on port 4) associated with the PCA module will toggle each time a match occurs between the PCA counter and the module's capture registers. To activate this mode the TOG, MAT, and ECOM bits in the module's CCAPMn SFR must be set (see Figure 16).

Pulse Width Modulator Mode
All of the PCA modules can be used as PWM outputs. Figure 17 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. the 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.

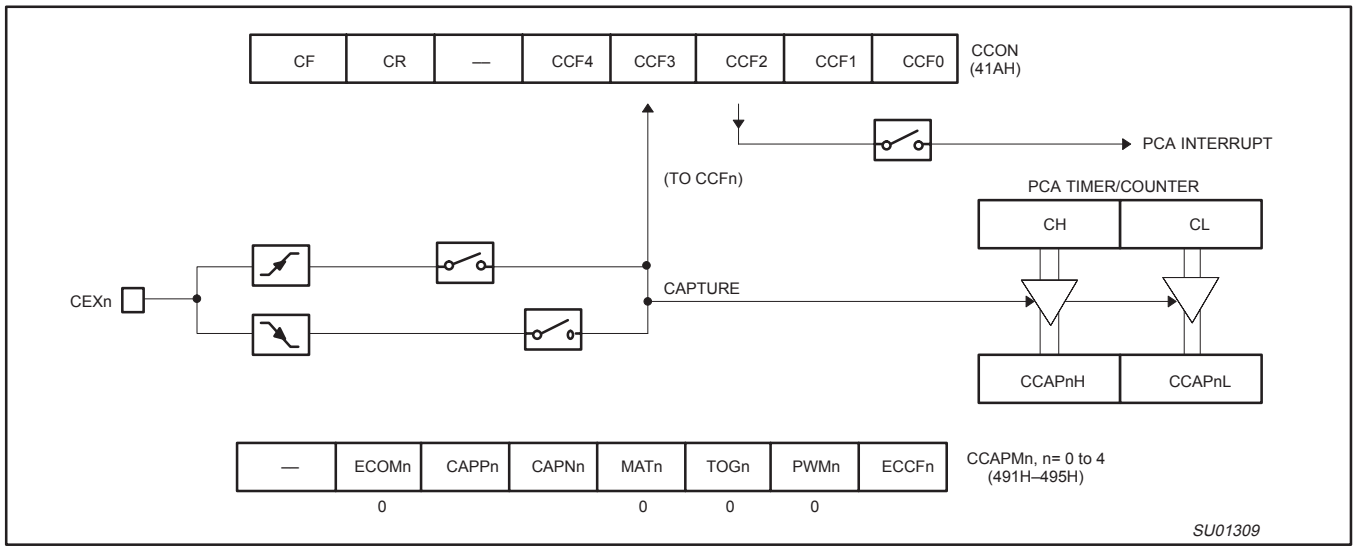


Figure 14. PCA Capture Mode

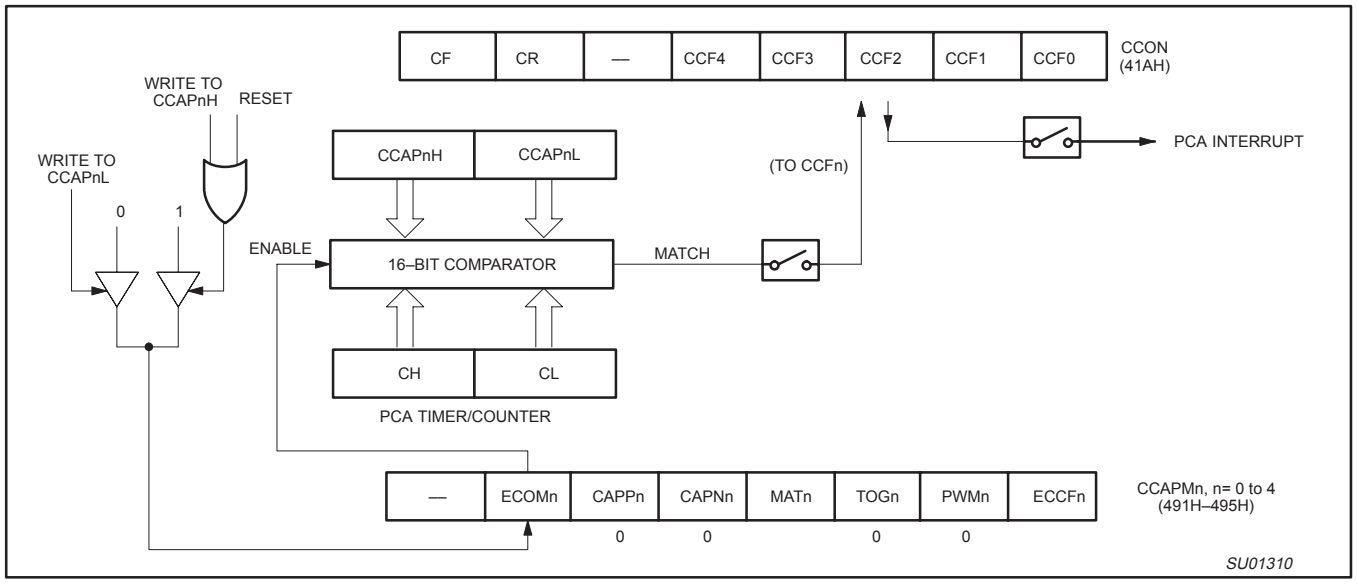


Figure 15. PCA Compare Mode

```
INIT_WATCHDOG:
   MOV CCAPM4, #4CH         ; Module 4 in compare mode
   MOV CCAP4L, #0FFH        ; Write to low byte first
   MOV CCAP4H, #0FFH        ; Before PCA timer counts up to
                            ; FFFF Hex, these compare values
                            ; must be changed
   OR CMOD, #40H            ; Set the WDTE bit to enable the
                            ; watchdog timer without changing
                            ; the other bits in CMOD
;
;*********************************************************************
;
; Main program goes here, but CALL WATCHDOG periodically.
;
;*********************************************************************
;
WATCHDOG:
   CLR EA                   ; Hold off interrupts
   MOV CCAP4L, #00          ; Next compare value is within
   MOV CCAP4H, CH           ; 255 counts of the current PCA
   SETB EA                  ; timer value
   RET
```

Figure 19. PCA Watchdog Timer Initialization Code

Watchdog Timer

This is a standard XA-G3 watchdog timer. This watchdog timer always comes up running at reset. The watchdog acts the same on EPROM, ROM, and ROMless parts, as in the XA-G3.

UARTs

Standard XA-S3 UART0 and UART1 with double buffered transmit register. A flag has been added to SnSTAT that is set if any of the status flags (BRn, FEn, or OEn) is set for the corresponding UART channel. This allows polling for UART errors quickly at the interrupt service routine. Baud rate sources may be timer 1 or timer 2.

The XA-S3 includes 2 UART ports that are compatible with the enhanced UART used on the XA-G3.

The UART has separate interrupt vectors for each UART's transmit and receive functions. The UART transmitter has been double buffered, allowing packed transmission of data with no gaps between bytes and less critical interrupt service routine timing. A break detect function has been added to the UART. This operates independently of the UART itself and provides a start-of-break status bit that the program may test. An Overrun Error flag allows detection of missed characters in the received data stream. The double buffered UART transmitter may require some software changes if code is used that was written for the original XA-G3 single buffered UART.

Each UART baud rate is determined by either a fixed division of the oscillator (in UART modes 0 and 2) or by the timer 1 or timer 2 overflow rate (in UART modes 1 and 3).

Timer 1 defaults to clock both UART0 and UART1. Timer 2 can be programmed to clock either UART0 through T2CON (via bits R0CLK and T0CLK) or UART1 through T2MOD (via bits R1CLK and T1CLK). In this case, the UART not clocked by T2 could use T1 as the clock source.

The serial port receive and transmit registers are both accessed at Special Function Register SnBUF. Writing to SnBUF loads the transmit register, and reading SnBUF accesses a physically separate receive register.

The serial port can operate in 4 modes:

Mode 0: Serial I/O expansion mode. Serial data enters and exits through RxDn. TxDn outputs the shift clock. 8 bits are transmitted/received (LSB first). (The baud rate is fixed at 1/16 the oscillator frequency.)

Mode 1: Standard 8-bit UART mode. 10 bits are transmitted (through TxDn) or received (through RxDn): 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 SnCON. The baud rate is variable.

Mode 2: Fixed rate 9-bit UART mode. 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_n in SnCON) can be assigned the value of 0 or 1. Or, for example, the parity bit (P, in the PSW) could be moved into TB8_n. On receive, the 9th data bit goes into RB8_n in Special Function Register SnCON, while the stop bit is ignored. The baud rate is programmable to 1/32 of the oscillator frequency.

Mode 3: Standard 9-bit UART mode. 11 bits are transmitted (through TxDn) or received (through RxDn): 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 SnBUF as a destination register. Reception is initiated in Mode 0 by the condition RI_n = 0 and REN_n = 1. Reception is initiated in the other modes by the incoming start bit if REN_n = 1.

SnSTAT Address: S0STAT 421
S1STAT 425
Bit Addressable
Reset Value: 00H

MSB							LSB
—	—	—	—	FEn	BRn	OEn	STINTn

BIT	SYMBOL	FUNCTION
SnSTAT.3	FEn	Framing Error flag is set when the receiver fails to see a valid STOP bit at the end of the frame. Cleared by software.
SnSTAT.2	BRn	Break Detect flag is set if a character is received with all bits (including STOP bit) being logic '0'. Thus it gives a "Start of Break Detect" on bit 8 for Mode 1 and bit 9 for Modes 2 and 3. The break detect feature operates independently of the UARTs and provides the START of Break Detect status bit that a user program may poll. Cleared by software.
SnSTAT.1	OEn	Overrun Error flag is set if a new character is received in the receiver buffer while it is still full (before the software has read the previous character from the buffer), i.e., when bit 8 of a new byte is received while RI in SnCON is still set. Cleared by software.
SnSTAT.0	STINTn	This flag must be set to enable any of the above status flags to generate a receive interrupt (RIn). The only way it can be cleared is by a software write to this register.

SU00607B

Figure 20. Serial Port Extended Status (SnSTAT) Register (See also Figure 22 regarding Framing Error flag)

UART INTERRUPT SCHEME

There are separate interrupt vectors for each UART's transmit and receive functions.

Table 5. Interrupt Vector Locations for UARTs

Vector Address	Interrupt Source	Arbitration
A0H – A3H	UART 0 Receiver	9
A4H – A7H	UART 0 Transmitter	10
A8H – ABH	UART 1 Receiver	11
ACH – AFH	UART 1 Transmitter	12

NOTE:
The transmit and receive vectors could contain the same ISR address to work like a 8051 interrupt scheme

Error Handling, Status Flags and Break Detect
XA UARTs have several error flags as described in Figures 20 and 22.

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 although this is better done with the Framing Error (FE) flag. In a Mode 1 reception, if SM2 = 1, the receive interrupt will not be activated unless a valid stop bit is received.

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 23.

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	=	1111 1101
	Given	=	1100 00X0
Slave 1	SADDR	=	1100 0000
	SADEN	=	1111 1110
	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

RSTSRC Address:463h
Not bit Addressable
Reset Value: see below

MSB							LSB
—	—	—	—	—	R_WD	R_CMD	R_EXT

BIT	SYMBOL	FUNCTION
RSTSRC.7	—	Reserved for future use. Should not be set to 1 by user programs.
RSTSRC.6	—	Reserved for future use. Should not be set to 1 by user programs.
RSTSRC.5	—	Reserved for future use. Should not be set to 1 by user programs.
RSTSRC.4	—	Reserved for future use. Should not be set to 1 by user programs.
RSTSRC.3	—	Reserved for future use. Should not be set to 1 by user programs.
RSTSRC.2	R_WD	Indicates that the last reset was caused by a watchdog timer overflow.
RSTSRC.1	R_CMD	Indicates that the last reset was caused by execution of the RESET instruction.
RSTSRC.0	R_EXT	Indicates that the last reset was caused by the external $\overline{RST}$ input.

SU00942

Figure 24. Reset source register (RSTSRC)

INTERRUPTS

XA-S3 interrupt sources include the following:

- External interrupts 0 and 1 (2)
- Timer 0, 1, and 2 interrupts (3)
- PCA: 1 global and 5 channel interrupts (6)
- A/D interrupt (1)
- UART 0 transmitter and receiver interrupts (2)
- UART 1 transmitter and receiver interrupts (2)
- I2C interrupt (1)
- Software interrupts (7)

There are a total of 17 **hardware** interrupt sources, enable bits, priority bit sets, etc.

The XA-S3 supports a total of 17 maskable event interrupt sources (for the various XA peripherals), seven software interrupts, 5 exception interrupts (plus reset), and 16 traps. The maskable event interrupts share a global interrupt disable bit (the EA bit in the IEL register) and each also has a separate individual interrupt enable bit (in the IEL or IEH registers). Only three bits of the IPA register values are used on the XA-S3. Each event interrupt can be set to occur at one of 8 priority levels via bits in the Interrupt Priority (IP) registers, IPA0 through IPA5. The value 0 in the IPA field gives the interrupt priority 0, in effect disabling the interrupt. A value of 1 gives the interrupt a priority of 9, the value 2 gives priority 10, etc. The result is the same as if all four bits were used and the top bit set for all values except 0. Details of the priority scheme may be found in the *XA User Guide*.

The complete interrupt vector list for the XA-S3, including all 4 interrupt types, is shown in the following tables. The tables include the address of the vector for each interrupt, the related priority register bits (if any), and the arbitration ranking for that interrupt source. The arbitration ranking determines the order in which interrupts are processed if more than one interrupt of the same priority occurs simultaneously.

EXCEPTION/TRAPS PRECEDENCE

DESCRIPTION	VECTOR ADDRESS	ARBITRATION RANKING
Reset (h/w, watchdog, s/w)	0000–0003	0 (High)
Breakpoint	0004–0007	1
Trace	0008–000B	1
Stack Overflow	000C–000F	1
Divide by 0	0010–0013	1
User RETI	0014–0017	1
TRAP 0–15 (software)	0040–007F	1

EVENT INTERRUPTS

DESCRIPTION	FLAG BIT	VECTOR ADDRESS	ENABLE BIT	INTERRUPT PRIORITY	ARBITRATION RANKING
External Interrupt 0	IE0	0080–0083	EX0	IPA0.2–0 (PX0)	2
Timer 0 Interrupt	TF0	0084–0087	ET0	IPA0.6–4 (PT0)	3
External Interrupt 1	IE1	0088–008B	EX1	IPA1.2–0 (PX1)	4
Timer 1 Interrupt	TF1	008C–008F	ET1	IPA1.6–4 (PT1)	5
Timer 2 Interrupt	TF2 (EXF2)	0090–0093	ET2	IPA2.2–0 (PT2)	6
PCA Interrupt	CCF0–CCF4, CF	0094–0097	EPC	IPA2.6–4 (PPC)	7
A/D Interrupt	ADINT	0098–009B	EAD	IPA3.2–0 (PAD)	8
Serial Port 0 Rx	RI_0	00A0–00A3	ERI0	IPA4.2–0 (PRI0)	9
Serial Port 0 Tx	TI_0	00A4–00A7	ETI0	IPA4.6–4 (PTI0)	10
Serial Port 1 Rx	RI_1	00A8–00AB	ERI1	IPA5.2–0 (PRI1)	11
Serial Port 1 Tx	TI_1	00AC–00AF	ETI1	IPA5.6–4 (PTI1)	12
PCA channel 0	CCF0	00C0–00C3	EC0	IPB0.2–0 (PC0)	17
PCA channel 1	CCF1	00C4–00C7	EC1	IPB0.6–4 (PC1)	18
PCA channel 2	CCF2	00C8–00CB	EC2	IPB1.2–0 (PC2)	19
PCA channel 3	CCF3	00CC–00CF	EC3	IPB1.6–4 (PC3)	20
PCA channel 4	CCF4	00D0–00D3	EC4	IPB2.2–0 (PC4)	21
I^2C Interrupt	SI	00D4–00D7	EI2	IPB2.6–4 (PI2)	22

SOFTWARE INTERRUPTS

DESCRIPTION	FLAG BIT	VECTOR ADDRESS	ENABLE BIT	INTERRUPT PRIORITY
Software Interrupt 1	SWR1	0100–0103	SWE1	(fixed at 1)
Software Interrupt 2	SWR2	0104–0107	SWE2	(fixed at 2)
Software Interrupt 3	SWR3	0108–010B	SWE3	(fixed at 3)
Software Interrupt 4	SWR4	010C–010F	SWE4	(fixed at 4)
Software Interrupt 5	SWR5	0110–0113	SWE5	(fixed at 5)
Software Interrupt 6	SWR6	0114–0117	SWE6	(fixed at 6)
Software Interrupt 7	SWR7	0118–011B	SWE7	(fixed at 7)

AC ELECTRICAL CHARACTERISTICS (5 V RANGE) (continued)

This set of parameters is referenced to the XA-S3 clock output.

SYMBOL	FIGURE	PARAMETER	LIMITS		UNIT
			MIN	MAX	
Address Cycle					
t_{CHLH}	26	CLKOUT rising edge to ALE rising edge	–	13	ns
t_{CLLL}	26	CLKOUT falling edge to ALE falling edge	–	9	ns
t_{CHAV}	26	CLKOUT rising edge to address valid	–	18	ns
t_{CHAX}	26	CLKOUT rising edge to address changing (hold time)	2	–	ns
Code Read Cycle					
t_{CHPL}	26	CLKOUT rising edge to $\overline{PSEN}$ asserted	–	14	ns
t_{CHPH}	26	CLKOUT rising edge to $\overline{PSEN}$ de-asserted	–	12	ns
t_{IVCH}	26	Instruction valid to CLKOUT rising edge (setup time)	20	–	ns
t_{CHIX}	26	CLKOUT rising edge to instruction changing (hold time)	0	–	ns
t_{CHIZ}	26	CLKOUT rising edge to Bus 3-State (code read)	–	t_C–8	ns
Data Read Cycle					
t_{CHRL}	28	CLKOUT rising edge to $\overline{RD}$ asserted	–	12	ns
t_{CHRH}	28	CLKOUT rising edge to $\overline{RD}$ de-asserted	–	10	ns
t_{DVCH}	28	Data valid to CLKOUT rising edge (setup time)	20	–	ns
t_{CHDX}	28	CLKOUT rising edge to Data changing (hold time)	0	–	ns
t_{CHDZ}	28	CLKOUT rising edge to Bus 3-State (data read)	–	t_C–8	ns
Data Write Cycle					
t_{CHWL}	30	CLKOUT falling edge to $\overline{WR}$ asserted	–	12	ns
t_{CHWH}	30	CLKOUT rising edge to $\overline{WR}$ de-asserted	–	10	ns
t_{QVCH}	30	Data valid to CLKOUT rising edge (setup time)	4	–	ns
t_{CHQX}	30	CLKOUT rising edge to Data changing (hold time)	0	–	ns
Wait Input					
t_{CHWTH}	31	WAIT valid prior to CLKOUT rising edge[8]	21	4	ns

NOTES ON PAGE 41.

AC ELECTRICAL CHARACTERISTICS (3 V)

V_{DD} = 2.7 V to 4.5 V; T_{amb} = 0 to +70°C for commercial, T_{amb} = –40°C to +85°C for industrial.

SYMBOL	FIGURE	PARAMETER	LIMITS		UNIT
			MIN	MAX	
Address Cycle					
t_{LHLL}	26, 28, 30	ALE pulse width (programmable)	$(V1 * t_C) – 10$		ns
t_{AVLL}	26, 28, 30	Address valid to ALE de-asserted (set-up)	$(V1 * t_C) – 18$		ns
t_{LLAX}	26, 28, 30	Address hold after ALE de-asserted	$(t_C/2) – 12$		ns
Code Read Cycle					
t_{PLPH}	26	$\overline{PSEN}$ pulse width	$(V2 * t_C) – 12$		ns
t_{LLPL}	26	ALE de-asserted to $\overline{PSEN}$ asserted	$(t_C/2) – 9$		ns
t_{AVIVA}	26	Address valid to instruction valid, ALE cycle (access time)		$(V3 * t_C) – 58$	ns
t_{AVIVB}	27	Address valid to instruction valid, non-ALE cycle (access time)		$(V4 * t_C) – 52$	ns
t_{PLIV}	26	$\overline{PSEN}$ asserted to instruction valid (enable time)		$(V2 * t_C) – 52$	ns
t_{PHIX}	26	Instruction hold after $\overline{PSEN}$ de-asserted	0		ns
t_{PHIZ}	26	Bus 3-State after $\overline{PSEN}$ de-asserted		$t_C – 8$	ns
t_{IXUA}	26	Hold time of unlatched part of address after instruction latched	0		ns
Data Read Cycle					
t_{RLRH}	28	$\overline{RD}$ pulse width	$(V7 * t_C) – 12$		ns
t_{LLRL}	28	ALE de-asserted to $\overline{RD}$ asserted	$(t_C/2) – 9$		ns
t_{AVDVA}	28	Address valid to data input valid, ALE cycle (access time)		$(V6 * t_C) – 58$	ns
t_{AVDVB}	29	Address valid to data input valid, non-ALE cycle (access time)		$(V5 * t_C) – 52$	ns
t_{RLDV}	28	$\overline{RD}$ low to valid data in (enable time)		$(V7 * t_C) – 52$	ns
t_{RHDX}	28	Data hold time after $\overline{RD}$ de–asserted	0		ns
t_{RHDZ}	28	Bus 3-State after $\overline{RD}$ de-asserted (disable time)		$t_C – 8$	ns
t_{DXUA}	28	Hold time of unlatched part of address after data latched	0		ns
Data Write Cycle					
t_{WLWH}	30	$\overline{WR}$ pulse width	$(V8 * t_C) – 12$		ns
t_{LLWL}	30	ALE falling edge to $\overline{WR}$ asserted	$(V12 * t_C) – 10$		ns
t_{QVWX}	30	Data valid before $\overline{WR}$ asserted (data set-up time)	$(V13 * t_C) – 28$		ns
t_{WHQX}	30	Data hold time after $\overline{WR}$ de-asserted (Note 6)	$(V11 * t_C) – 8$		ns
t_{AVWL}	30	Address valid to $\overline{WR}$ asserted (address set-up time) (Note 5)	$(V9 * t_C) – 28$		ns
t_{UAWH}	30	Hold time of unlatched part of address after $\overline{WR}$ is de-asserted	$(V11 * t_C) – 10$		ns
Wait Input					
t_{WTH}	31	WAIT stable after bus strobe ($\overline{RD}$, $\overline{WR}$, or $\overline{PSEN}$) asserted		$(V10 * t_C) – 40$	ns
t_{WTL}	31	WAIT hold after bus strobe ($\overline{RD}$, $\overline{WR}$, or $\overline{PSEN}$) asserted	$(V10 * t_C) – 5$		ns

NOTES ON PAGE 41.

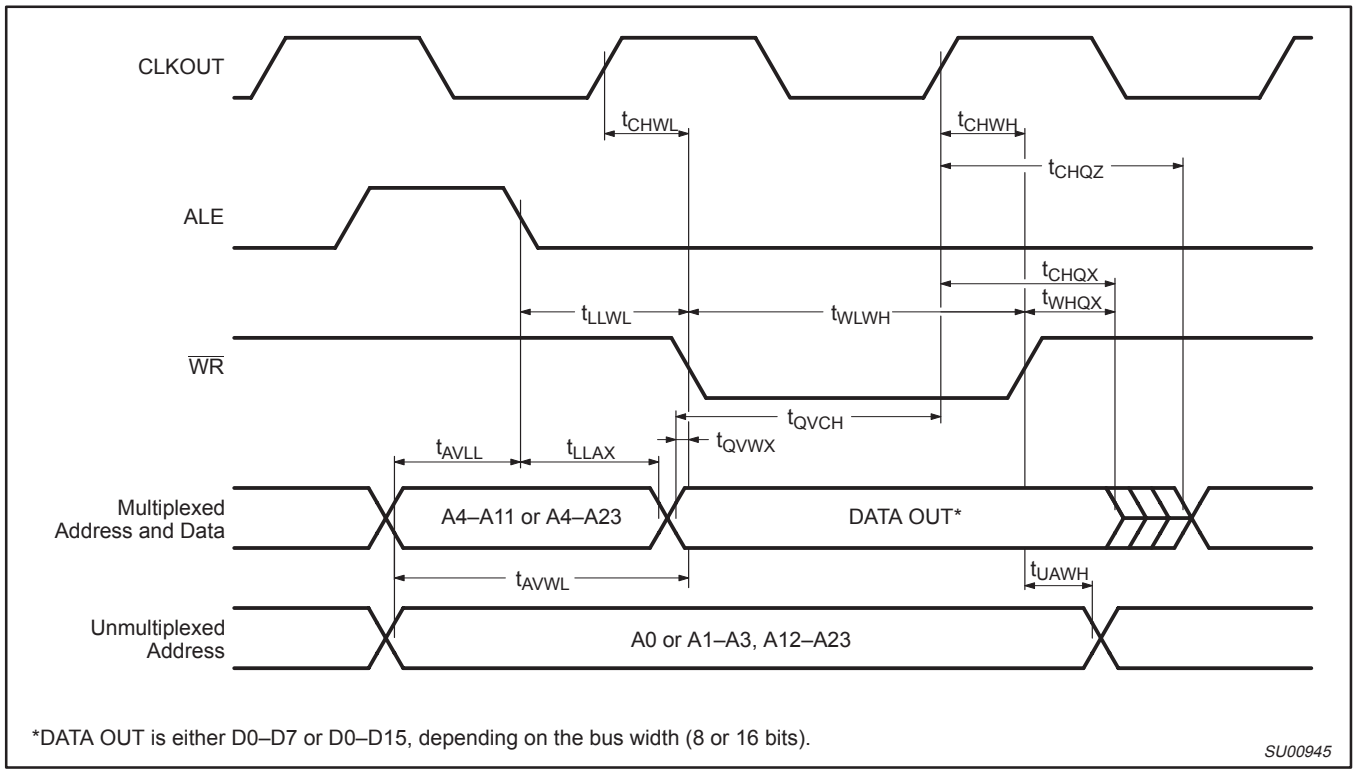


Figure 30. External Data Memory Write Cycle

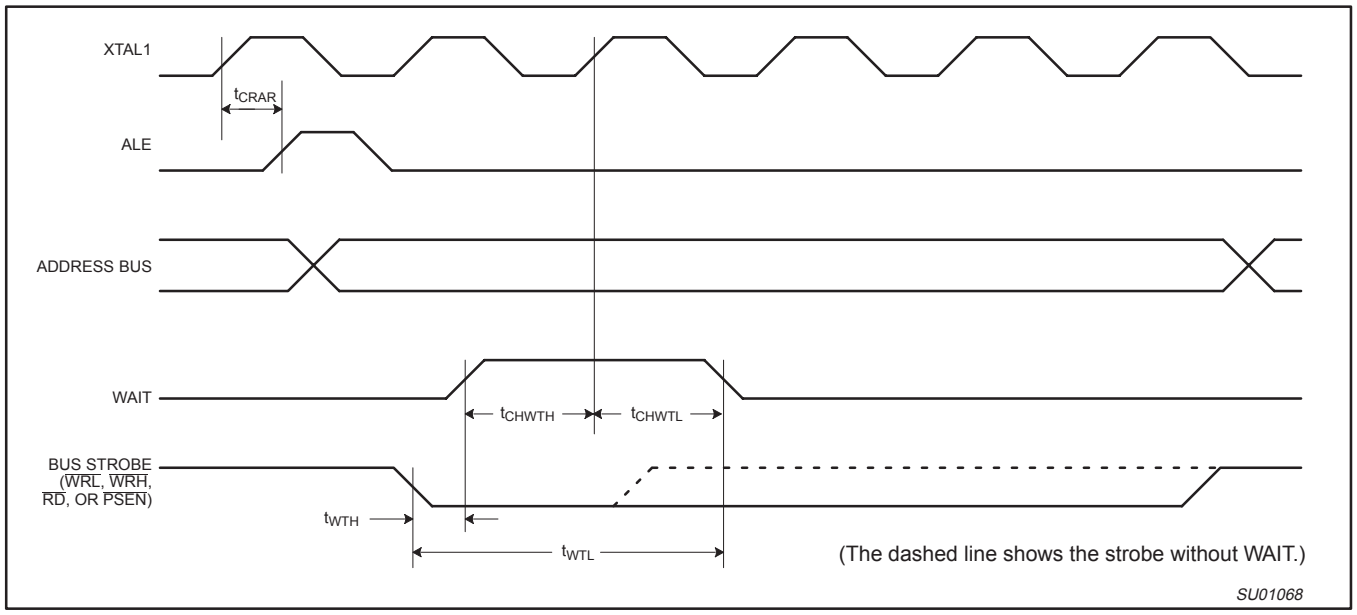


Figure 31. WAIT Signal Timing

EPROM CHARACTERISTICS

The XA-S3 is programmed by using a modified Improved Quick-Pulse Programming™ algorithm. This algorithm is essentially the same as that used by 80C51 family EPROM parts. However different pins are used for many programming functions.

The XA-S3 contains three signature bytes that can be read and used by an EPROM programming system to identify the device. The signature bytes identify the device as an XA-S3 manufactured by Philips.

Security Bits

With none of the security bits programmed the code in the program memory can be verified. When only security bit 1 is programmed, MOVC instructions executed from external program memory are disabled from fetching code bytes from the internal memory. All further programming of the EPROM is disabled. When security bits 1 and 2 are programmed, in addition to the above, verify mode is disabled. When all three security bits are programmed, all of the conditions above apply and all external program memory execution is disabled. (See Table 6.)

Table 6. Program Security Bits

PROGRAM LOCK BITS				
	SB1	SB2	SB3	PROTECTION DESCRIPTION
1	U	U	U	No Program Security features enabled.
2	P	U	U	MOVC instructions executed from external program memory are disabled from fetching code bytes from internal memory and further programming of the EPROM is disabled.
3	P	P	U	Same as 2, also verify is disabled.
4	P	P	P	Same as 3, external execution is disabled. Internal data RAM is not accessible.

NOTES:
1. P – programmed. U – unprogrammed.
2. Any other combination of the security bits is not defined.

ROM CODE SUBMISSION

When submitting ROM code for the XA-S3, the following must be specified:

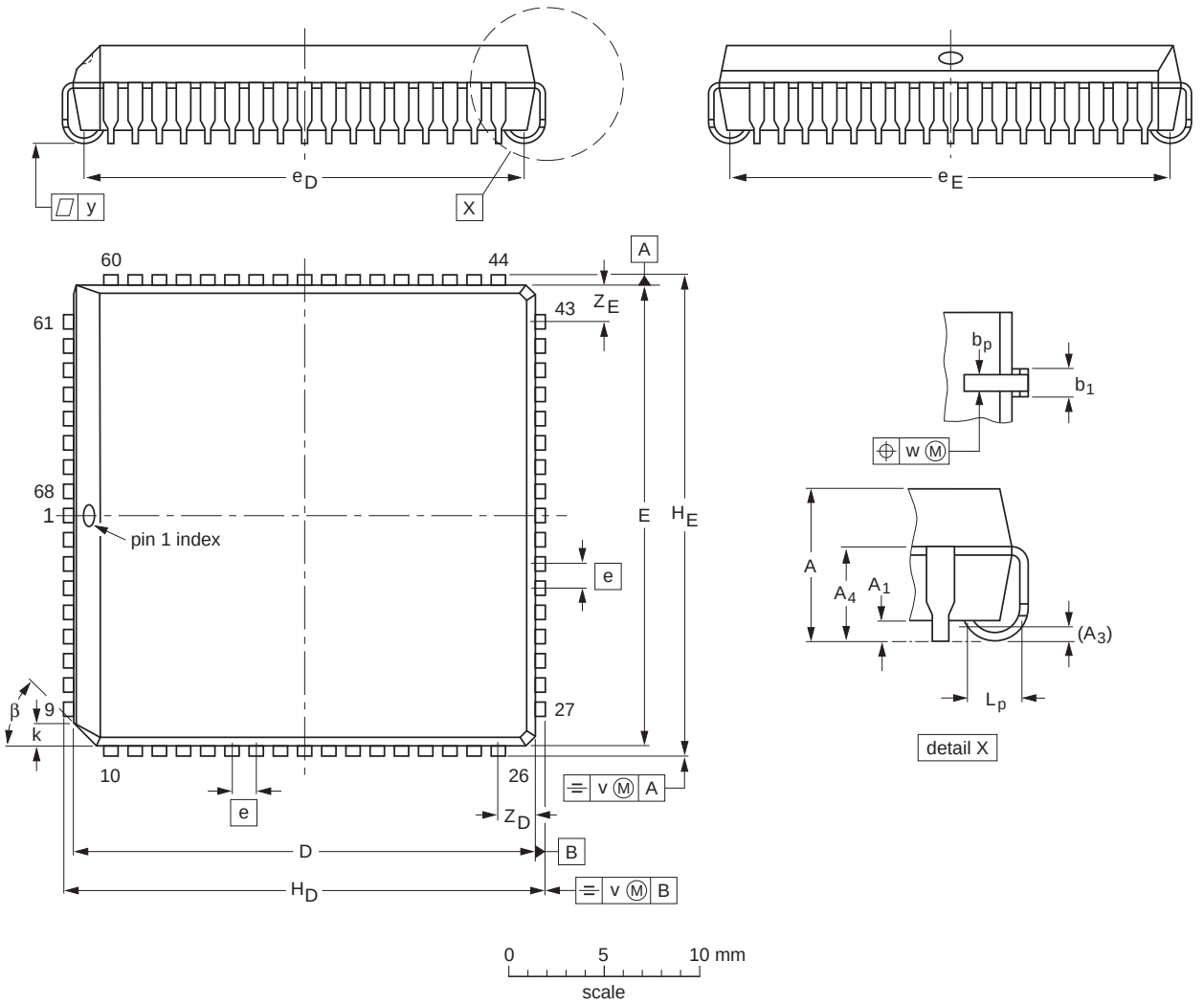
1. 32k byte user ROM data
2. ROM security bits.

ADDRESS	CONTENT	BIT(S)	COMMENT
0000H to 7FFFH	DATA	7:0	User ROM Data
8020H	SEC	0	ROM Security Bit 1
8020H	SEC	1	ROM Security Bit 2 0 = enable security 1 = disable security
8020H	SEC	3	ROM Security Bit 3 0 = enable security 1 = disable security

™Trademark phrase of Intel Corporation.

PLCC68: plastic leaded chip carrier; 68 leads

SOT188-2



DIMENSIONS (mm dimensions are derived from the original inch dimensions)

UNIT	A	A_1 min.	A_3	A_4 max.	b_p	b_1	$D^{(1)}$	$E^{(1)}$	e	e_D	e_E	H_D	H_E	k	L_p	v	w	y	$Z_D^{(1)}$ max.	$Z_E^{(1)}$ max.	β
mm	4.57 4.19	0.51	0.25	3.3	0.53 0.33	0.81 0.66	24.33 24.13	24.33 24.13	1.27	23.62 22.61	23.62 22.61	25.27 25.02	25.27 25.02	1.22 1.07	1.44 1.02	0.18	0.18	0.1	2.16	2.16	45°
inches	0.180 0.165	0.02	0.01	0.13	0.021 0.013	0.032 0.026	0.958 0.950	0.958 0.950	0.05	0.93 0.89	0.93 0.89	0.995 0.985	0.995 0.985	0.048 0.042	0.057 0.040	0.007	0.007	0.004	0.085	0.085	

Note

1. Plastic or metal protrusions of 0.25 mm (0.01 inch) maximum per side are not included.

OUTLINE VERSION	REFERENCES				EUROPEAN PROJECTION	ISSUE DATE
	IEC	JEDEC	JEITA			
SOT188-2	112E10	MS-018	EDR-7319			~~99-12-27~~ 01-11-14

NOTES

NXP Semiconductors

Customer notification

This data sheet was changed to reflect the new company name NXP Semiconductors.
Changes to content include: Corrected SOT188-3 to SOT188-2; changed data sheet specification to Product; updated legal definitions and disclaimers.

Contact information

For additional information please visit: **http://www.nxp.com**
For sales offices addresses send e-mail to: **salesaddresses@nxp.com**

Printed in The Netherlands R05/10/pp54 Date of release: 2013 Sep 04 Document order number: 9397 750 07816