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	32KB (32K x 8)
Program Memory Type	OTP
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	68-LCC (J-Lead)
Supplier Device Package	68-PLCC (24.18x24.18)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/pxas37kba-512

PIN DESCRIPTIONS

MNEMONIC	PIN NUMBER		TYPE	NAME AND FUNCTION
	PLCC	LQFP		
V_{SS}	1, 20, 55	12, 13, 53, 54, 69, 70	I	**Ground:** 0 V reference.
V_{DD}	2, 21, 54	14, 15, 51, 52, 71, 72	I	**Power Supply:** This is the power supply voltage for normal, idle, and power down operation.
$\overline{\text{RST}}$	50	47	I	**Reset:** A low on this pin resets the microcontroller, causing I/O ports and peripherals to take on their default states, and the processor to begin execution at the address contained in the reset vector.
$\overline{\text{RSTOUT}}$	19	11	O	**Reset Output:** This pin outputs a low whenever the XA-S3 processor is reset for any reason. This includes an external reset via the $\overline{\text{RST}}$ pin, watchdog reset, and the $\overline{\text{RESET}}$ instruction.
ALE/$\overline{\text{PROG}}$	47	44	I/O	**Address Latch Enable/Program Pulse:** A high output on the ALE pin signals external circuitry to latch the address portion of the multiplexed address/data bus. A pulse on ALE occurs only when it is needed in order to process a bus cycle.
$\overline{\text{PSEN}}$	48	45	O	**Program Store Enable:** The read strobe for external program memory. When the microcontroller accesses external program memory, $\overline{\text{PSEN}}$ is driven low in order to enable memory devices. $\overline{\text{PSEN}}$ is only active when external code accesses are performed.
$\overline{\text{EA}}$/WAIT/V_{PP}	22	16	I	**External Access/Bus Wait:** The $\overline{\text{EA}}$ input determines whether the internal program memory of the microcontroller is used for code execution. The value on the $\overline{\text{EA}}$ pin is latched as the external reset input is released and applies during later execution. When latched as a 0, external program memory is used exclusively. When latched as a 1, internal program memory will be used up to its limit, and external program memory used above that point. After reset is released, this pin takes on the function of bus WAIT input. If WAIT is asserted high during an external bus access, that cycle will be extended until WAIT is released.
XTAL1	68	68	I	**Crystal 1:** Input to the inverting amplifier used in the oscillator circuit and input to the internal clock generator circuits.
XTAL2	67	67	I	**Crystal 2:** Output from the oscillator amplifier.
CLKOUT	49	46	O	**Clock Output:** This pin outputs a buffered version of the internal CPU clock. The clock output may be used in conjunction with the external bus to synchronize WAIT state generators, etc. The clock output may be disabled by software.
AV_{DD}	33	28, 29	I	**Analog Power Supply:** Positive power supply input for the A/D converter.
AV_{SS}	34	30, 31	I	**Analog Ground.**
AV_{REF+}	32	27	I	**A/D Positive Reference Voltage:** High end reference for the A/D converter.
AV_{REF-}	31	26	I	**A/D Negative Reference Voltage:** Low end reference for the A/D converter.
P0.0 – P0.7	45, 46, 51–53, 56–58	42, 43, 48–50, 55–57	I/O	**Port 0:** Port 0 is an 8-bit I/O port with a user-configurable output type. Port 0 latches have 1s written to them and are configured in the quasi-bidirectional mode during reset. The operation of port 0 pins as inputs and outputs depends upon the port configuration selected. Each port pin is configured independently. Refer to the section on I/O port configuration and the DC Electrical Characteristics for details. When the external program/data bus is used, Port 0 becomes the multiplexed low data/instruction byte and address lines 4 through 11.

		SFR	BIT FUNCTIONS AND ADDRESSES								Reset
NAME	DESCRIPTION	Address	MSB							LSB	Value
CCAPM2#	PCA module 2 mode	493	–	ECOM2	CAPP2	CAPN2	MAT2	TOG2	PWM2	ECCF2	00h
CCAPM3#	PCA module 3 mode	494	–	ECOM3	CAPP3	CAPN3	MAT3	TOG3	PWM3	ECCF3	00h
CCAPM4#	PCA module 4 mode	495	–	ECOM4	CAPP4	CAPN4	MAT4	TOG4	PWM4	ECCF4	00h
CCAP0H#	PCA module 0 capture high byte	497									xx
CCAP1H#	PCA module 1 capture high byte	499									xx
CCAP2H#	PCA module 2 capture high byte	49B									xx
CCAP3H#	PCA module 3 capture high byte	49D									xx
CCAP4H#	PCA module 4 capture high byte	49F									xx
CCAP0L#	PCA module 0 capture low byte	496									xx
CCAP1L#	PCA module 1 capture low byte	498									xx
CCAP2L#	PCA module 2 capture low byte	49A									xx
CCAP3L#	PCA module 3 capture low byte	49C									xx
CCAP4L#	PCA module 4 capture low byte	49E									xx
CS	Code segment	443									00h
DS	Data segment	441									00h
ES	Extra segment	442									00h
			367	366	365	364	363	362	361	360	
I2CON#*	I^{2}C control register	42C	CR2	ENA	STA	STO	SI	AA	CR1	CR0	00h
I2STAT#	I^{2}C status register	46C	I^{2}C Status Code/Vector					0	0	0	F8h
I2DAT#	I^{2}C data register	46D									xx
I2ADDR#	I^{2}C address register	46E	I^{2}C Slave Address							GC	00h
			33F	33E	33D	33C	33B	33A	339	338	
IEH*	Interrupt enable high byte	427	–	–	–	–	ETI1	ERI1	ETI0	ERI0	00h
			337	336	335	334	333	332	331	330	
IEL#*	Interrupt enable low byte	426	EA	EAD	EPC	ET2	ET1	EX1	ET0	EX0	00h
			377	376	375	374	373	372	371	370	
IELB#*	Interrupt enable B low byte	42E	–	–	EI2	EC4	EC3	EC2	EC1	EC0	00h
IPA0	Interrupt priority A0	4A0	–	PT0			–	PX0			00h
IPA1	Interrupt priority A1	4A1	–	PT1			–	PX1			00h
IPA2#	Interrupt priority A2	4A2	–	PPC			–	PT2			00h
IPA3#	Interrupt priority A3	4A3	–	–			–	PAD			00h
IPA4	Interrupt priority A4	4A4	–	PTI0			–	PRI0			00h
IPA5	Interrupt priority A5	4A5	–	PTI1			–	PRI1			00h
IPB0#	Interrupt priority B0	4A8	–	PC1			–	PC0			00h
IPB1#	Interrupt priority B1	4A9	–	PC3			–	PC2			00h
IPB2#	Interrupt priority B2	4AA	–	PI2			–	PC4			00h
			387	386	385	384	383	382	381	380	
P0*	Port 0	430	A11D7	A10D6	A9D5	A8D4	A7D3	A6D2	A5D1	A4D0	FFh
			38F	38E	38D	38C	38B	38A	389	388	
P1*	Port 1	431	T2EX	T2	TxD1	RxD1	A3	A2	A1	A0/WRH	FFh
			397	396	395	394	393	392	391	390	
P2*	Port 2	432	A19D15	A18D14	A17D13	A16D12	A15D11	A14D10	A13D9	A12D8	FFh
			39F	39E	39D	39C	39B	39A	399	398	
P3*	Port 3	433	RD	WRL	T1	T0	INT1	INT0	TxD0	RxD0	FFh
			3A7	3A6	3A5	3A4	3A3	3A2	3A1	3A0	
P4#*	Port 4	434	A21	A20	CEX4	CEX3	CEX2	CEX1	CEX0	ECI	FFh

		SFR	BIT FUNCTIONS AND ADDRESSES								Reset
NAME	DESCRIPTION	Address	MSB							LSB	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 for other 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, based on 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. This prevents 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.

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

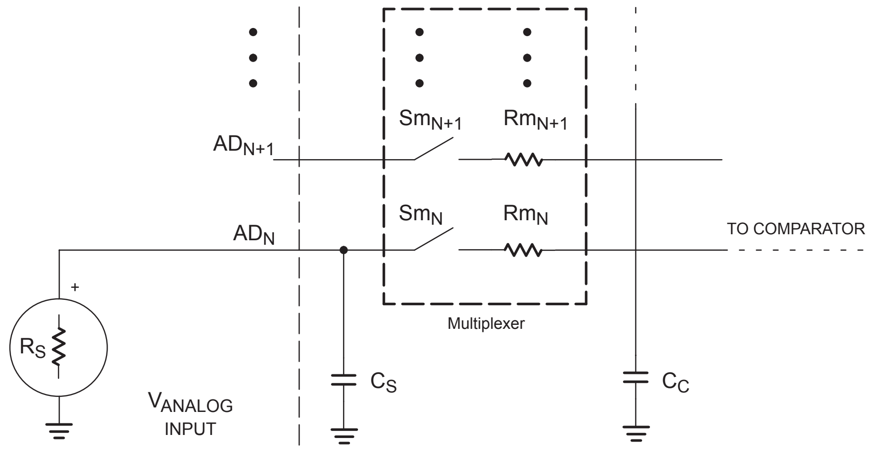
NOTE:
1. These settings provide additional A/D input sampling time, in order to allow accurate readings with a higher external sourceimpedance.

Table 3. A/D Timing Configuration for 10-bit Mode

ADCFG.3–0	Max. Oscillator Frequency (MHz)	Conversion Time		Sampling Time (Osc. Clocks)
		Osc. Clocks	μsec at max. Osc.	
0h (0000)	6.66	88	13.21	4
1h (0001)	8	92	9.2	6
2h (0010)	8	96	8.64	8
3h (0011)	12	116	8.7	8
4h (0100)	12	120	7.2	10
5h (0101)	12	124	6.2	12
6h (0110)	12	136	6.8	24
7h (0111)	12	128	5.77	14
8h (1000)	13	148	6.35	14
9h (1001)	13	152	5.71	16
Ah (1010)	13	156	5.2	18
Bh (1011)	13	170	5.67	32
Ch (1100)	13	160	5.0	20
Dh (1101)	16	180	5.41	20
Eh (1110)	20	204	5.57	22
Fh (1111)	20	208	5.2	24

A/D Inputs

In order to obtain accurate measurements with the A/D Converter, the source drive must be sufficient to adequately charge the sampling capacitor during the sampling time. Figure 4 shows the equivalent resistance and capacitance related to the A/D inputs. A/D timing configurations indicated in Table 1 allow for full A/D accuracy (according to the A/D specifications) assuming a source resistance of less than or equal to 20kΩ. Larger source resistances may be accommodated by increasing the sampling time with a different A/D timing configuration.



R_m (multiplexer resistance) = 3 kΩ maximum
C_S (pin capacitance) = 10 pF maximum
C_C (sampling capacitor) = 2 pF maximum
R_S (source resistance) = Recommended less than 20kΩ for full specified accuracy. This allows time for the sampling capacitor (C_C) to fully charge while the multiplexer switch is closed. Please note that sampling causes the analog input to present a varying load to the pin.

SU00948

Figure 4. A/D Input: Equivalent Circuit

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.

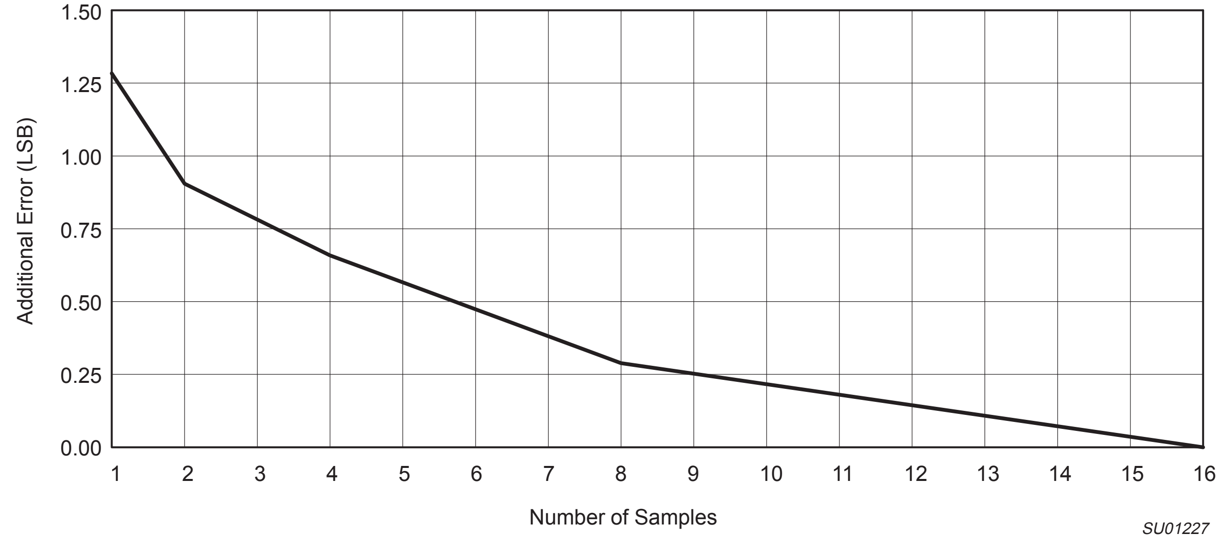


Figure 5. A/D accuracy by number of averaging samples
(Pertains to 10-bit mode only. Note that 10-bit mode is only specified up to f_C = 20 MHz.)

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

MSB LSB

CR2	ENA	STA	STO	SI	AA	CR1	CR0

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

SU00941

Figure 6. I^2C Control Register (I2CON)

I^{2}C Interface

The I^{2}C interface on the XA-S3 is identical to the standard byte-style I^{2}C interface found on devices such as the 8xC552 except for the rate selection. ***The I^{2}C interface conforms to the 100 kHz I^{2}C specification, but may be used at rates up to 400 kHz (non-conforming).***

Important: Before the I^{2}C interface may be used, the port pins P5.6 and 5.7, which correspond to the I^{2}C functions SCL and SDA respectively, must be set to the open drain mode.

The processor interfaces to the I^{2}C logic via the following four special function registers: I2CON (I^{2}C control register), I2STA (I^{2}C status register), I2DAT (I^{2}C data register), and I2ADR (I^{2}C slave address register). The I^{2}C control logic interfaces to the external I^{2}C 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 I^{2}C 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 I^{2}C bus. The STO bit is also cleared when ENA = “0”.

ENA, the I^{2}C 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 I^{2}C-bus since, when ENA is reset, the I^{2}C-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 I^{2}C hardware checks the status of the I^{2}C bus and generates a START condition if the bus is free. If the bus is not free, the I^{2}C 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 I^{2}C 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 I^{2}C 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 I^{2}C interface is in a master mode, a STOP condition is transmitted to the I^{2}C 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 I^{2}C 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 I^{2}C 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 I^{2}C 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 (I^{2}C interrupt enable) bits are also set, an I^{2}C interrupt is requested. SI is set by hardware when one of 25 of the 26 possible I^{2}C 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 I^{2}C interface is in the master receiver mode.
- A data byte has been received while the I^{2}C 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 I^{2}C interface is in the master receiver mode.
- A data byte has been received while the I^{2}C interface is in the addressed slave receiver mode.

When the I^{2}C 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 I^{2}C 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 I^{2}C 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 I^{2}C 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.

XA-S3 Timer/Counters

The XA-S3 has three general purpose counter/timers, two of which may also be used as baud rate generators for either or both of the UARTs.

Timer 0 and 1
These are identical to the standard XA-G3 timer 0 and 1.

Timer 2
This is identical to the standard XA-G3 timer 2.

Programmable Counter Array (PCA)

The Programmable Counter Array available on the XA-S3 is a special 16-bit Timer that has five 16-bit capture/compare modules associated with it. Each of the modules can be programmed to operate in one of four modes: rising and/or falling edge capture, software timer, high-speed output, or pulse width modulator. Each module has a pin associated with it in port 1. Module 0 is connected to P4.1(CEX0), module 1 to P4.2(CEX1), etc. The basic PCA configuration is shown in Figure 7.

The PCA timer is a common time base for all five modules and can be programmed to run at: the TCLK rate (Osc/4, Osc/16, or Osc/64), the Timer 0 overflow, or the input on the ECI pin (P4.0). When the ECI input is used, the falling edge clocks the PCA counter. The timer count source is determined from the CPS1 and CPS0 bits in the CMOD SFR as follows (see Figure 10):

CPS1	CPS0	PCA Timer Count Source
0	X	TCLK (Osc/4, Osc/16, or Osc/64)
1	0	Timer 0 overflow
1	1	ECI (PCA External Clock Input (max rate = Osc/4)

In the CMOD SFR are three additional bits associated with the PCA. They are CIDL which allows the PCA to stop during idle mode, WDTE which enables or disables the watchdog function on module 4, and ECF which when set causes an interrupt and the PCA overflow flag CF (in the CCON SFR) to be set when the PCA timer overflows. These functions are shown in Figure 8. In addition, each PCA module may generate a separate interrupt.

The watchdog timer function is implemented in module 4 (see Figure 17).

The CCON SFR contains the run control bit for the PCA and the flags for the PCA timer (CF) and each module (refer to Figure 11). To run the PCA the CR bit (CCON.6) must be set by software. The PCA is shut off by clearing this bit. The CF bit (CCON.7) is set when the PCA counter overflows and an interrupt will be generated if the ECF bit in the CMOD register is set, The CF bit can only be cleared by software. Bits 0 through 4 of the CCON register are the flags for the modules (bit 0 for module 0, bit 1 for module 1, etc.) and are set by hardware when either a match or a capture occurs. These flags also can only be cleared by software. The PCA interrupt system shown in Figure 9.

Each module in the PCA has a special function register associated with it. These registers are: CCAPM0 for module 0, CCAPM1 for module 1, etc. (see Figure 12). The registers contain the bits that control the mode that each module will operate in. The ECCF bit (CCAPMn.0 where n=0, 1, 2, 3, or 4 depending on the module) enables the CCF flag in the CCON SFR to generate an interrupt when a match or compare occurs in the associated module. PWM (CCAPMn.1) enables the pulse width modulation mode. The TOG bit (CCAPMn.2) when set causes the CEX output associated with the module to toggle when there is a match between the PCA counter and the module's capture/compare register. The match bit MAT (CCAPMn.3) when set will cause the CCFn bit in the CCON register to be set when there is a match between the PCA counter and the module's capture/compare register.

The next two bits CAPN (CCAPMn.4) and CAPP (CCAPMn.5) determine the edge that a capture input will be active on. The CAPN bit enables the negative edge, and the CAPP bit enables the positive edge. If both bits are set both edges will be enabled and a capture will occur for either transition. The last bit in the register ECOM (CCAPMn.6) when set enables the comparator function. Figure 13 shows the CCAPMn settings for the various PCA functions.

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.

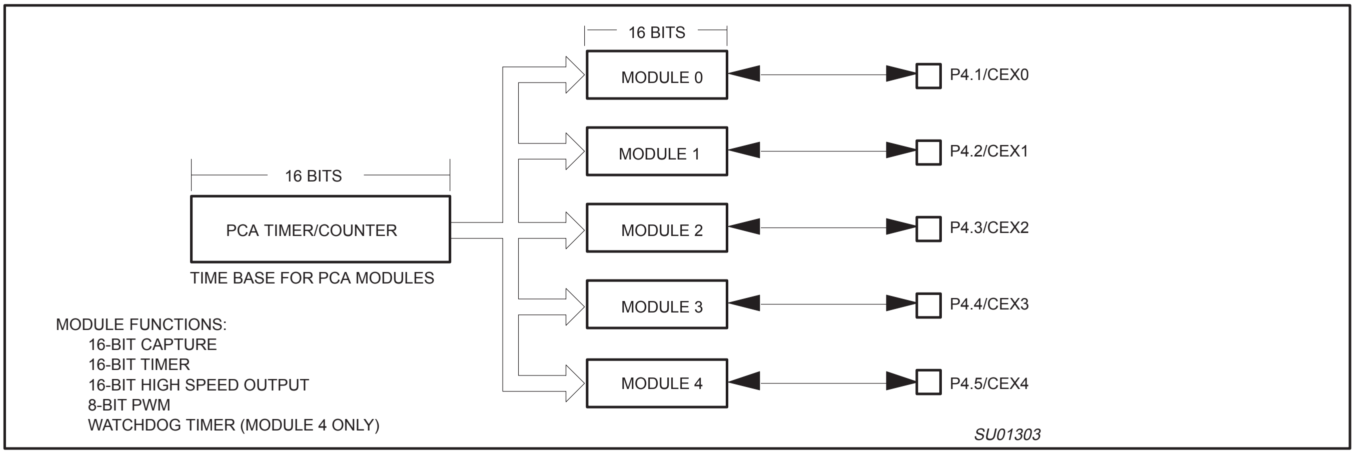


Figure 7. Programmable Counter Array (PCA)

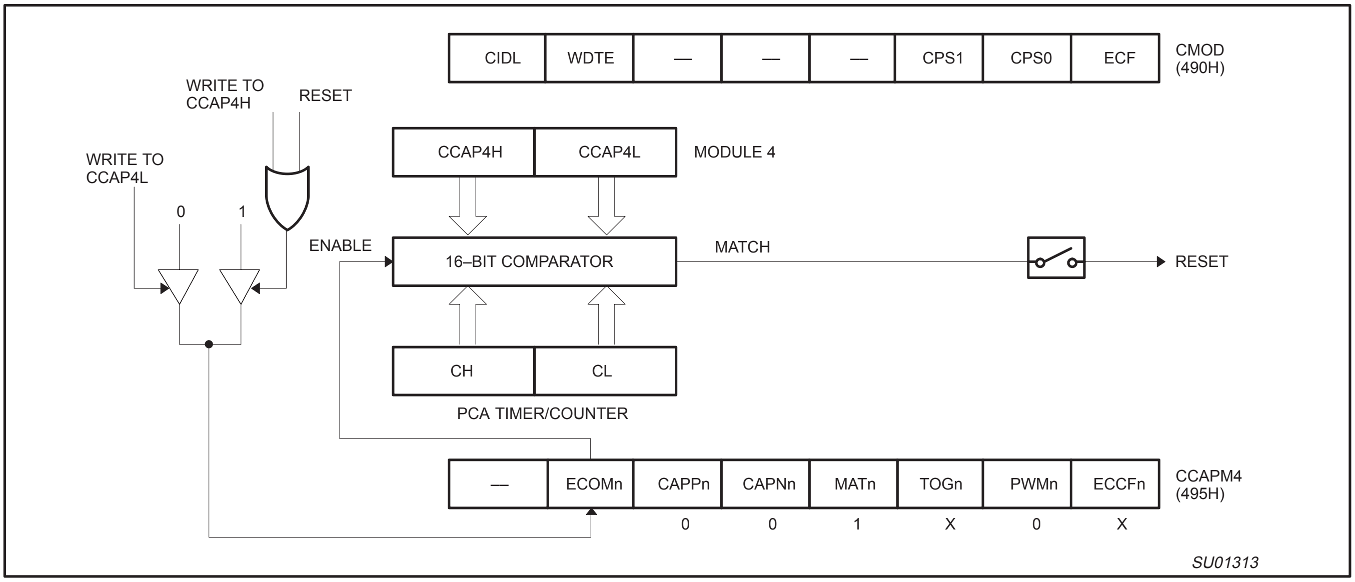


Figure 18. PCA Watchdog Timer m(Module 4 only)

PCA Watchdog Timer

An on-board watchdog timer is available with the PCA to improve the reliability of the system without increasing chip count. Watchdog timers are useful for systems that are susceptible to noise, power glitches, or electrostatic discharge. Module 4 is the only PCA module that can be programmed as a watchdog. However, this module can still be used for other modes if the watchdog is not needed.

Figure 18 shows a diagram of how the watchdog works. The user pre-loads a 16-bit value in the compare registers. Just like the other compare modes, this 16-bit value is compared to the PCA timer value. If a match is allowed to occur, an internal reset will be generated. This will not cause the RST pin to be driven low.

In order to hold off the reset, the user has three options:

1. periodically change the compare value so it will never match the PCA timer,
2. periodically change the PCA timer value so it will never match the compare values, or
3. disable the watchdog by clearing the WDTE bit before a match occurs and then re-enable it.

The first two options are more reliable because the watchdog timer is never disabled as in option #3. If the program counter ever goes astray, a match will eventually occur and cause an internal reset. The second option is also not recommended if other PCA modules are being used. Remember, the PCA timer is the time base for **all** modules; changing the time base for other modules would not be a good idea. Thus, in most applications the first solution is the best option.

Figure 19 shows the code for initializing the watchdog timer. Module 4 can be configured in either compare mode, and the WDTE bit in CMOD must also be set. The user's software then must periodically change (CCAP4H,CCAP4L) to keep a match from occurring with the PCA timer (CH,CL). This code is given in the WATCHDOG routine in Figure 19.

This routine should not be part of an interrupt service routine, because if the program counter goes astray and gets stuck in an infinite loop, interrupts will still be serviced and the watchdog will keep getting reset. Thus, the purpose of the watchdog would be defeated. Instead, call this subroutine from the main program within 2^{16} count of the PCA timer.

```
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

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^{2}C 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)

ABSOLUTE MAXIMUM RATINGS

PARAMETER	RATING	UNIT
Operating temperature under bias	–55 to +125	°C
Storage temperature range	–65 to +150	°C
Voltage on $\overline{EA}/V_{PP}$ pin to V_{SS}	0 to +13.0	V
Voltage on any other pin to V_{SS}	–0.5 to V_{DD}+0.5 V	V
Maximum I_{OL} per I/O pin	15	mA
Power dissipation (based on package heat transfer, not device power consumption)	1.5	W

DC ELECTRICAL CHARACTERISTICS

V_{DD} = 2.7 V to 5.5 V, unless otherwise specified.
T_{amb} = 0 to +70°C for commercial, T_{amb} = –40°C to +85°C for industrial, unless otherwise specified.

SYMBOL	PARAMETER	TEST CONDITIONS	LIMITS			UNIT
			MIN	TYP	MAX	
I_{DD}	Power supply current, operating	5.0 V, 30 MHz			80	mA
I_{ID}	Power supply current, Idle mode	5.0 V, 30 MHz			35	mA
I_{PD}	Power supply current, Power Down mode	5.0 V, 3.0 V		5	100	µA
		5.0 V, 3.0 V, –40 to +85°C			150	µA
V_{RAM}	RAM keep-alive voltage		1.5			V
V_{IL}	Input low voltage		–0.5		0.22 V_{DD}	V
V_{IH}	Input high voltage, except XTAL1, $\overline{RST}$	V_{DD} = 5.0 V	2.2			V
		V_{DD} = 3.0 V	2.0			V
V_{IH1}	Input high voltage to XTAL1, $\overline{RST}$	For both 3.0 V and 5.0 V	0.7 V_{DD}			V
V_{OL}	Output low voltage, all ports, ALE, $\overline{PSEN}$[4], CLKOUT	I_{OL} = 3.2 mA, V_{DD} = 5.0 V			0.5	V
		I_{OL} = 1.0 mA, V_{DD} = 3.0 V			0.4	V
V_{OH1}	Output high voltage, all ports, ALE, $\overline{PSEN}$[2], CLKOUT	I_{OH} = –100 µA, V_{DD} = 4.5 V	2.4			V
		I_{OH} = –30 µA, V_{DD} = 2.7 V	2.0			V
V_{OH2}	Output high voltage, all ports ALE, $\overline{PSEN}$[3], CLKOUT	I_{OH} = –3.2 mA, V_{DD} = 4.5 V	2.4			V
		I_{OH} = –1.0 mA, V_{DD} = 2.7 V	2.2			V
C_{IO}	Input/Output pin capacitance[1]				15	pF
I_{IL}	Logical 0 input current, all ports[7]	V_{IN} = 0.45 V			–50	µA
I_{LI}	Input leakage current, all ports[6]	V_{IN} = V_{IL} or V_{IH}			±10	µA
I_{TL}	Logical 1 to 0 transition current, all ports[5]	At V_{DD} = 5.5 V			–650	µA
		At V_{DD} = 2.7 V			–250	µA

NOTES:

1. Maximum 15pF for $\overline{EA}/V_{PP}$.
2. Ports in quasi-bidirectional mode with weak pullup (applies to ALE, $\overline{PSEN}$ only during $\overline{RESET}$).
3. Ports in PUSH-PULL mode, both pullup and pulldown assumed to be the same strength.
4. In all output modes.
5. Port pins source a transition current when used in quasi-bidirectional mode and externally driven from 1 to 0. This current is highest when V_{IN} is approximately 2 V.
6. Measured with port in high impedance mode.
7. Measured with port in quasi-bidirectional mode.
8. Under steady state (non-transient) conditions, I_{OL} must be externally limited as follows:

Maximum I_{OL} per port pin:	15 mA
Maximum I_{OL} per 8-bit port:	26 mA
Maximum total I_{OL} for all outputs:	71 mA

If I_{OL} exceeds the test condition, V_{OL} may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.

10-BIT[10] MODE A/D CONVERTER DC ELECTRICAL CHARACTERISTICS

T_{amb} = 0 to +70°C for commercial, T_{amb} = –40 to +85°C for industrial, unless otherwise specified.

SYMBOL	PARAMETER	TEST CONDITIONS	LIMITS		UNIT
			MIN	MAX	
AV_{DD}	Analog supply voltage		2.7	3.3	V
AI_{DD}	Analog supply current (operating)	Port 5 = 0 to AV_{DD}		2.5	mA
AI_{ID}	Analog supply current (Idle mode)			2.5	µA
AI_{PD}	Analog supply current (Power-Down mode)	Commercial temperature range		100	µA
		Industrial temperature range		150	µA
AV_{IN}	Analog input voltage		AV_{SS} –0.2	AV_{DD} +0.2	V
R_{REF}	Resistance between V_{REF+} and V_{REF-}		125	225	kΩ
C_{IA}	Analog input capacitance			15	pF
DL_e	Differential non-linearity[1, 2, 3]			±1 [9]	LSB
IL_e	Integral non-linearity[1, 4]			±2.5 [9]	LSB
OS_e	Offset error[1, 5]			±6 [9]	LSB
G_e	Gain error[1, 6]			±1 [9]	%
A_e	Absolute voltage error (with averaging)[1, 7]			±8 [9]	LSB
M_{CTC}	Channel-to-channel matching			±1	LSB
C_t	Crosstalk between inputs of port[8]	0 – 100 kHz		–60	dB

NOTES:

1. Conditions: AV_{REF-} = 0 V; AV_{REF+} = 3.07 V.
2. The differential non-linearity (DL_e) is the difference between the actual step width and the ideal step width. See Figure 25.
3. The ADC is monotonic, there are no missing codes.
4. The integral non-linearity (IL_e) is the peak difference between the center of the steps of the actual and the ideal transfer curve after appropriate adjustment of gain and offset errors. See Figure 25.
5. The offset error (OS_e) is the absolute difference between the straight line which fits the actual transfer curve (after removing gain error), and the straight line which fits the ideal transfer curve. See Figure 25.
6. The gain error (G_e) is the relative difference in percent between the straight line fitting the actual transfer curve (after removing offset error), and the straight line which fits the ideal transfer curve. Gain error is constant at every point on the transfer curve. See Figure 25.
7. The absolute voltage error (A_e) is the maximum difference between the center of the steps of the actual transfer curve of the non-calibrated ADC and the ideal transfer curve.
8. This should be considered when both analog and digital signals are input simultaneously to Port 5. Parameter is guaranteed bydesign.
9. 10-bit mode only.
10. 10-bit mode is only operational up to f_C = 20 MHz.

AC ELECTRICAL CHARACTERISTICS (5 V)

V_{DD} = 4.5 V to 5.5 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	
External Clock					
f_C	32	Oscillator frequency	0	30	MHz
t_C	32	Clock period and CPU timing cycle	1/f_C		ns
t_{CHCX}	32	Clock high-time (Note 7)	t_C * 0.5		ns
t_{CLCX}	32	Clock low time (Note 7)	t_C * 0.4		ns
t_{CLCH}	32	Clock rise time (Note 7)		5	ns
t_{CHCL}	32	Clock fall time (Note 7)		5	ns
Address Cycle					
t_{LHLL}	26, 28, 30	ALE pulse width (programmable)	(V1 * t_C) – 6		ns
t_{AVLL}	26, 28, 30	Address valid to ALE de-asserted (set-up)	(V1 * t_C) – 12		ns
t_{LLAX}	26, 28, 30	Address hold after ALE de-asserted	(t_C/2) – 10		ns
Code Read Cycle					
t_{PLPH}	26	$\overline{PSEN}$ pulse width	(V2 * t_C) – 10		ns
t_{LLPL}	26	ALE de-asserted to $\overline{PSEN}$ asserted	(t_C/2) – 7		ns
t_{AVIVA}	26	Address valid to instruction valid, ALE cycle (access time)		(V3 * t_C) – 36	ns
t_{AVIVB}	27	Address valid to instruction valid, non-ALE cycle (access time)		(V4 * t_C) – 29	ns
t_{PLIV}	26	$\overline{PSEN}$ asserted to instruction valid (enable time)		(V2 * t_C) – 29	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) – 10		ns
t_{LLRL}	28	ALE de-asserted to $\overline{RD}$ asserted	(t_C/2) – 7		ns
t_{AVDVA}	28	Address valid to data input valid, ALE cycle (access time)		(V6 * t_C) – 36	ns
t_{AVDVB}	29	Address valid to data input valid, non-ALE cycle (access time)		(V5 * t_C) – 29	ns
t_{RLDV}	28	$\overline{RD}$ low to valid data in (enable time)		(V7 * t_C) – 29	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) – 10		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) – 22		ns
t_{WHQX}	30	Data hold time after $\overline{WR}$ de-asserted (Note 6)	(V11 * t_C) – 5		ns
t_{AVWL}	30	Address valid to $\overline{WR}$ asserted (address set-up time) (Note 5)	(V9 * t_C) – 22		ns
t_{UAWH}	30	Hold time of unlatched part of address after $\overline{WR}$ is de-asserted	(V11 * t_C) – 7		ns
Wait Input					
t_{WTH}	31	WAIT stable after bus strobe ($\overline{RD}$, $\overline{WR}$, or $\overline{PSEN}$) asserted		(V10 * t_C) – 30	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.

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	$\overline{\text{PSEN}}$ pulse width	(V2 * t_C) – 12		ns
t_{LLPL}	26	ALE de-asserted to $\overline{\text{PSEN}}$ 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	$\overline{\text{PSEN}}$ asserted to instruction valid (enable time)		(V2 * t_C) – 52	ns
t_{PHIX}	26	Instruction hold after $\overline{\text{PSEN}}$ de-asserted	0		ns
t_{PHIZ}	26	Bus 3-State after $\overline{\text{PSEN}}$ 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	$\overline{\text{RD}}$ pulse width	(V7 * t_C) – 12		ns
t_{LLRL}	28	ALE de-asserted to $\overline{\text{RD}}$ 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	$\overline{\text{RD}}$ low to valid data in (enable time)		(V7 * t_C) – 52	ns
t_{RHDX}	28	Data hold time after $\overline{\text{RD}}$ de–asserted	0		ns
t_{RHDZ}	28	Bus 3-State after $\overline{\text{RD}}$ 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	$\overline{\text{WR}}$ pulse width	(V8 * t_C) – 12		ns
t_{LLWL}	30	ALE falling edge to $\overline{\text{WR}}$ asserted	(V12 * t_C) – 10		ns
t_{QVWX}	30	Data valid before $\overline{\text{WR}}$ asserted (data set-up time)	(V13 * t_C) – 28		ns
t_{WHQX}	30	Data hold time after $\overline{\text{WR}}$ de-asserted (Note 6)	(V11 * t_C) – 8		ns
t_{AVWL}	30	Address valid to $\overline{\text{WR}}$ asserted (address set-up time) (Note 5)	(V9 * t_C) – 28		ns
t_{UAWH}	30	Hold time of unlatched part of address after $\overline{\text{WR}}$ is de-asserted	(V11 * t_C) – 10		ns
Wait Input					
t_{WTH}	31	WAIT stable after bus strobe ($\overline{\text{RD}}$, $\overline{\text{WR}}$, or $\overline{\text{PSEN}}$) asserted		(V10 * t_C) – 40	ns
t_{WTL}	31	WAIT hold after bus strobe ($\overline{\text{RD}}$, $\overline{\text{WR}}$, or $\overline{\text{PSEN}}$) asserted	(V10 * t_C) – 5		ns

NOTES ON PAGE 41.

V7) This variable represents the programmed width of the $\overline{\text{RD}}$ pulse as determined by the DR1 and DR0 bits or the DRA1, DRA0 in the BTRH register, and the SLEW bit in the BTRL register. Note that during a 16-bit operation on an 8-bit external bus, $\overline{\text{RD}}$ remains low and does not exhibit a transition between the first and second byte bus cycles. V7 still applies for the purpose of determining peripheral timing requirements. The timing for the first byte is for a bus cycle with ALE, the timing for the second byte is for a bus cycle with no ALE.

- For a bus cycle with **no** ALE, V7 = 1 if DR1/0 = 00, 2 if DR1/0 = 01, 3 if DR1/0 = 10, and 4 if DR1/0 = 11.
- For a bus cycle **with** an ALE, V7 = the total bus cycle duration (2 if DRA1/0 = 00, 3 if DRA1/0 = 01, 4 if DRA1/0 = 10, and 5 if DRA1/0 = 11) minus the number of clocks used by ALE (V1 + 0.5).
  Example: if DRA1/0 = 00 and ALEW = 0, then V7 = 2 – (0.5 +0.5) = 1.

V8) This variable represents the programmed width of the $\overline{\text{WRL}}$ and/or $\overline{\text{WRH}}$ pulse as determined by the WM1 bit in the BTRL register. V8 = 1 if WM1 = 0, and 2 if WM1 = 1.

V9) This variable represents the programmed address setup time for a write as determined by the data write cycle duration (defined by DW1 and DW0 or the DWA1 and DWA0 bits in the BTRH register), the WM0 bit in the BTRL register, and the value of V8.

- For a bus cycle **with** an ALE, V9 = the total bus write cycle duration (2 if DWA1/0 = 00, 3 if DWA1/0 = 01, 4 if DWA1/0 = 10, and 5 if DWA1/0 = 11) minus the number of clocks used by the $\overline{\text{WRL}}$ and/or $\overline{\text{WRH}}$ pulse (V8) minus the number of clocks used by data hold time (0 if WM0 = 0 and 1 if WM0 = 1).
  Example: If DWA1/0 = 10, WM0 = 1, and WM1 = 1, then V9 = 4 – 1 – 2 = 1.
- For a bus cycle with **no** ALE, V9 = the total bus cycle duration (2 if DW1/0 = 00, 3 if DW1/0 = 01, 4 if DW1/0 = 10, and 5 if DW1/0 = 11) minus the number of clocks used by the $\overline{\text{WRL}}$ and/or $\overline{\text{WRH}}$ pulse (V8), minus the number of clocks used by data hold time (0 if WMo = 0 and 1 if WM0 = 1).
  Example: If DW1/0 = 11, WM0 = 1, and WM1 = 0, then V9 = 5 – 1 – 1 = 3.

V10) This variable represents the length of a bus strobe for calculation of WAIT set-up and hold times. The strobe may be $\overline{\text{RD}}$ (for data read cycles), $\overline{\text{WRL}}$ and/or $\overline{\text{WRH}}$ (for data write cycles), or $\overline{\text{PSEN}}$ (for code read cycles), depending on the type of bus cycle being widened by WAIT. V10 = 2 for WAIT associated with a code read cycle using $\overline{\text{PSEN}}$. V10 = V8 for a data write cycle using $\overline{\text{WRL}}$ and/or $\overline{\text{WRH}}$. V10 = V7 – 1 for a data read cycle using $\overline{\text{RD}}$. This means that a single clock data read cycle cannot be stretched using WAIT. If WAIT is used to vary the duration of data read cycles, the $\overline{\text{RD}}$ strobe width must be set to be at least two clocks in duration. Also see Note 4.

V11) This variable represents the programmed write hold time as determined by the WM0 bit in the BTRL register. V11 0 if the WM0 bit = 0, and 1 if the WM0 bit = 1.

V12) this variable represents the programmed period between the end of the ALE pulse and the beginning of the $\overline{\text{WRL}}$ and/or $\overline{\text{WRH}}$ pulse as determined by the data write cycle duration (defined by the DWA1 and DWA0 bits in the BTRH register), the WM0 bit in the BTRL register, and the values of V1 and V8. V12 = the total bus cycle duration (2 if DWA1/0 = 00, 3 if DWA1/0 = 01, 4 if DWA1/0 = 10, and 5 if DWA1/0 = 11) minus the number of clocks used by the $\overline{\text{WRL}}$ and/or $\overline{\text{WRH}}$ pulse (V8), minus the number of clocks used by data hold time (0 if WM0 = 0 and 1 if WM0 = 1), minus the width of the ALE pulse (V1).
Example: If SWA1/0 = 11, WM0 = 1, WM1 = 0, and ALEW = 1, then V12 = 5 – 1 – 1 – 1.5 = 1.5.

V13) This variable represents the programmed data setup time for a write as determined by the data write cycle duration (defined by DW1 and DW0 or the DWA1 and DWA0 bits in the BTRH register), the WM0 bit in the BTRL register, and the values of V1 and V8.

- For a bus cycle **with** an ALE, V13 = the total bus cycle duration (2 if DWA1/0 = 00, 3 if DWA1/0 = 01, 4 if DWA1/0 = 10, and 5 if DWA1/0 = 11) minus the number of clocks used by the $\overline{\text{WRL}}$ and/or $\overline{\text{WRH}}$ pulse (V8), minus the number of clocks used by data hold time (0 if WM0 = 0 and 1 if WM0 = 1), minus the number of clocks used by ALE (V1 + 0.5).
  Example: If DWA1/0 = 11, WM0 = 1, WM1 = 1, and ALEW = 0, then V13 = 5 – 1 – 2 – 1 = 1.
- For a bus cycle with **no** ALE, V13 = the total bus cycle duration (2 if DW1/0 = 00, 3 if DW1/0 = 01, 4 if DW1/0 = 10, and 5 if DW1/0 = 11) minus the number of clocks used by the $\overline{\text{WRL}}$ and/or $\overline{\text{WRH}}$ pulse (V8), minus the number of clocks used by data hold time (0 if WM0 = 0 and 1 if WM0 = 1).
  Example: If DW1/0 = 01, WM0 = 1, and WM1 = 0, then V13 = 3 – 1 – 1 = 1.

3. Not all combinations of bus timing configuration values result in valid bus cycles. Please refer to the *XA User Guide* section on the External Bus for details.
4. When code is being fetched for execution on the external bus, a burst mode fetch is used that dows not have $\overline{\text{PSEN}}$ edges in every fetch cycle. This would be A3–A0 for an 8-bit bus, and A3–A1 for a 16-bit bus. Also, a 16-bit read operation conducted on an 8-bit wide bus similarly does not include two separate $\overline{\text{RD}}$ strobes. So, a rising edge on the low order address line (A0) must be used to trigger a WAIT in the second half of such a cycle.
5. This parameter is provided for peripherals that have the data clocked in on the falling edge of the $\overline{\text{WR}}$ strobe. This is not usually the case and in most applications this parameter is not used.
6. Please note that the XA-S3 requires that extended data bus hold time (WM0 = 1) to be used with external bus write cycles.
7. Applies only to an external clock source, not when a crystal is connected to the XTAL1 and XTAL2 pins.
8. WAIT should not change between these times.

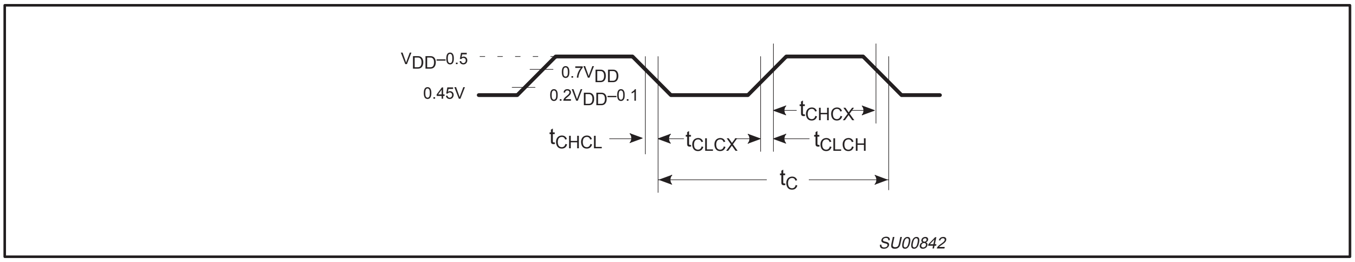
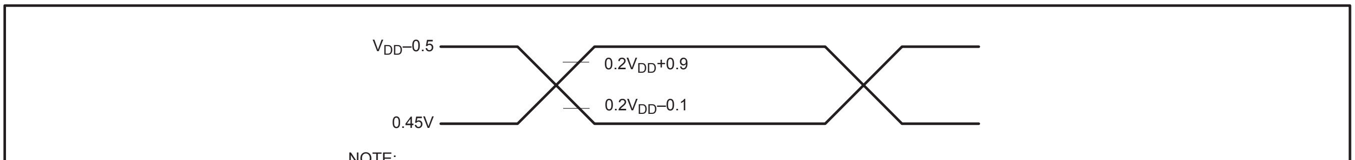
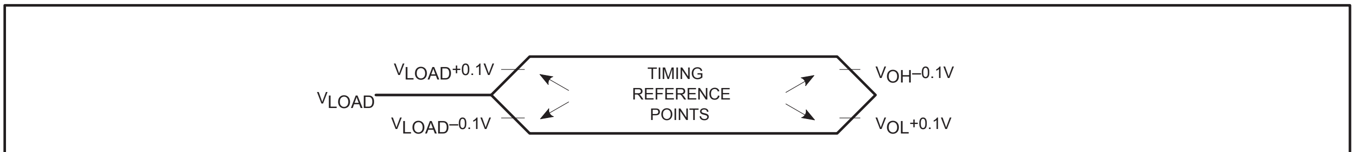


Figure 32. External Clock Drive



NOTE:
AC inputs during testing are driven at V_{DD} –0.5 for a logic ‘1’ and 0.45V for a logic ‘0’.
Timing measurements are made at the 50% point of transitions.

Figure 33. AC Testing Input/Output



NOTE:
For timing purposes, a port is no longer floating when a 100mV change from load voltage occurs, and begins to float when a 100mV change from the loaded V_{OH}/V_{OL} level occurs. $I_{OH}/I_{OL} \geq \pm 20mA$.

Figure 34. Float Waveform

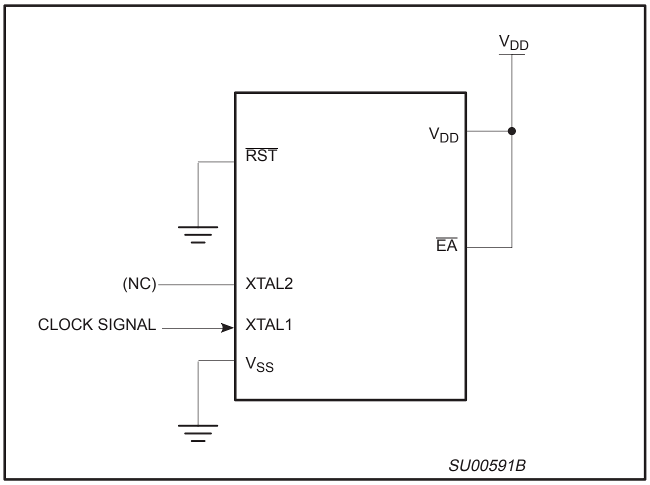


**Figure 35. I_{DD} Test Condition, Active Mode
All other pins are disconnected**

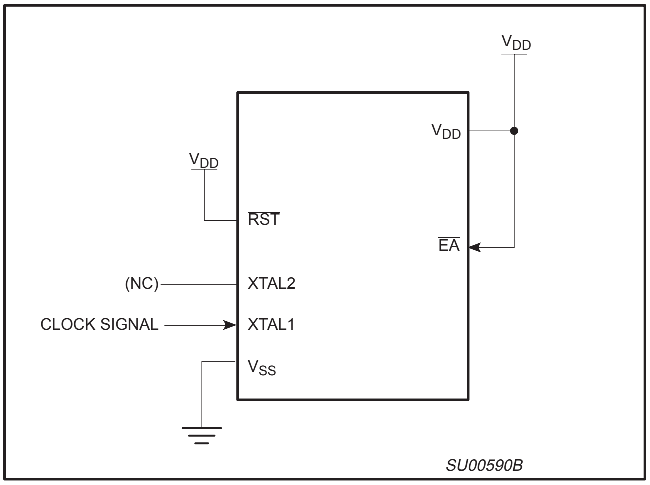


**Figure 36. I_{DD} Test Condition, Idle Mode
All other pins are disconnected**

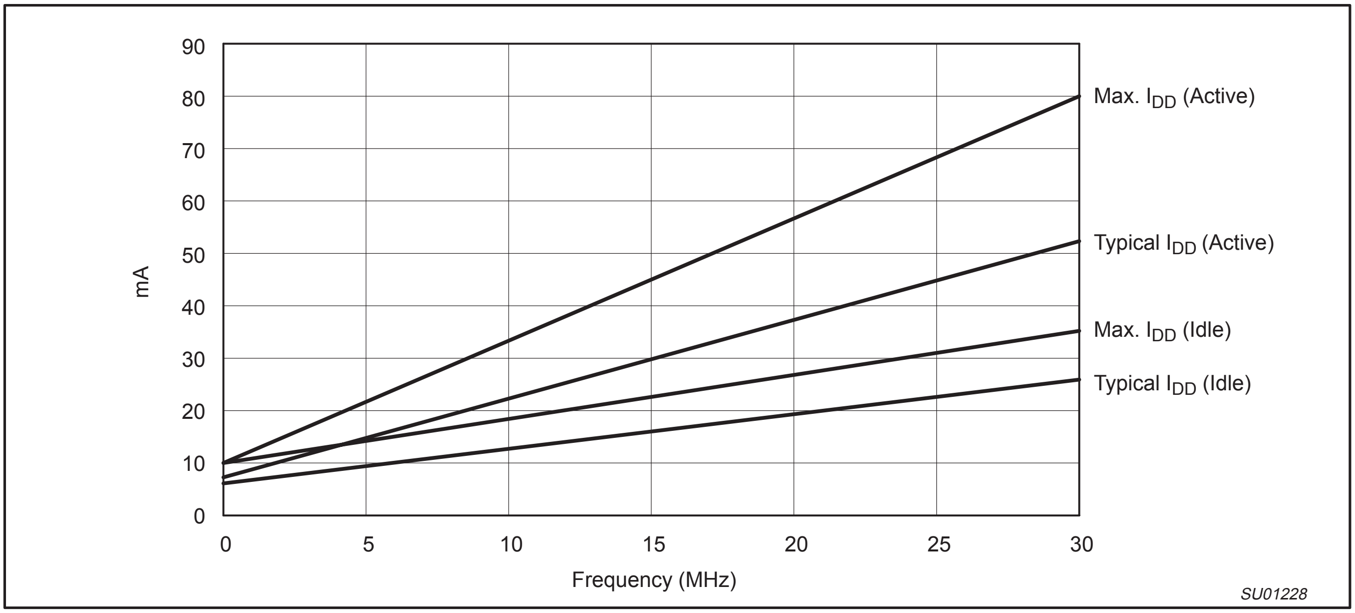


Figure 37. I_{DD} vs. Frequency
Valid only within frequency specification of the device under test.

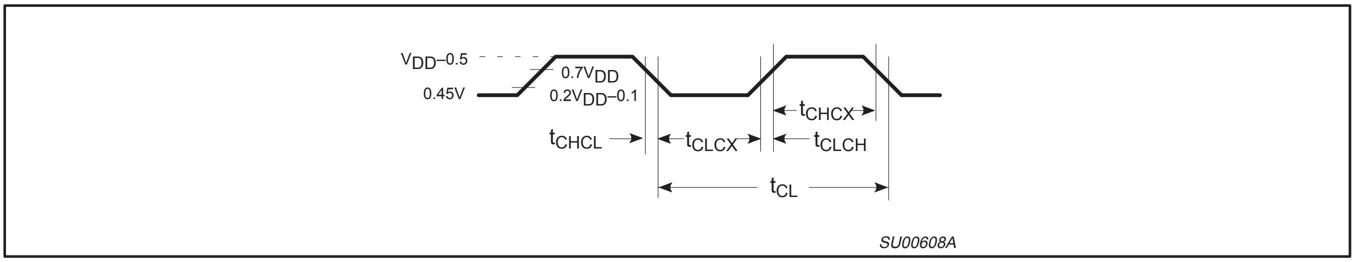


Figure 38. Clock Signal Waveform for I_{DD} Tests in Active and Idle Modes
$t_{CLCH} = t_{CHCL} = 5$ ns

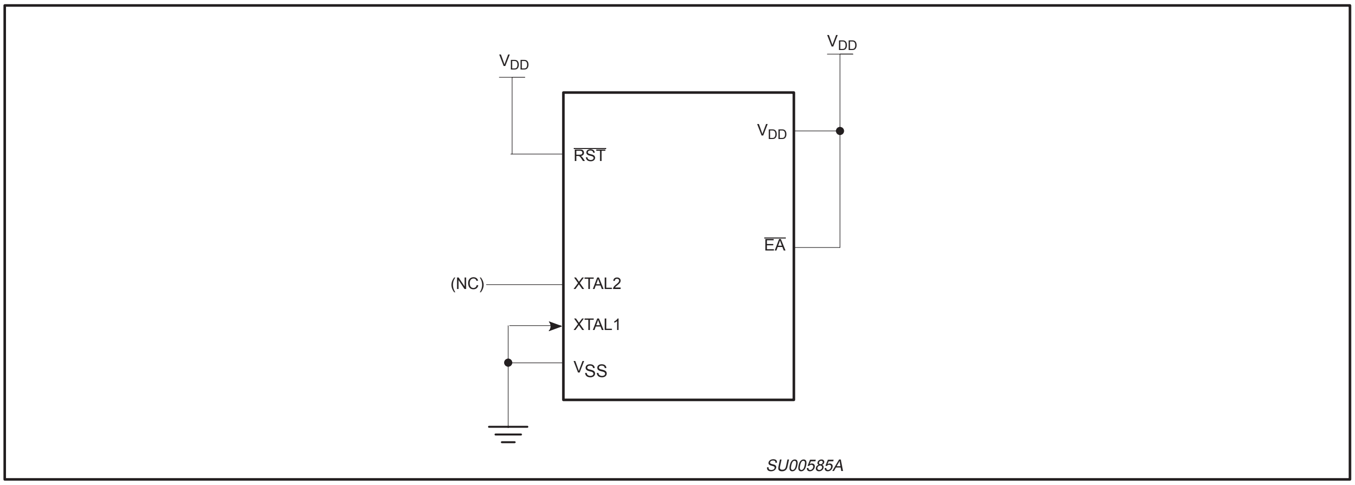


Figure 39. I_{DD} Test Condition, Power Down Mode
All other pins are disconnected. V_{DD}=2 V to 5.5 V

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**

The information presented in this document does not form part of any quotation or contract, is believed to be accurate and reliable and may be changed without notice. No liability will be accepted by the publisher for any consequence of its use. Publication thereof does not convey nor imply any license under patent- or other industrial or intellectual property rights.

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