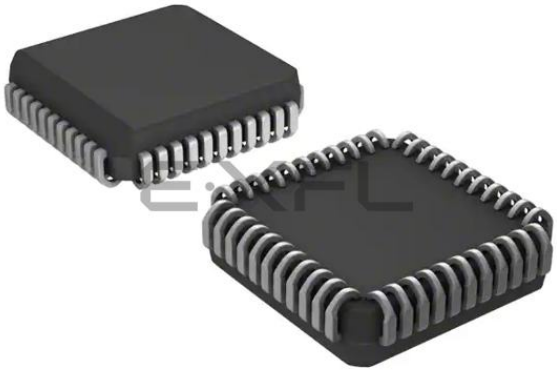


Welcome to E-XFL.COM

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	UART/USART
Peripherals	PWM, WDT
Number of I/O	32
Program Memory Size	-
Program Memory Type	ROMless
EEPROM Size	-
RAM Size	512 x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 5.5V
Data Converters	-
Oscillator Type	External
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Surface Mount
Package / Case	44-LCC (J-Lead)
Supplier Device Package	44-PLCC (16.59x16.59)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/pxag30kba-529

XA 16-bit microcontroller family
512 B RAM, watchdog, 2 UARTs

XA-G30

FAMILY DESCRIPTION

The Philips Semiconductors XA (eXtended Architecture) family of 16-bit single-chip microcontrollers is powerful enough to easily handle the requirements of high performance embedded applications, yet inexpensive enough to compete in the market for high-volume, low-cost applications.

The XA family provides an upward compatibility path for 80C51 users who need higher performance and 64k or more of program memory. Existing 80C51 code can also easily be translated to run on XA microcontrollers.

The performance of the XA architecture supports the comprehensive bit-oriented operations of the 80C51 while incorporating support for multi-tasking operating systems and high-level languages such as C. The speed of the XA architecture, at 10 to 100 times that of the 80C51, gives designers an easy path to truly high performance embedded control.

The XA architecture supports:

- Upward compatibility with the 80C51 architecture
- 16-bit fully static CPU with a 24-bit program and data address range
- Eight 16-bit CPU registers each capable of performing all arithmetic and logic operations as well as acting as memory pointers. Operations may also be performed directly to memory.
- Both 8-bit and 16-bit CPU registers, each capable of performing all arithmetic and logic operations.
- An enhanced instruction set that includes bit intensive logic operations and fast signed or unsigned 16 × 16 multiply and 32 / 16 divide
- Instruction set tailored for high level language support
- Multi-tasking and real-time executives that include up to 32 vectored interrupts, 16 software traps, segmented data memory, and banked registers to support context switching
- Low power operation, which is intrinsic to the XA architecture, includes power-down and idle modes.

More detailed information on the core is available in the XA User Guide.

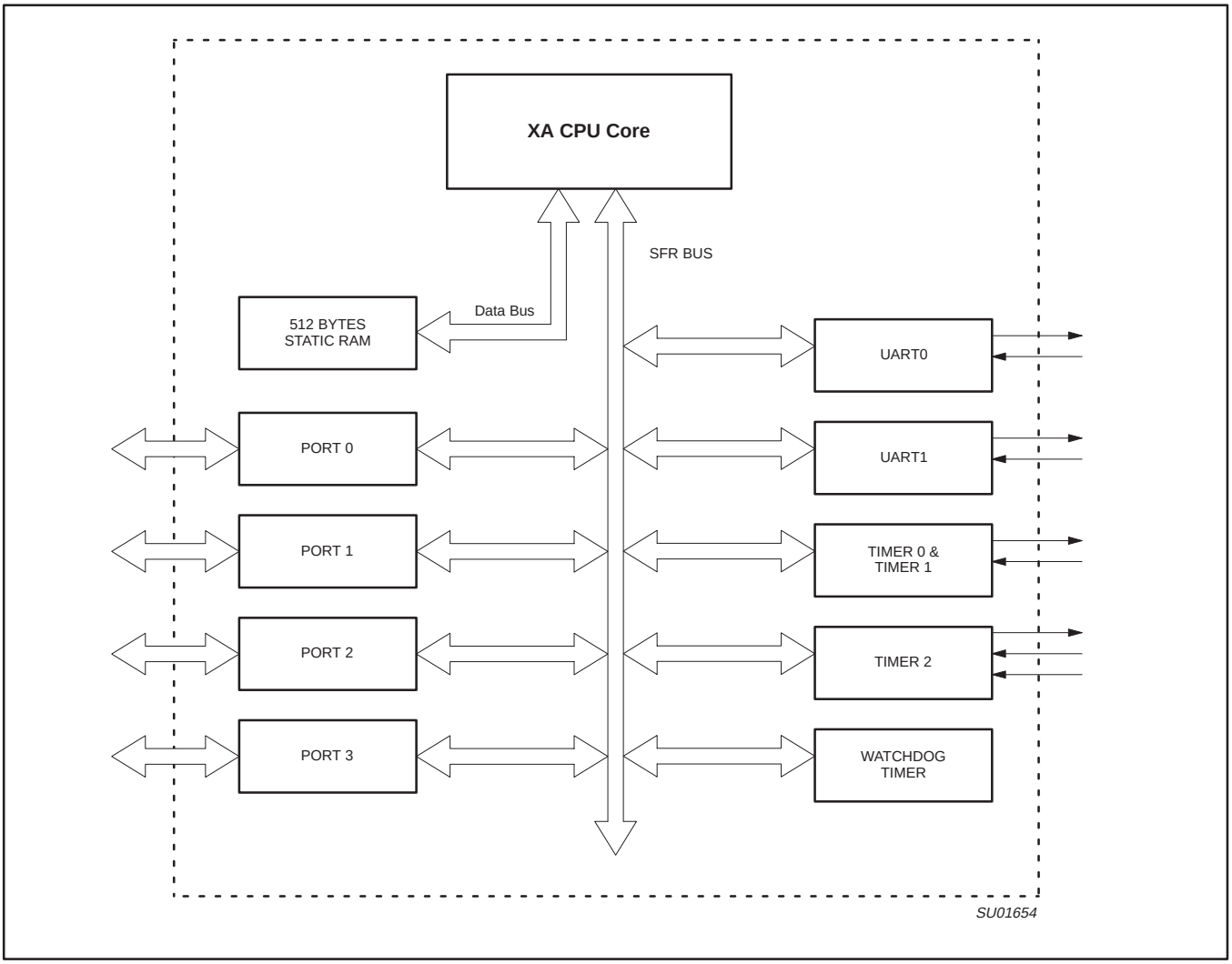
SPECIFIC FEATURES OF THE XA-G30

- 20-bit address range, 1 megabyte each program and data space. (Note that the XA architecture supports up to 24 bit addresses.)
- 2.7 V to 5.5 V operation
- 512 bytes of on-chip data RAM
- Three counter/timers with enhanced features (equivalent to 80C51 T0, T1, and T2)
- Watchdog timer
- Two enhanced UARTs
- Four 8-bit I/O ports with 4 programmable output configurations
- 44-pin PLCC and 44-pin LQFP packages

ORDERING INFORMATION

Type number	Package			
	Name	Description	Temperature Range (°C)	Version
PXAG30KBBD	LQFP44	plastic low profile quad flat package; 44 leads; body 10 × 10 × 1.4 mm	0 to +70	SOT389-1
PXAG30KBA	PLCC44	plastic leaded chip carrier; 44 leads	0 to +70	SOT187-2
PXAG30KFBD	LQFP44	plastic low profile quad flat package; 44 leads; body 10 × 10 × 1.4 mm	–40 to +85	SOT389-1
PXAG30KFA	PLCC44	plastic leaded chip carrier; 44 leads	–40 to +85	SOT187-2

BLOCK DIAGRAM



PIN DESCRIPTIONS

MNEMONIC	PIN. NO.		TYPE	NAME AND FUNCTION
	PLCC	LQFP		
V_{SS}	1, 22	16	I	**Ground:** 0 V reference.
V_{DD}	23, 44	17	I	**Power Supply:** This is the power supply voltage for normal, idle, and power down operation.
P0.0 – P0.7	43–36	37–30	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.
P1.0 – P1.7	2–9	40–44, 1–3	I/O	**Port 1:** Port 1 is an 8-bit I/O port with a user-configurable output type. Port 1 latches have 1s written to them and are configured in the quasi-bidirectional mode during reset. The operation of port 1 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. Port 1 also provides special functions as described below.
	2	40	O	**A0/$\overline{\text{WRH}}$:** Address bit 0 of the external address bus when the external data bus is configured for an 8 bit width. When the external data bus is configured for a 16 bit width, this pin becomes the high byte write strobe.
	3	41	O	**A1:** Address bit 1 of the external address bus.
	4	42	O	**A2:** Address bit 2 of the external address bus.
	5	43	O	**A3:** Address bit 3 of the external address bus.
	6	44	I	**RxD1 (P1.4):** Receiver input for serial port 1.
	7	1	O	**TxD1 (P1.5):** Transmitter output for serial port 1.
	8	2	I/O	**T2 (P1.6):** Timer/counter 2 external count input/clockout.
	9	3	I	**T2EX (P1.7):** Timer/counter 2 reload/capture/direction control
P2.0 – P2.7	24–31	18–25	I/O	**Port 2:** Port 2 is an 8-bit I/O port with a user-configurable output type. Port 2 latches have 1s written to them and are configured in the quasi-bidirectional mode during reset. The operation of port 2 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 in 16-bit mode, Port 2 becomes the multiplexed high data/instruction byte and address lines 12 through 19. When the external program/data bus is used in 8-bit mode, the number of address lines that appear on port 2 is user programmable.
P3.0 – P3.7	11, 13–19	5, 7–13	I/O	**Port 3:** Port 3 is an 8-bit I/O port with a user configurable output type. Port 3 latches have 1s written to them and are configured in the quasi-bidirectional mode during reset. the operation of port 3 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. Port 3 also provides various special functions as described below.
	11	5	I	**RxD0 (P3.0):** Receiver input for serial port 0.
	13	7	O	**TxD0 (P3.1):** Transmitter output for serial port 0.
	14	8	I	**$\overline{\text{INT0}}$ (P3.2):** External interrupt 0 input.
	15	9	I	**$\overline{\text{INT1}}$ (P3.3):** External interrupt 1 input.
	16	10	I/O	**T0 (P3.4):** Timer 0 external input, or timer 0 overflow output.
	17	11	I/O	**T1/BUSW (P3.5):** Timer 1 external input, or timer 1 overflow output. The value on this pin is latched as the external reset input is released and defines the default external data bus width (BUSW). 0 = 8-bit bus and 1 = 16-bit bus.
	18	12	O	**$\overline{\text{WRL}}$ (P3.6):** External data memory low byte write strobe.
	19	13	O	**$\overline{\text{RD}}$ (P3.7):** External data memory read strobe.
$\overline{\text{RST}}$	10	4	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. Refer to the section on Reset for details.
ALE/$\overline{\text{PROG}}$	33	27	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.

MNEMONIC	PIN. NO.		TYPE	NAME AND FUNCTION
	PLCC	LQFP		
$\overline{PSEN}$	32	26	O	**Program Store Enable:** The read strobe for external program memory. When the microcontroller accesses external program memory, $\overline{PSEN}$ is driven low in order to enable memory devices. $\overline{PSEN}$ is only active when external code accesses are performed.
$\overline{EA}$/WAIT	35	29	I	**External Access/Wait:** The $\overline{EA}$ input determines whether the internal program memory of the microcontroller is used for code execution. The value on the $\overline{EA}$ 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. $\overline{EA}$ must be LOW since the XA-G30 does not have on-chip code memory. After reset is released, this pin takes on the function of bus Wait input. If Wait is asserted high during any external bus access, that cycle will be extended until Wait is released.
XTAL1	21	15	I	**Crystal 1:** Input to the inverting amplifier used in the oscillator circuit and input to the internal clock generator circuits.
XTAL2	20	14	O	**Crystal 2:** Output from the oscillator amplifier.

SPECIAL FUNCTION REGISTERS

NAME	DESCRIPTION	SFR ADDRESS	BIT FUNCTIONS AND ADDRESSES MSB							LSB	RESET VALUE
BCR	Bus configuration register	46A	—	—	—	WAITD	BUSD	BC2	BC1	BC0	Note 1
BTRH	Bus timing register high byte	469	DW1	DW0	DWA1	DWA0	DR1	DR0	DRA1	DRA0	FF
BTRL	Bus timing register low byte	468	WM1	WM0	ALEW	—	CR1	CR0	CRA1	CRA0	EF
CS	Code segment	443									00
DS	Data segment	441									00
ES	Extra segment	442									00
			33F	33E	33D	33C	33B	33A	339	338	
IEH*	Interrupt enable high byte	427	—	—	—	—	ETI1	ERI1	ETI0	ERI0	00
			337	336	335	334	333	332	331	330	
IEL*	Interrupt enable low byte	426	EA	—	—	ET2	ET1	EX1	ET0	EX0	00
IPA0	Interrupt priority 0	4A0	—	PT0			—	PX0			00
IPA1	Interrupt priority 1	4A1	—	PT1			—	PX1			00
IPA2	Interrupt priority 2	4A2	—	—			—	PT2			00
IPA4	Interrupt priority 4	4A4	—	PTI0			—	PRI0			00
IPA5	Interrupt priority 5	4A5	—	PTI1			—	PRI1			00
			387	386	385	384	383	382	381	380	
P0*	Port 0	430	AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0	FF
			38F	38E	38D	38C	38B	38A	389	388	
P1*	Port 1	431	T2EX	T2	TxD1	RxD1	A3	A2	A1	WRH	FF
			397	396	395	394	393	392	391	390	
P2*	Port 2	432	P2.7	P2.6	P2.5	P2.4	P2.3	P2.2	P2.1	P2.0	FF

NAME	DESCRIPTION	SFR ADDRESS	BIT FUNCTIONS AND ADDRESSES MSB							LSB	RESET VALUE
			357	356	355	354	353	352	351	350	
SWR*	Software Interrupt Request	42A	—	SWR7	SWR6	SWR5	SWR4	SWR3	SWR2	SWR1	00
			2C7	2C6	2C5	2C4	2C3	2C2	2C1	2C0	
T2CON*	Timer 2 control register	418	TF2	EXF2	RCLK0	TCLK0	EXEN2	TR2	C/T2	CP/RL2	00
			2CF	2CE	2CD	2CC	2CB	2CA	2C9	2C8	
T2MOD*	Timer 2 mode control	419	—	—	RCLK1	TCLK1	—	—	T2OE	DCEN	00
TH2	Timer 2 high byte	459									00
TL2	Timer 2 low byte	458									00
T2CAPH	Timer 2 capture register, high byte	45B									00
T2CAPL	Timer 2 capture register, low byte	45A									00
			287	286	285	284	283	282	281	280	
TCON*	Timer 0 and 1 control register	410	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0	00
TH0	Timer 0 high byte	451									00
TH1	Timer 1 high byte	453									00
TL0	Timer 0 low byte	450									00
TL1	Timer 1 low byte	452									00
TMOD	Timer 0 and 1 mode control	45C	GATE	C/T	M1	M0	GATE	C/T	M1	M0	00
			28F	28E	28D	28C	28B	28A	289	288	
TSTAT*	Timer 0 and 1 extended status	411	—	—	—	—	—	T1OE	—	T0OE	00
			2FF	2FE	2FD	2FC	2FB	2FA	2F9	2F8	
WDCON*	Watchdog control register	41F	PRE2	PRE1	PRE0	—	—	WDRUN	WDTOF	—	Note 6
WDL	Watchdog timer reload	45F									00
WFEED1	Watchdog feed 1	45D									x
WFEED2	Watchdog feed 2	45E									x

NOTES:

* SFRs are bit addressable.

1. At reset, the BCR register is loaded with the binary value 0000 0a11, where “a” is the value on the BUSW pin. This defaults the address bus size to 20 bits since the XA-G30 has only 20 address lines.
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 EA pin. Thus all PnCFGA registers will contain FF and PnCFGB registers will contain 00. 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 XA-G30 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.

XA-G30 TIMER/COUNTERS

The XA has two standard 16-bit enhanced Timer/Counters: Timer 0 and Timer 1. Additionally, it has a third 16-bit Up/Down timer/counter, T2. A central timing generator in the XA core provides the time-base for all XA Timers and Counters. The timer/event counters can perform the following functions:

– Measure time intervals and pulse duration
– Count external events
– Generate interrupt requests
– Generate PWM or timed output waveforms

All of the timer/counters (Timer 0, Timer 1 and Timer 2) can be independently programmed to operate either as timers or event counters via the C/T bit in the TnCON register. All timers count up unless otherwise stated. These timers may be dynamically read during program execution.

The base clock rate of all of the timers is user programmable. This applies to timers T0, T1, and T2 when running in timer mode (as opposed to counter mode), and the watchdog timer. The clock driving the timers is called TCLK and is determined by the setting of two bits (PT1, PT0) in the System Configuration Register (SCR). The frequency of TCLK may be selected to be the oscillator input divided by 4 (Osc/4), the oscillator input divided by 16 (Osc/16), or the oscillator input divided by 64 (Osc/64). This gives a range of possibilities for the XA timer functions, including baud rate generation, Timer 2 capture. Note that this single rate setting applies to all of the timers.

When timers T0, T1, or T2 are used in the counter mode, the register will increment whenever a falling edge (high to low transition) is detected on the external input pin corresponding to the timer clock. These inputs are sampled once every 2 oscillator cycles, so it can take as many as 4 oscillator cycles to detect a transition. Thus the maximum count rate that can be supported is Osc/4. The duty cycle of the timer clock inputs is not important, but any high or low state on the timer clock input pins must be present for 2 oscillator cycles before it is guaranteed to be “seen” by the timer logic.

Timer 0 and Timer 1

The “Timer” or “Counter” function is selected by control bits C/T in the special function register TMOD. These two Timer/Counters have four operating modes, which are selected by bit-pairs (M1, M0) in the TMOD register. Timer modes 1, 2, and 3 in XA are kept identical to the 80C51 timer modes for code compatibility. Only the mode 0 is replaced in the XA by a more powerful 16-bit auto-reload mode. This will give the XA timers a much larger range when used as time bases.

The recommended M1, M0 settings for the different modes are shown in Figure 2.

SCR Address:440
Not Bit Addressable
Reset Value: 00H

MSB							LSB
—	—	—	—	PT1	PT0	CM	PZ

PT1	PT0	OPERATING
		Prescaler selection.
0	0	Osc/4
0	1	Osc/16
1	0	Osc/64
1	1	Reserved
CM		Compatibility Mode allows the XA to execute most translated 80C51 code on the XA. The XA register file must copy the 80C51 mapping to data memory and mimic the 80C51 indirect addressing scheme.
PZ		Page Zero mode forces all program and data addresses to 16-bits only. This saves stack space and speeds up execution but limits memory access to 64k.

SU00589

Figure 1. System Configuration Register (SCR)

TMOD Address:45C
Not Bit Addressable
Reset Value: 00H

MSB							LSB
GATE	C/$\overline{T}$	M1	M0	GATE	C/$\overline{T}$	M1	M0
TIMER 1				TIMER 0			

GATE	Gating control when set. Timer/Counter “n” is enabled only while “$\overline{INTn}$” pin is high and “TRn” control bit is set. When cleared Timer “n” is enabled whenever “TRn” control bit is set.
C/$\overline{T}$	Timer or Counter Selector cleared for Timer operation (input from internal system clock.) Set for Counter operation (input from “Tn” input pin).

M1	M0	OPERATING
0	0	16-bit auto-reload timer/counter
0	1	16-bit non-auto-reload timer/counter
1	0	8-bit auto-reload timer/counter
1	1	Dual 8-bit timer mode (timer 0 only)

SU00605

Figure 2. Timer/Counter Mode Control (TMOD) Register

New Enhanced Mode 0

For timers T0 or T1 the 13-bit count mode on the 80C51 (current Mode 0) has been replaced in the XA with a 16-bit auto-reload mode. Four additional 8-bit data registers (two per timer: RTHn and RTLn) are created to hold the auto-reload values. In this mode, the TH overflow will set the TF flag in the TCON register and cause both the TL and TH counters to be loaded from the RTL and RTH registers respectively.

These new SFRs will also be used to hold the TL reload data in the 8-bit auto-reload mode (Mode 2) instead of TH.

The overflow rate for Timer 0 or Timer 1 in Mode 0 may be calculated as follows:

Timer_Rate = Osc / (N * (65536 – Timer_Reload_Value))

where N = the TCLK prescaler value: 4 (default), 16, or 64.

Mode 1

Mode 1 is the 16-bit non-auto reload mode.

Mode 2

Mode 2 configures the Timer register as an 8-bit Counter (TLn) with automatic reload. Overflow from TLn not only sets TFn, but also reloads TLn with the contents of RTLn, which is preset by software. The reload leaves THn unchanged.

Mode 2 operation is the same for Timer/Counter 0.

The overflow rate for Timer 0 or Timer 1 in Mode 2 may be calculated as follows:

Timer_Rate = Osc / (N * (256 – Timer_Reload_Value))

where N = the TCLK prescaler value: 4, 16, or 64.

Mode 3

Timer 1 in Mode 3 simply holds its count. The effect is the same as setting TR1 = 0.

Timer 0 in Mode 3 establishes TL0 and TH0 as two separate counters. TL0 uses the Timer 0 control bits: C/T, GATE, TR0, INT0, and TF0. TH0 is locked into a timer function and takes over the use of TR1 and TF1 from Timer 1. Thus, TH0 now controls the “Timer 1” interrupt.

Mode 3 is provided for applications requiring an extra 8-bit timer. When Timer 0 is in Mode 3, Timer 1 can be turned on and off by switching it out of and into its own Mode 3, or can still be used by the serial port as a baud rate generator, or in fact, in any application not requiring an interrupt.

TCON Address:410
Bit Addressable
Reset Value: 00H

MSB							LSB
TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0

BIT	SYMBOL	FUNCTION
TCON.7	TF1	Timer 1 overflow flag. Set by hardware on Timer/Counter overflow. This flag will not be set if T1OE (TSTAT.2) is set. Cleared by hardware when processor vectors to interrupt routine, or by clearing the bit in software.
TCON.6	TR1	Timer 1 Run control bit. Set/cleared by software to turn Timer/Counter 1 on/off.
TCON.5	TF0	Timer 0 overflow flag. Set by hardware on Timer/Counter overflow. This flag will not be set if T0OE (TSTAT.0) is set. Cleared by hardware when processor vectors to interrupt routine, or by clearing the bit in software.
TCON.4	TR0	Timer 0 Run control bit. Set/cleared by software to turn Timer/Counter 0 on/off.
TCON.3	IE1	Interrupt 1 Edge flag. Set by hardware when external interrupt edge detected. Cleared when interrupt processed.
TCON.2	IT1	Interrupt 1 type control bit. Set/cleared by software to specify falling edge/low level triggered external interrupts.
TCON.1	IE0	Interrupt 0 Edge flag. Set by hardware when external interrupt edge detected. Cleared when interrupt processed.
TCON.0	IT0	Interrupt 0 Type control bit. Set/cleared by software to specify falling edge/low level triggered external interrupts.

SU00604C

Figure 3. Timer/Counter Control (TCON) Register

timer register is loaded with FFFF hex. The underflow also sets the TF2 flag, which can generate an interrupt if enabled.

The external flag EXF2 toggles when Timer 2 underflows or overflows. This EXF2 bit can be used as a 17th bit of resolution, if needed. the EXF2 flag does not generate an interrupt in this mode. As the baud rate generator, timer T2 is incremented by TCLK.

Baud Rate Generator Mode
By setting the TCLKn and/or RCLKn in T2CON or T2MOD, the Timer 2 can be chosen as the baud rate generator for either or both UARTs. The baud rates for transmit and receive can be simultaneously different.

Programmable Clock-Out

A 50% duty cycle clock can be programmed to come out on P1.6. This pin, besides being a regular I/O pin, has two alternate functions. It can be programmed (1) to input the external clock for Timer/Counter 2 or (2) to output a 50% duty cycle clock ranging from 3.58Hz to 3.75MHz at a 30MHz operating frequency.

To configure the Timer/Counter 2 as a clock generator, bit C/T2 (in T2CON) must be cleared and bit T20E in T2MOD must be set. Bit TR2 (T2CON.2) also must be set to start the timer.

The Clock-Out frequency depends on the oscillator frequency and the reload value of Timer 2 capture registers (TCAP2H, TCAP2L) as shown in this equation:

$$\frac{\text{TCLK}}{2 \times (65536 - \text{TCAP2H, TCAP2L})}$$

In the Clock-Out mode Timer 2 roll-overs will not generate an interrupt. This is similar to when it is used as a baud-rate generator. It is possible to use Timer 2 as a baud-rate generator and a clock generator simultaneously. Note, however, that the baud-rate will be 1/8 of the Clock-Out frequency.

Table 1. Timer 2 Operating Modes

TR2	CP/RL2	RCLK+TCLK	DCEN	MODE
0	X	X	X	Timer off (stopped)
1	0	0	0	16-bit auto-reload, counting up
1	0	0	1	16-bit auto-reload, counting up or down depending on T2EX pin
1	1	0	X	16-bit capture
1	X	1	X	Baud rate generator

TSTAT Address:411
Bit Addressable
Reset Value: 00H

MSB							LSB
—	—	—	—	—	T1OE	—	T0OE

BIT	SYMBOL	FUNCTION
TSTAT.2	T1OE	When 0, this bit allows the T1 pin to clock Timer 1 when in the counter mode. When 1, T1 acts as an output and toggles at every Timer 1 overflow.
TSTAT.0	T0OE	When 0, this bit allows the T0 pin to clock Timer 0 when in the counter mode. When 1, T0 acts as an output and toggles at every Timer 0 overflow.

SU00612B

Figure 5. Timer 0 And 1 Extended Status (TSTAT)

T2MOD Address:419
Bit Addressable
Reset Value: 00H

MSB							LSB
—	—	RCLK1	TCLK1	—	—	T2OE	DCEN

BIT	SYMBOL	FUNCTION
T2MOD.5	RCLK1	Receive Clock Flag.
T2MOD.4	TCLK1	Transmit Clock Flag. RCLK1 and TCLK1 are used to select Timer 2 overflow rate as a clock source for UART1 instead of Timer T1.
T2MOD.1	T2OE	When 0, this bit allows the T2 pin to clock Timer 2 when in the counter mode. When 1, T2 acts as an output and toggles at every Timer 2 overflow.
T2MOD.0	DCEN	Controls count direction for Timer 2 in autoreload mode. DCEN=0 counter set to count up only DCEN=1 counter set to count up or down, depending on T2EX (see text).

SU00610B

Figure 6. Timer 2 Mode Control (T2MOD)

WATCHDOG TIMER

The watchdog timer subsystem protects the system from incorrect code execution by causing a system reset when the watchdog timer underflows as a result of a failure of software to feed the timer prior to the timer reaching its terminal count. It is important to note that the watchdog timer is running after any type of reset and must be turned off by user software if the application does not use the watchdog function.

Watchdog Function

The watchdog consists of a programmable prescaler and the main timer. The prescaler derives its clock from the TCLK source that also drives timers 0, 1, and 2. The watchdog timer subsystem consists of a programmable 13-bit prescaler, and an 8-bit main timer. The main timer is clocked (decremented) by a tap taken from one of the top 8-bits of the prescaler as shown in Figure 10. The clock source for the prescaler is the same as TCLK (same as the clock source for the timers). Thus the main counter can be clocked as often as once every 32 TCLKs (see Table 2). The watchdog generates an underflow signal (and is autoloaded from WDL) when the watchdog is at count 0 and the clock to decrement the watchdog occurs. The watchdog is 8 bits wide and the autoload value can range from 0 to FFH. (The autoload value of 0 is permissible since the prescaler is cleared upon autoload).

This leads to the following user design equations. Definitions: t_{OSC} is the oscillator period, N is the selected prescaler tap value, W is the main counter autoload value, P is the prescaler value from Table 2, t_{MIN} is the minimum watchdog time-out value (when the autoload value is 0), t_{MAX} is the maximum time-out value (when the autoload value is FFH), t_D is the design time-out value.

$t_{MIN} = t_{OSC} \times 4 \times 32$ (W = 0, N = 4)

$t_{MAX} = t_{OSC} \times 64 \times 4096 \times 256$ (W = 255, N = 64)

$t_D = t_{OSC} \times N \times P \times (W + 1)$

The watchdog timer is not directly loadable by the user. Instead, the value to be loaded into the main timer is held in an autoload register. In order to cause the main timer to be loaded with the appropriate value, a special sequence of software action must take place. This operation is referred to as feeding the watchdog timer.

To feed the watchdog, two instructions must be sequentially executed successfully. No intervening SFR accesses are allowed, so interrupts should be disabled before feeding the watchdog. The instructions should move A5H to the WFEED1 register and then 5AH to the WFEED2 register. If WFEED1 is correctly loaded and WFEED2 is not correctly loaded, then an immediate watchdog reset will occur. The program sequence to feed the watchdog timer or cause new WDCON settings to take effect is as follows:

```
clr     ea            ; disable global interrupts.
mov.b   wfeed1,#A5h   ; do watchdog feed part 1
mov.b   wfeed2,#5Ah   ; do watchdog feed part 2
setb    ea            ; re-enable global interrupts.
```

This sequence assumes that the XA interrupt system is enabled and there is a possibility of an interrupt request occurring during the feed sequence. If an interrupt was allowed to be serviced and the service routine contained any SFR access, it would trigger a watchdog reset. If it is known that no interrupt could occur during the feed sequence, the instructions to disable and re-enable interrupts may be removed.

The software must be written so that a feed operation takes place every t_D seconds from the last feed operation. Some tradeoffs may need to be made. It is not advisable to include feed operations in minor loops or in subroutines unless the feed operation is a specific subroutine.

To turn the watchdog timer completely off, the following code sequence should be used:

```
mov.b   wdcon,#0      ; set WD control register to clear WDRUN.
mov.b   wfeed1,#A5h   ; do watchdog feed part 1
mov.b   wfeed2,#5Ah   ; do watchdog feed part 2
```

This sequence assumes that the watchdog timer is being turned off at the beginning of initialization code and that the XA interrupt system has not yet been enabled. If the watchdog timer is to be turned off at a point when interrupts may be enabled, instructions to disable and re-enable interrupts should be added to this sequence.

Watchdog Control Register (WDCON)

The reset values of the WDCON and WDL registers will be such that the watchdog timer has a timeout period of $4 \times 4096 \times t_{OSC}$ and the watchdog is running. WDCON can be written by software but the changes only take effect after executing a valid watchdog feed sequence.

Table 2. Prescaler Select Values in WDCON

PRE2	PRE1	PRE0	DIVISOR
0	0	0	32
0	0	1	64
0	1	0	128
0	1	1	256
1	0	0	512
1	0	1	1024
1	1	0	2048
1	1	1	4096

Watchdog Detailed Operation

When external $\overline{\text{RESET}}$ is applied, the following takes place:

- Watchdog run control bit set to ON (1).
- Autoload register WDL set to 00 (min. count).
- Watchdog time-out flag cleared.
- Prescaler is cleared.
- Prescaler tap set to the highest divide.
- Autoload takes place.

When coming out of a hardware reset, the software should load the autoload register and then feed the watchdog (cause an autoload).

If the watchdog is running and happens to underflow at the time the external RESET is applied, the watchdog time-out flag will be cleared.

Serial Port Control Register
The serial port control and status register is the Special Function Register SnCON, shown in Figure 12. This register contains not only the mode selection bits, but also the 9th data bit for transmit and receive (TB8_n and RB8_n), and the serial port interrupt bits (TI_n and RI_n).

TI Flag
In order to allow easy use of the double buffered UART transmitter feature, the TI_n flag is set by the UART hardware under two conditions. The first condition is the completion of any byte transmission. This occurs at the end of the stop bit in modes 1, 2, or 3, or at the end of the eighth data bit in mode 0. The second condition is when SnBUF is written while the UART transmitter is idle. In this case, the TI_n flag is set in order to indicate that the second UART transmitter buffer is still available.

Typically, UART transmitters generate one interrupt per byte transmitted. In the case of the XA UART, one additional interrupt is generated as defined by the stated conditions for setting the TI_n flag. This additional interrupt does not occur if double buffering is bypassed as explained below. Note that if a character oriented approach is used to transmit data through the UART, there could be a second interrupt for each character transmitted, depending on the timing of the writes to SBUF. For this reason, it is generally better to bypass double buffering when the UART transmitter is used in character oriented mode. This is also true if the UART is polled rather than interrupt driven, and when transmission is character oriented rather than message or string oriented. The interrupt occurs at the end of the last byte transmitted when the UART becomes idle. Among other things, this allows a program to determine when a message has been transmitted completely. The interrupt service routine should handle this additional interrupt.

The recommended method of using the double buffering in the application program is to have the interrupt service routine handle a single byte for each interrupt occurrence. In this manner the program essentially does not require any special considerations for double buffering. Unless higher priority interrupts cause delays in the servicing of the UART transmitter interrupt, the double buffering will result in transmitted bytes being tightly packed with no intervening gaps.

9-bit Mode
Please note that the ninth data bit (TB8) is not double buffered. Care must be taken to insure that the TB8 bit contains the intended data at the point where it is transmitted. Double buffering of the UART transmitter may be bypassed as a simple means of synchronizing TB8 to the rest of the data stream.

Bypassing Double Buffering
The UART transmitter may be used as if it is single buffered. The recommended UART transmitter interrupt service routine (ISR) technique to bypass double buffering first clears the TI_n flag upon entry into the ISR, as in standard practice. This clears the interrupt that activated the ISR. Secondly, the TI_n flag is cleared immediately following each write to SnBUF. This clears the interrupt flag that would otherwise direct the program to write to the second transmitter buffer. If there is any possibility that a higher priority interrupt might become active between the write to SnBUF and the clearing of the TI_n flag, the interrupt system may have to be temporarily disabled during that sequence by clearing, then setting the EA bit in the IEL register.

Note Regarding Older XA-G30 Devices
Older versions of the XA-G30, XA-G37, and XA-G35 emulation bondout devices do not have the double buffering feature enabled. Contact factory for details.

CLOCKING SCHEME/BAUD RATE GENERATION

The XA UARTS clock rates are determined by either a fixed division (modes 0 and 2) of the oscillator clock or by the Timer 1 or Timer 2 overflow rate (modes 1 and 3).

The clock for the UARTs in XA runs at 16x the Baud rate. If the timers are used as the source for Baud Clock, since maximum speed of timers/Baud Clock is Osc/4, the maximum baud rate is timer overflow divided by 16 i.e. Osc/64.

In Mode 0, it is fixed at Osc/16. In Mode 2, however, the fixed rate is Osc/32.

Pre-scaler for all Timers T0,1,2 controlled by PT1, PT0 bits in SCR	00	Osc/4
	01	Osc/16
	10	Osc/64
	11	reserved

Baud Rate for UART Mode 0:

Baud_Rate = Osc/16

Baud Rate calculation for UART Mode 1 and 3:

Baud_Rate = Timer_Rate/16

Timer_Rate = Osc/(N*(Timer_Range– Timer_Reload_Value))

where N = the TCLK prescaler value: 4, 16, or 64.
and Timer_Range = 256 for timer 1 in mode 2.
65536 for timer 1 in mode 0 and timer 2 in count up mode.

The timer reload value may be calculated as follows:

Timer_Reload_Value = Timer_Range–(Osc/(Baud_Rate*N*16))

NOTES:

1. The maximum baud rate for a UART in mode 1 or 3 is Osc/64.
2. The lowest possible baud rate (for a given oscillator frequency and N value) may be found by using a timer reload value of 0.
3. The timer reload value may never be larger than the timer range.
4. If a timer reload value calculation gives a negative or fractional result, the baud rate requested is not possible at the given oscillator frequency and N value.

Baud Rate for UART Mode 2:

Baud_Rate = Osc/32

Using Timer 2 to Generate Baud Rates

Timer T2 is a 16-bit up/down counter in XA. As a baud rate generator, timer 2 is selected as a clock source for either/both UART0 and UART1 transmitters and/or receivers by setting TCLKn and/or RCLKn in T2CON and T2MOD. As the baud rate generator, T2 is incremented as Osc/N where N = 4, 16 or 64 depending on TCLK as programmed in the SCR bits PT1, and PTO. So, if T2 is the source of one UART, the other UART could be clocked by either T1 overflow or fixed clock, and the UARTs could run independently with different baud rates.

T2CON 0x418		bit5	bit4	
		RCLK0	TCLK0	

T2MOD 0x419		bit5	bit4	
		RCLK1	TCLK1	

Prescaler Select for Timer Clock (TCLK)

SCR 0x440		bit3	bit2	
		PT1	PT0	

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 11. Serial Port Extended Status (SnSTAT) Register (See also Figure 13 regarding Framing Error flag)

I/O PORT OUTPUT CONFIGURATION

Each I/O port pin can be user configured to one of 4 output types. The types are Quasi-bidirectional (essentially the same as standard 80C51 family I/O ports), Open-Drain, Push-Pull, and Off (high impedance). The default configuration after reset is Quasi-bidirectional. However, in the ROMless mode (the $\overline{EA}$ pin is low at reset), the port pins that comprise the external data bus will default to push-pull outputs.

I/O port output configurations are determined by the settings in port configuration SFRs. There are 2 SFRs for each port, called PnCFGA and PnCFGB, where “n” is the port number. One bit in each of the 2 SFRs relates to the output setting for the corresponding port pin, allowing any combination of the 2 output types to be mixed on those port pins. For instance, the output type of port 1 pin 3 is controlled by the setting of bit 3 in the SFRs P1CFGA and P1CFGB.

Table 4 shows the configuration register settings for the 4 port output types. The electrical characteristics of each output type may be found in the DC Characteristic table.

Table 4. Port Configuration Register Settings

PnCFGB	PnCFGA	Port Output Mode
0	0	Open Drain
0	1	Quasi-bidirectional
1	0	Off (high impedance)
1	1	Push-Pull

NOTE:
Mode changes may cause glitches to occur during transitions. When modifying both registers, WRITE instructions should be carried out consecutively.

EXTERNAL BUS

The external program/data bus allows for 8-bit or 16-bit bus width, and address sizes from 12 to 20 bits. The bus width is selected by an input at reset (see Reset Options below), while the address size is set by the program in a configuration register. If all off-chip code is selected (through the use of the $\overline{EA}$ pin), the initial code fetches will be done with the maximum address size (20 bits).

RESET

The device is reset whenever a logic “0“ is applied to $\overline{RST}$ for at least 10 microseconds, placing a low level on the pin re-initializes the on-chip logic. Reset must be asserted when power is initially applied to the XA and held until the oscillator is running.

The duration of reset must be extended when power is initially applied or when using reset to exit power down mode. This is due to the need to allow the oscillator time to start up and stabilize. For most power supply ramp up conditions, this time is 10 milliseconds.

As it is brought high again, an exception is generated which causes the processor to jump to the address contained in the memory location 0000. The destination of the reset jump must be located in the first 64k of code address on power-up, all vectors are 16-bit values and so point to page zero addresses only. After a reset the RAM contents are indeterminate.

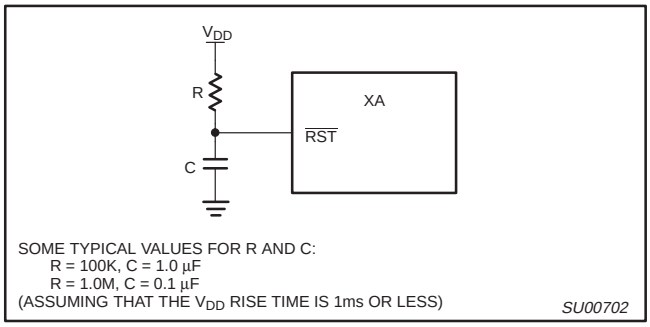


Figure 15. Recommended Reset Circuit

RESET OPTIONS

The $\overline{EA}$ pin is sampled on the rising edge of the $\overline{RST}$ pulse, and determines whether the device is to begin execution from internal or external code memory. $\overline{EA}$ pulled high configures the XA in single-chip mode. If $\overline{EA}$ is driven low, the device enters ROMless mode. After Reset is released, the $\overline{EA}$/WAIT pin becomes a bus wait signal for external bus transactions.

The BUSW/P3.5 pin is weakly pulled high while reset is asserted, allowing simple biasing of the pin with a resistor to ground to select the alternate bus width. If the BUSW pin is not driven at reset, the weak pullup will cause a 1 to be loaded for the bus width, giving a 16-bit external bus. BUSW may be pulled low with a 2.7K or smaller value resistor, giving an 8-bit external bus. The bus width setting from the BUSW pin may be overridden by software once the user program is running.

Both $\overline{EA}$ and BUSW must be held for three oscillator clock times after reset is deasserted to guarantee that their values are latched correctly.

POWER REDUCTION MODES

The XA-G30 supports Idle and Power Down modes of power reduction. The idle mode leaves some peripherals running to allow them to wake up the processor when an interrupt is generated. The power down mode stops the oscillator in order to minimize power. The processor can be made to exit power down mode via reset or one of the external interrupt inputs. In order to use an external interrupt to re-activate the XA while in power down mode, the external interrupt must be enabled and be configured to level sensitive mode. In power down mode, the power supply voltage may be reduced to the RAM keep-alive voltage (2 V), retaining the RAM, register, and SFR values at the point where the power down mode was entered.

V3) This variable represents the programmed length of an entire code read cycle **with** ALE. This time is determined by the CRA1 and CRA0 bits in the BTRL register. V3 = the total bus cycle duration (2 if CRA1/0 = 00, 3 if CRA1/0 = 01, 4 if CRA1/0 = 10, and 5 if CRA1/0 = 11).

V4) This variable represents the programmed length of an entire code read cycle with **no** ALE. This time is determined by the CR1 and CR0 bits in the BTRL register. V4 = 1 if CR1/0 = 00, 2 if CR1/0 = 01, 3 if CR1/0 = 10, and 4 if CR1/0 = 11.

V5) This variable represents the programmed length of an entire data read cycle with **no** ALE. this time is determined by the DR1 and DR0 bits in the BTRH register. V5 = 1 if DR1/0 = 00, 2 if DR1/0 = 01, 3 if DR1/0 = 10, and 4 if DR1/0 = 11.

V6) This variable represents the programmed length of an entire data read cycle **with** ALE. The time is determined by the DRA1 and DRA0 bits in the BTRH register. V6 = 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).

V7) This variable represents the programmed width of the $\overline{RD}$ pulse as determined by the DR1 and DR0 bits or the DRA1, DRA0 in the BTRH register, and the ALEW bit in the BTRL register. Note that during a 16-bit operation on an 8-bit external bus, $\overline{RD}$ 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 WRL and/or 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 $\overline{WRL}$ and/or $\overline{WRH}$ 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 $\overline{WRL}$ and/or $\overline{WRH}$ 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 = 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 setup and hold times. The strobe may be $\overline{RD}$ (for data read cycles), $\overline{WRL}$ and/or $\overline{WRH}$ (for data write cycles), or $\overline{PSEN}$ (for code read cycles), depending on the type of bus cycle being widened by WAIT. V10 = V2 for WAIT associated with a code read cycle using $\overline{PSEN}$. V10 = V8 for a data write cycle using $\overline{WRL}$ and/or $\overline{WRH}$. V10 = V7–1 for a data read cycle using $\overline{RD}$. 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 $\overline{RD}$ 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 $\overline{WRL}$ and/or $\overline{WRH}$ 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 $\overline{WRL}$ and/or $\overline{WRH}$ 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 DWA1/0 = 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 $\overline{WRL}$ and/or $\overline{WRH}$ 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 $\overline{WRL}$ and/or $\overline{WRH}$ 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 does not have $\overline{PSEN}$ edges in every fetch cycle. Thus, if WAIT is used to delay code fetch cycles, a change in the low order address lines must be detected to locate the beginning of a cycle. This would be A3–A0 for an 8-bit bus, and A3–A1 for a 16-bit bus. Also, a 16-bit data read operation conducted on a 8-bit wide bus similarly does not include two separate $\overline{RD}$ 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 $\overline{WR}$ strobe. This is not usually the case, and in most applications this parameter is not used.
6. Please note that the XA-G30 requires that extended data bus hold time (WM0 = 1) to be used with external bus write cycles.

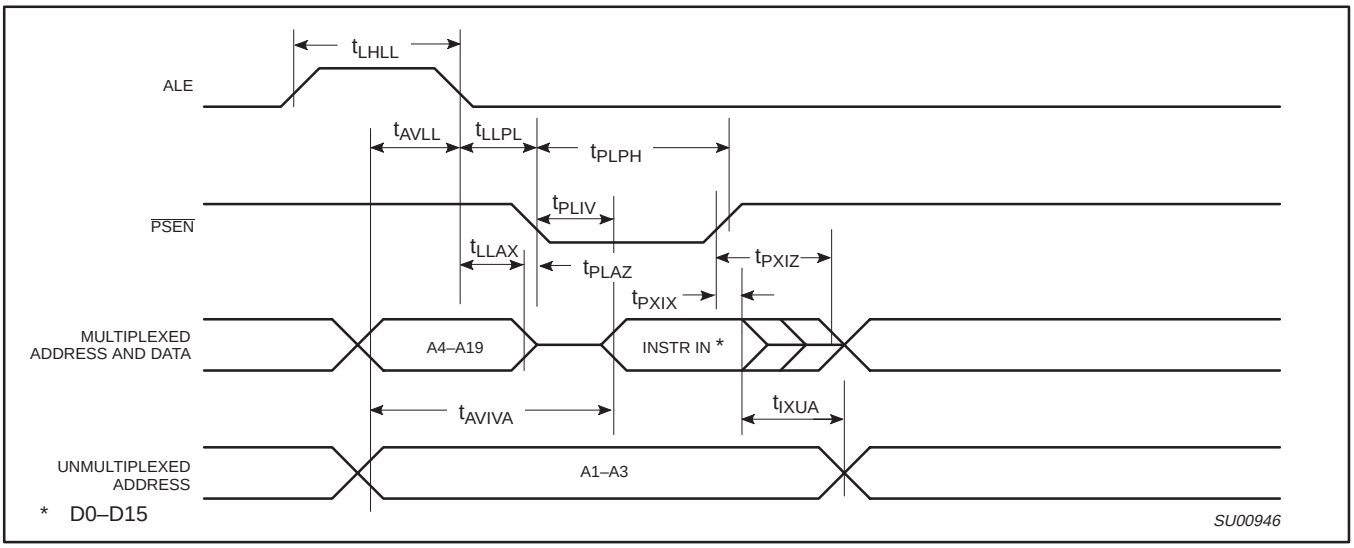


Figure 16. External Program Memory Read Cycle (ALE Cycle)

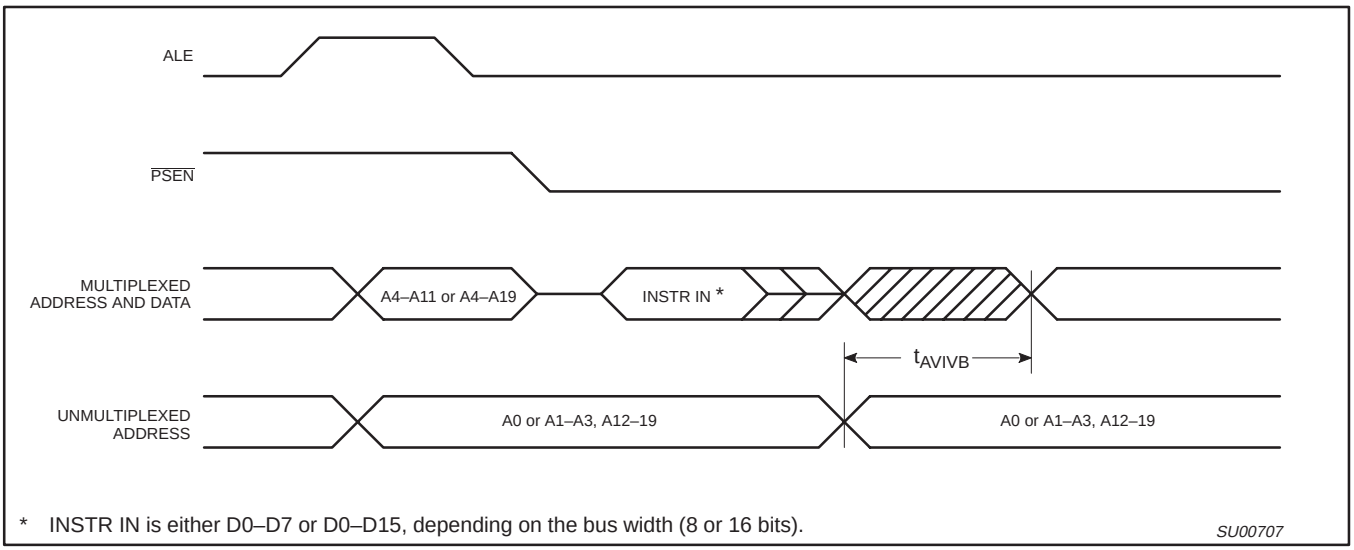


Figure 17. External Program Memory Read Cycle (Non-ALE Cycle)

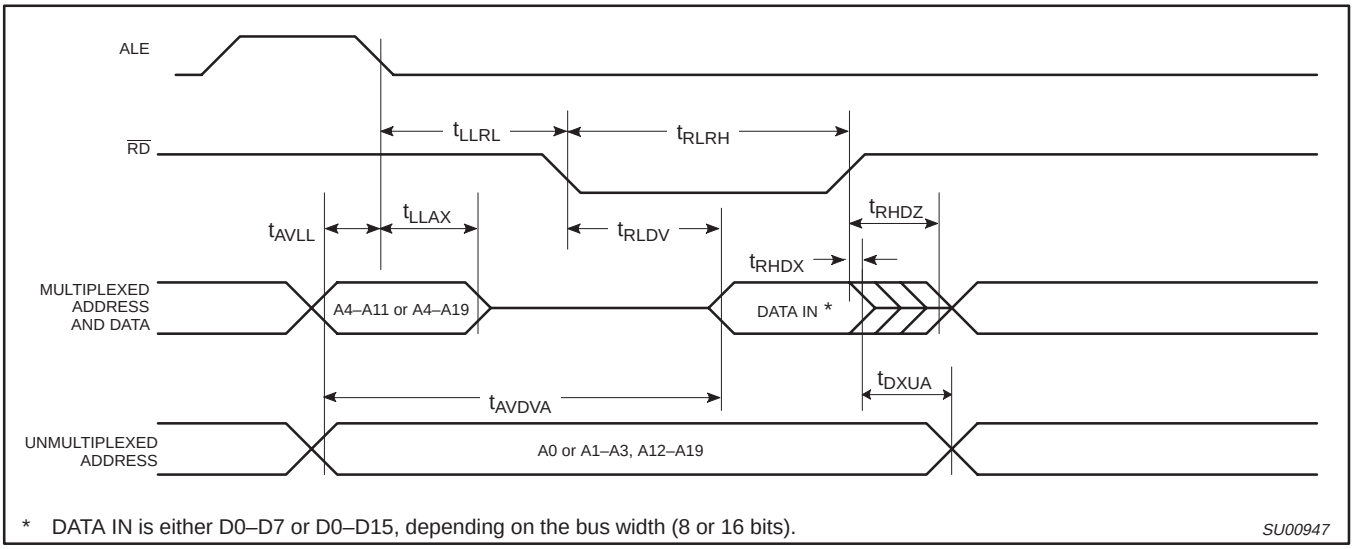


Figure 18. External Data Memory Read Cycle (ALE Cycle)

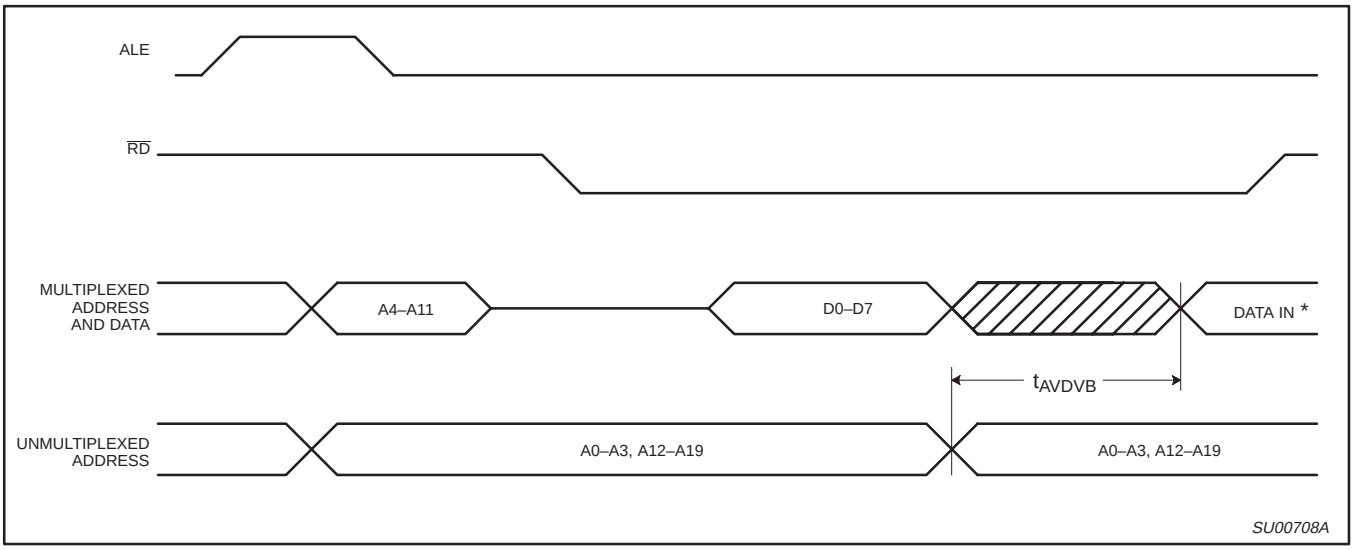


Figure 19. External Data Memory Read Cycle (Non-ALE Cycle) 8 Bit Bus Only

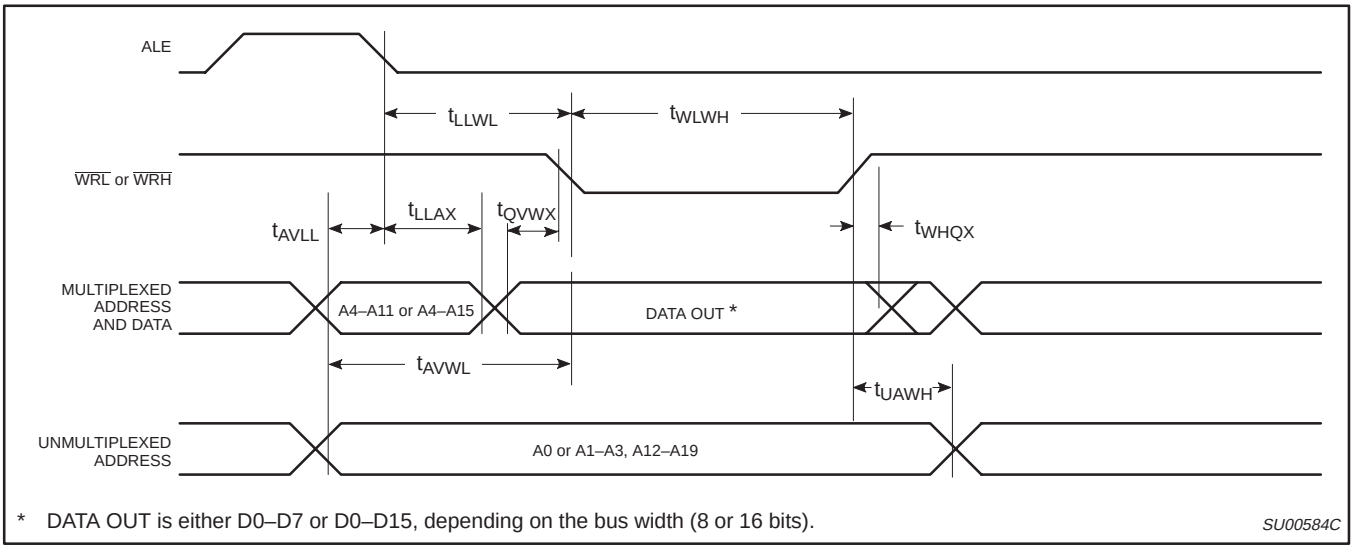


Figure 20. External Data Memory Write Cycle

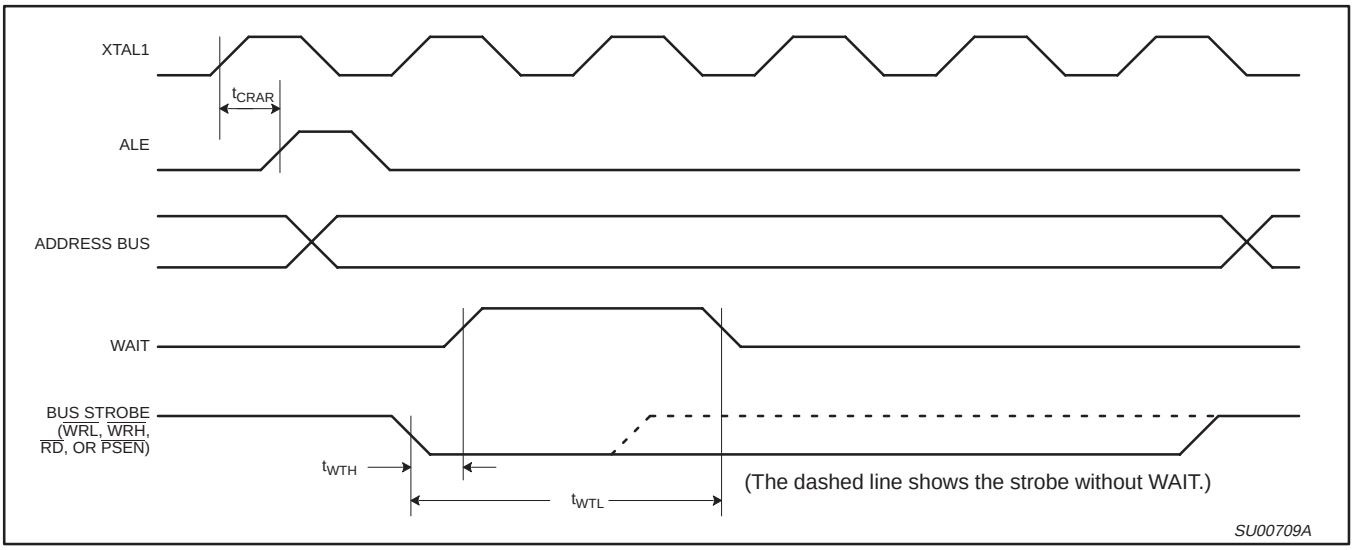


Figure 21. WAIT Signal Timing

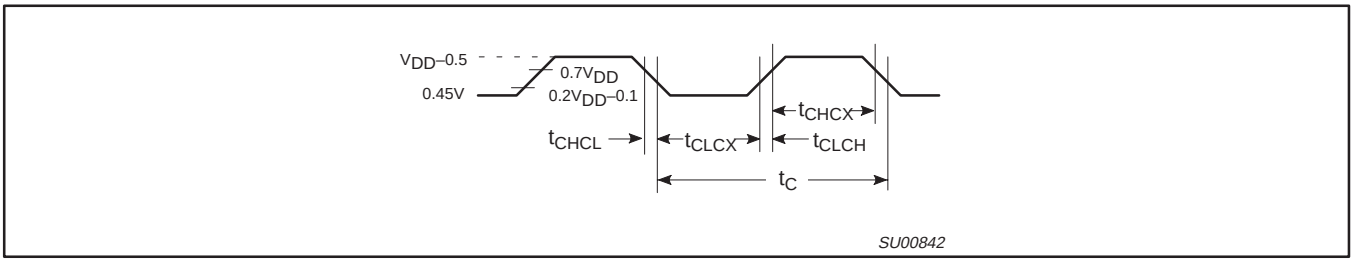


Figure 22. External Clock Drive

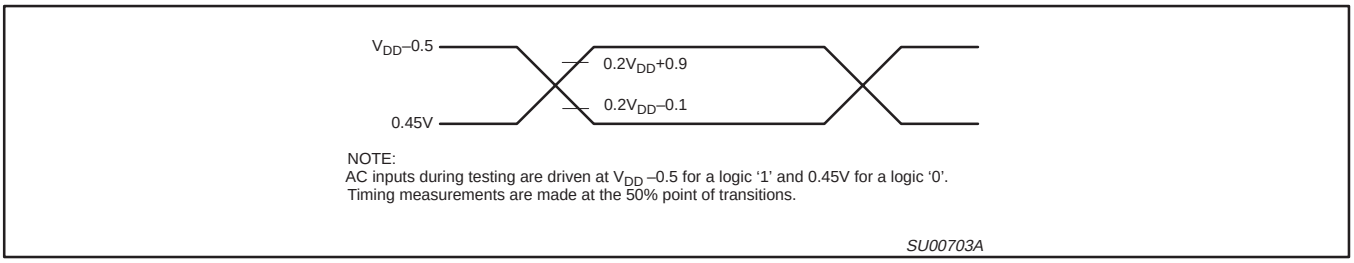


Figure 23. AC Testing Input/Output

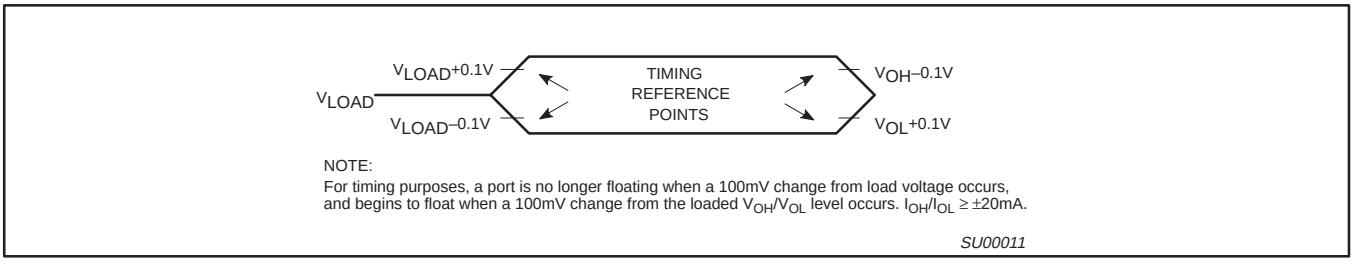


Figure 24. Float Waveform

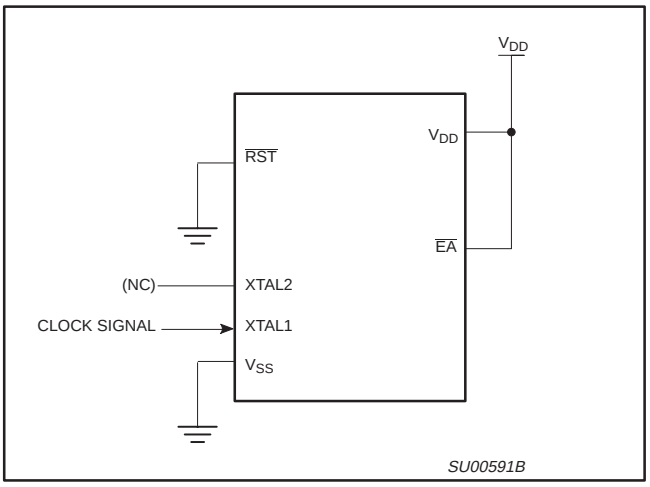


Figure 25. I_{DD} Test Condition, Active Mode
All other pins are disconnected

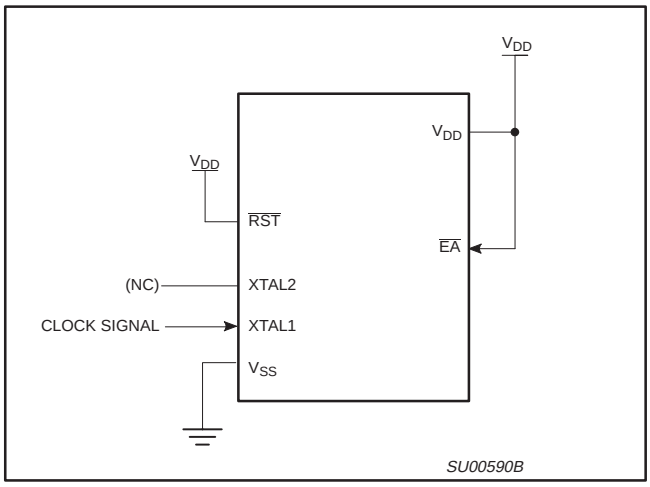


Figure 26. I_{DD} Test Condition, Idle Mode
All other pins are disconnected

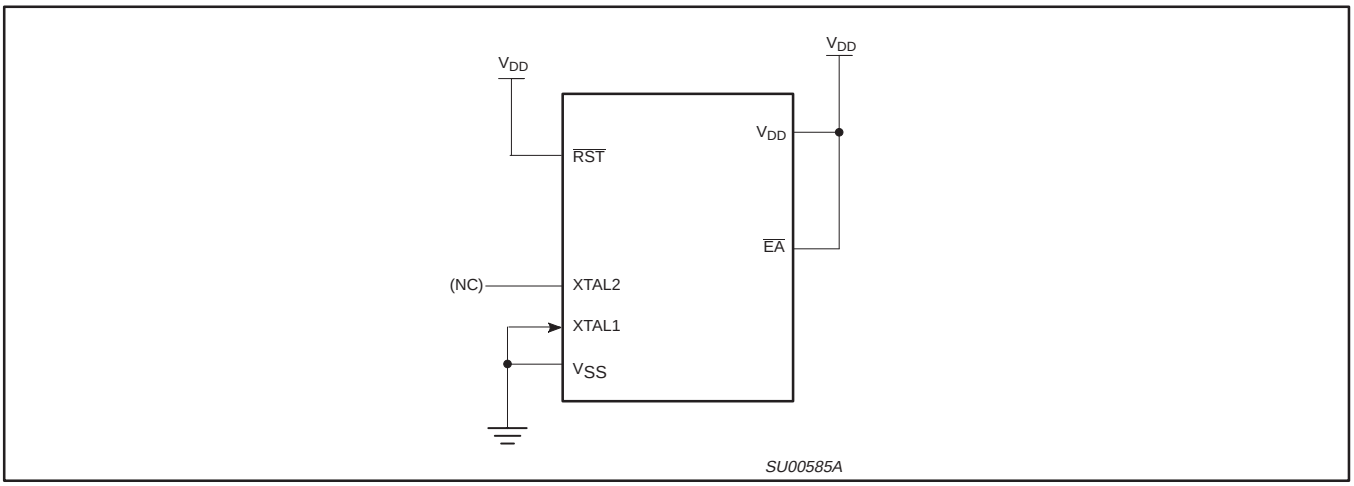
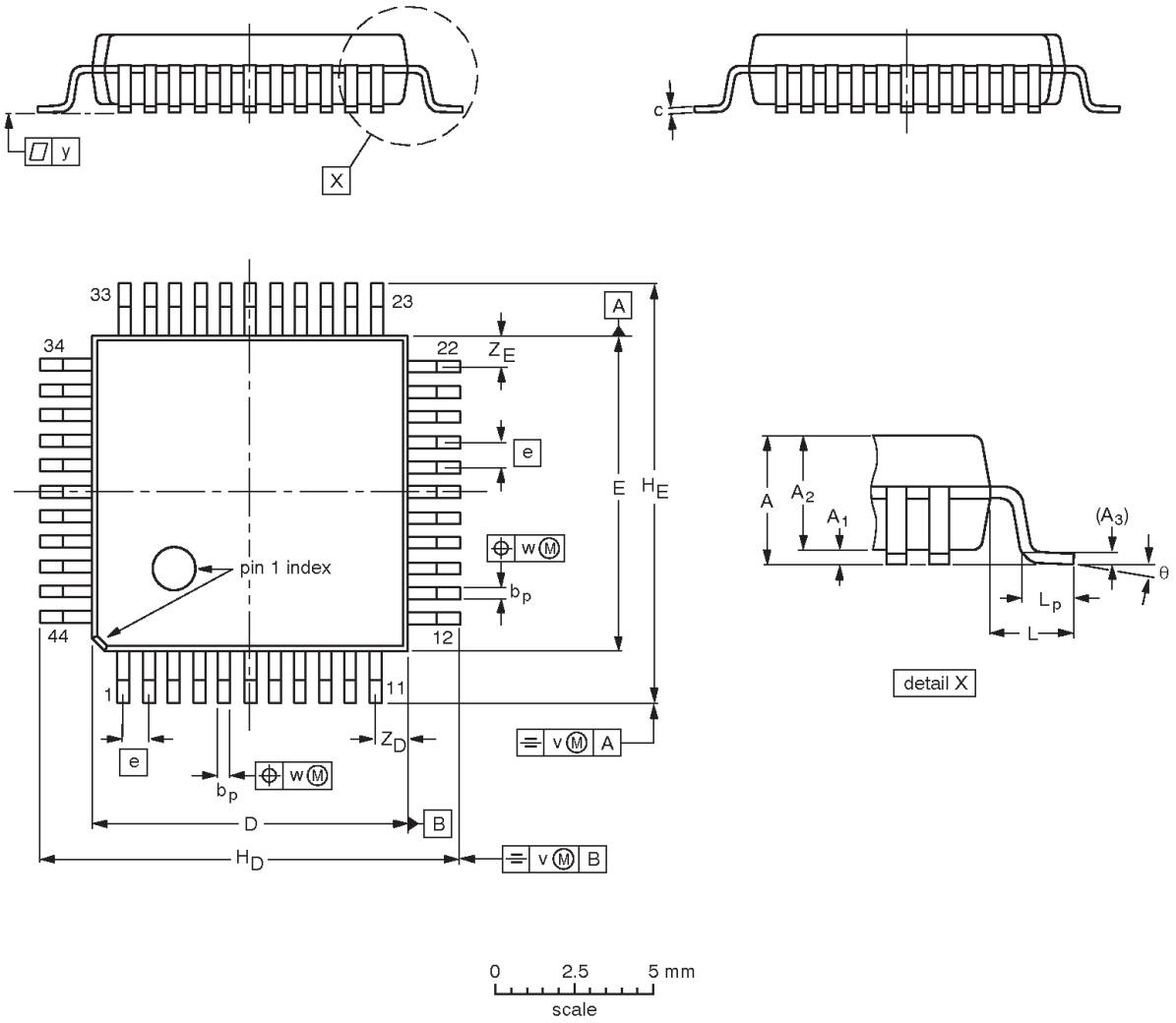


Figure 30. I_{DD} Test Condition, Power Down Mode
All other pins are disconnected. V_{DD}=2 V to 5.5 V

LQFP44: plastic low profile quad flat package; 44 leads; body 10 x 10 x 1.4 mm **SOT389-1**



DIMENSIONS (mm are the original dimensions)

UNIT	A max.	A_1	A_2	A_3	b_p	c	$D^{(1)}$	$E^{(1)}$	e	H_D	H_E	L	L_p	v	w	y	$Z_D^{(1)}$	$Z_E^{(1)}$	θ
mm	1.60	0.15 0.05	1.45 1.35	0.25	0.45 0.30	0.20 0.12	10.10 9.90	10.10 9.90	0.80	12.15 11.85	12.15 11.85	1.0	0.75 0.45	0.20	0.20	0.10	1.14 0.85	1.14 0.85	7° 0°

Note

1. Plastic or metal protrusions of 0.25 mm maximum per side are not included.

OUTLINE VERSION	REFERENCES				EUROPEAN PROJECTION	ISSUE DATE
	IEC	JEDEC	EIAJ			
SOT389-1	136E08	MS-026				~~99-12-17~~ 00-01-19

XA 16-bit microcontroller family
512 B RAM, watchdog, 2 UARTs

XA-G30

Data sheet status

Data sheet status [1]	Product status [2]	Definitions
Objective data	Development	This data sheet contains data from the objective specification for product development. Philips Semiconductors reserves the right to change the specification in any manner without notice.
Preliminary data	Qualification	This data sheet contains data from the preliminary specification. Supplementary data will be published at a later date. Philips Semiconductors reserves the right to change the specification without notice, in order to improve the design and supply the best possible product.
Product data	Production	This data sheet contains data from the product specification. Philips Semiconductors reserves the right to make changes at any time in order to improve the design, manufacturing and supply. Changes will be communicated according to the Customer Product/Process Change Notification (CPCN) procedure SNW-SQ-650A.

[1] Please consult the most recently issued data sheet before initiating or completing a design.

[2] The product status of the device(s) described in this data sheet may have changed since this data sheet was published. The latest information is available on the Internet at URL http://www.semiconductors.philips.com.

Definitions

Short-form specification — The data in a short-form specification is extracted from a full data sheet with the same type number and title. For detailed information see the relevant data sheet or data handbook.

Limiting values definition — Limiting values given are in accordance with the Absolute Maximum Rating System (IEC 60134). Stress above one or more of the limiting values may cause permanent damage to the device. These are stress ratings only and operation of the device at these or at any other conditions above those given in the Characteristics sections of the specification is not implied. Exposure to limiting values for extended periods may affect device reliability.

Application information — Applications that are described herein for any of these products are for illustrative purposes only. Philips Semiconductors make no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

Disclaimers

Life support — These products are not designed for use in life support appliances, devices or systems where malfunction of these products can reasonably be expected to result in personal injury. Philips Semiconductors customers using or selling these products for use in such applications do so at their own risk and agree to fully indemnify Philips Semiconductors for any damages resulting from such application.

Right to make changes — Philips Semiconductors reserves the right to make changes, without notice, in the products, including circuits, standard cells, and/or software, described or contained herein in order to improve design and/or performance. Philips Semiconductors assumes no responsibility or liability for the use of any of these products, conveys no license or title under any patent, copyright, or mask work right to these products, and makes no representations or warranties that these products are free from patent, copyright, or mask work right infringement, unless otherwise specified.

Contact information

For additional information please visit
http://www.semiconductors.philips.com. Fax: **+31 40 27 24825**

For sales offices addresses send e-mail to:
sales.addresses@www.semiconductors.philips.com.

© Koninklijke Philips Electronics N.V. 2002
All rights reserved. Printed in U.S.A.

Date of release: 03-02

Document order number: 9397 750 09576

Let's make things better.

Philips Semiconductors

PHILIPS