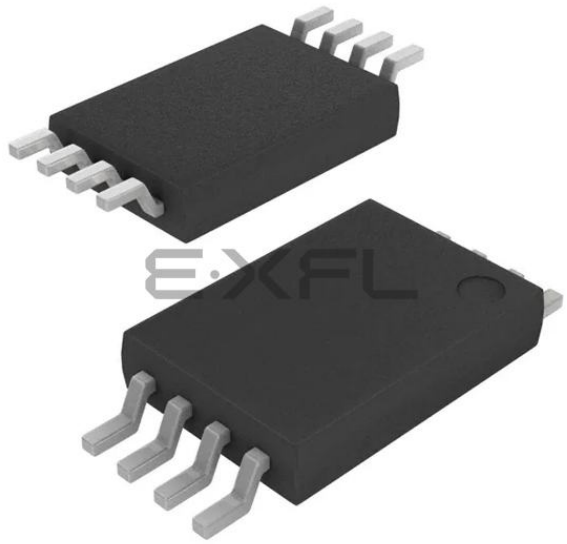


Welcome to E-XFL.COM

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	ACE1001
Core Size	8-Bit
Speed	4MHz
Connectivity	-
Peripherals	Brown-out Detect/Reset, LVD, POR, PWM, WDT
Number of I/O	6
Program Memory Size	1KB (1K x 8)
Program Memory Type	EEPROM
EEPROM Size	64 x 8
RAM Size	64 x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 5.5V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	8-TSSOP (0.173", 4.40mm Width)
Supplier Device Package	8-TSSOP
Purchase URL	https://www.e-xfl.com/product-detail/onsemi/ace1101bemt8x

Figure 2: ACE1101 Application Example (Remote Keyless Entry)

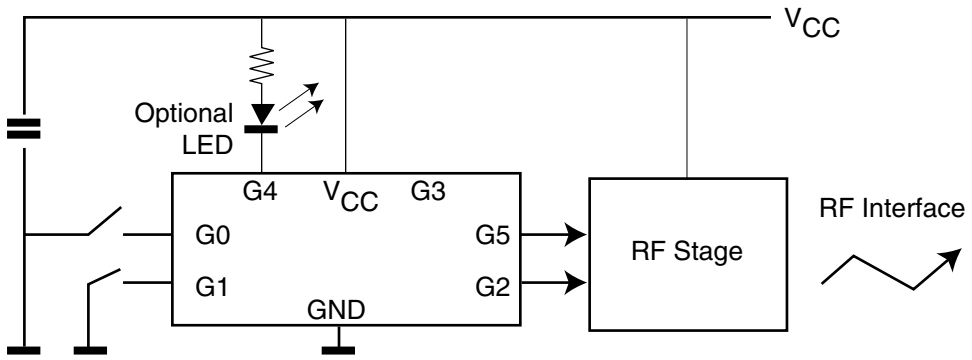
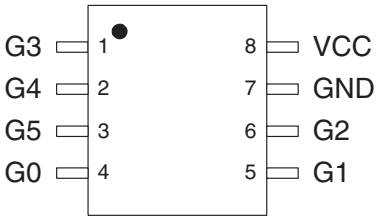


Figure 3: ACE1101 TSSOP/DIP 8-pin Device

a) Normal Mode Operation



b) Programming Mode Operation

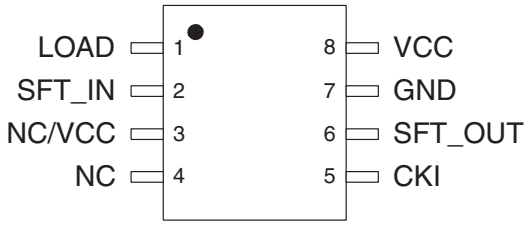
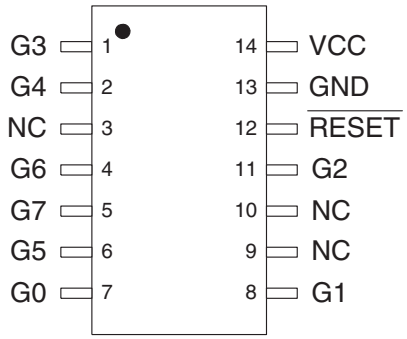


Figure 4: ACE1101 DIP 14-pin Device

a) Normal Mode Operation



b) Programming Mode Operation

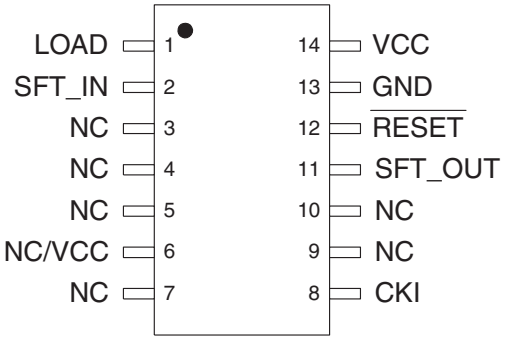
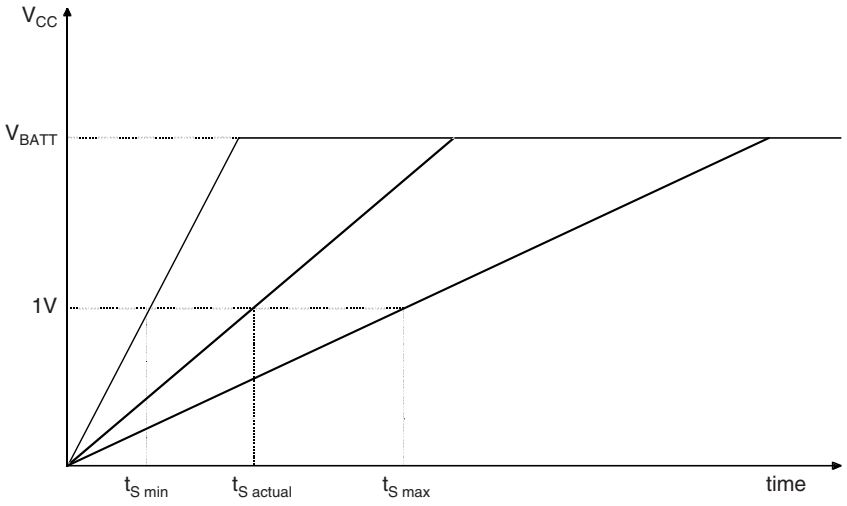
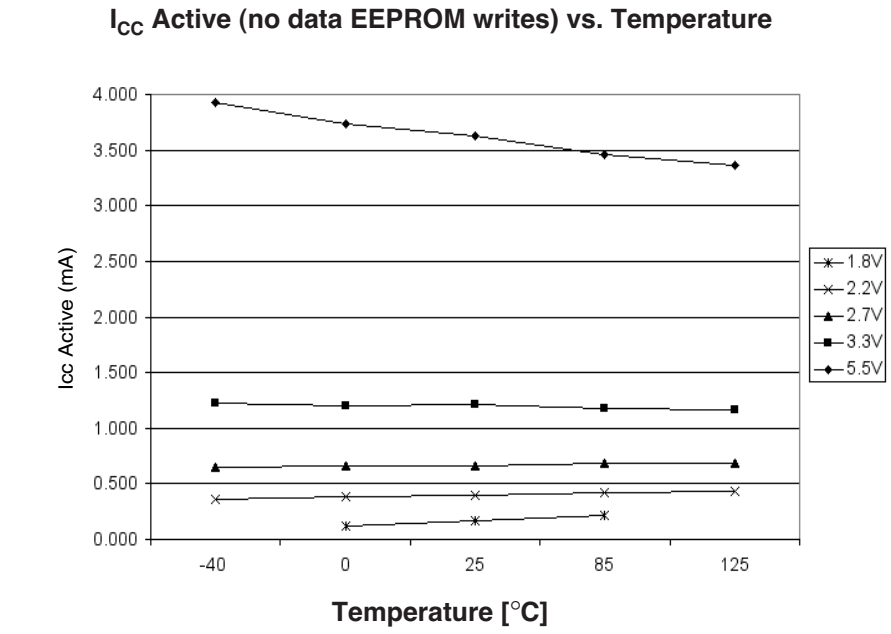


Figure 8: Power Supply Rise Time



Name	Parameter	Unit
V_{CC}	Supply Voltage	[V]
V_{BATT}	Battery Voltage (Nominal Operating Voltage)	[V]
$t_{S\ min}$	Minimum Time for V_{CC} to Rise by 1V	[ms]
$t_{S\ actual}$	Actual Time for V_{CC} to Rise by 1V	[ms]
$t_{S\ max}$	Maximum Time for V_{CC} to Rise by 1V	[ms]
S_{VCC}	Power Supply Slope	[ms/V]

Figure 9: I_{CC} Active



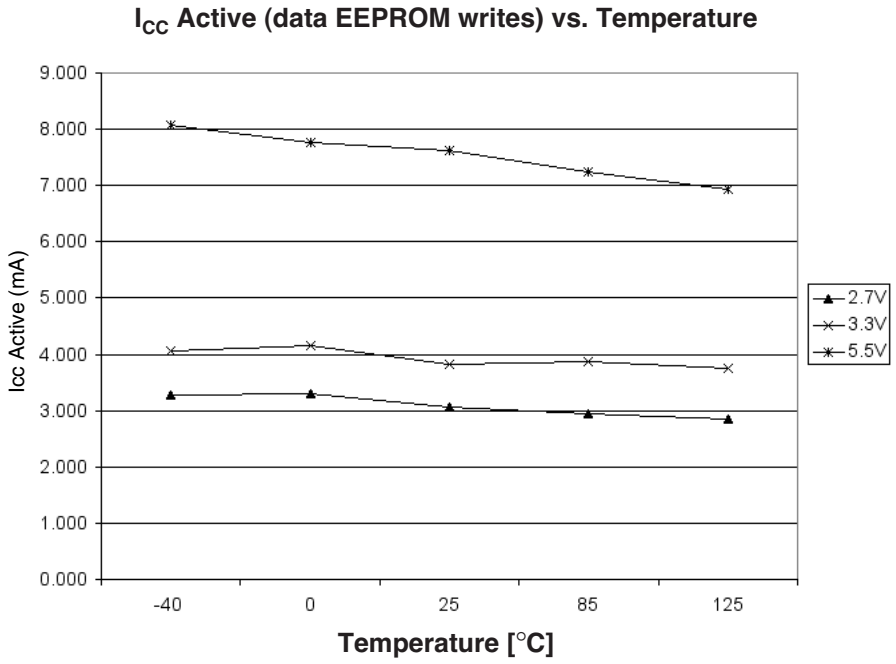
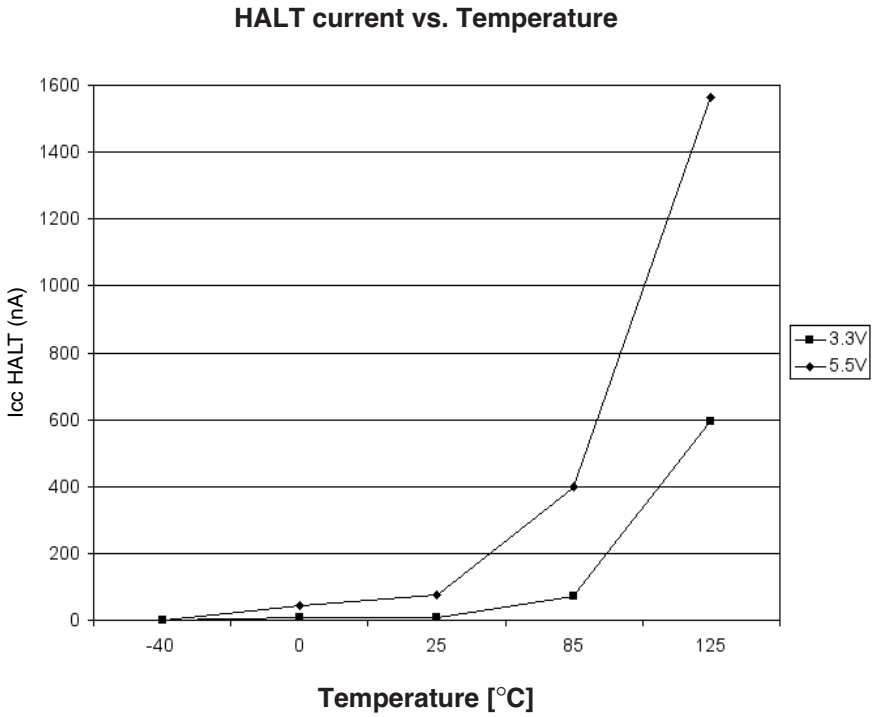


Figure 10: HALT Mode Currents



4.0 Arithmetic Controller Core

The ACEx microcontroller core is specifically designed for low cost applications involving bit manipulation, shifting block encryption. It is based on a modified Harvard architecture meaning peripheral, I/O, and RAM locations are addressed separately from instruction data.

The core differs from the traditional Harvard architecture by aligning the data and instruction memory sequentially. This allows the X-pointer (11-bits) to point to any memory location in either segment of the memory map. This modification improves the overall code efficiency of the ACEx microcontroller and takes advantage of the flexibility found on Von Neumann style machines.

4.1 CPU Registers

The ACEx microcontroller has five general-purpose registers. These registers are the Accumulator (A), X-Pointer (X), Program Counter (PC), Stack Pointer (SP), and Status Register (SR). The X, SP, and SR registers are all memory-mapped.

Figure 12: Programming Model

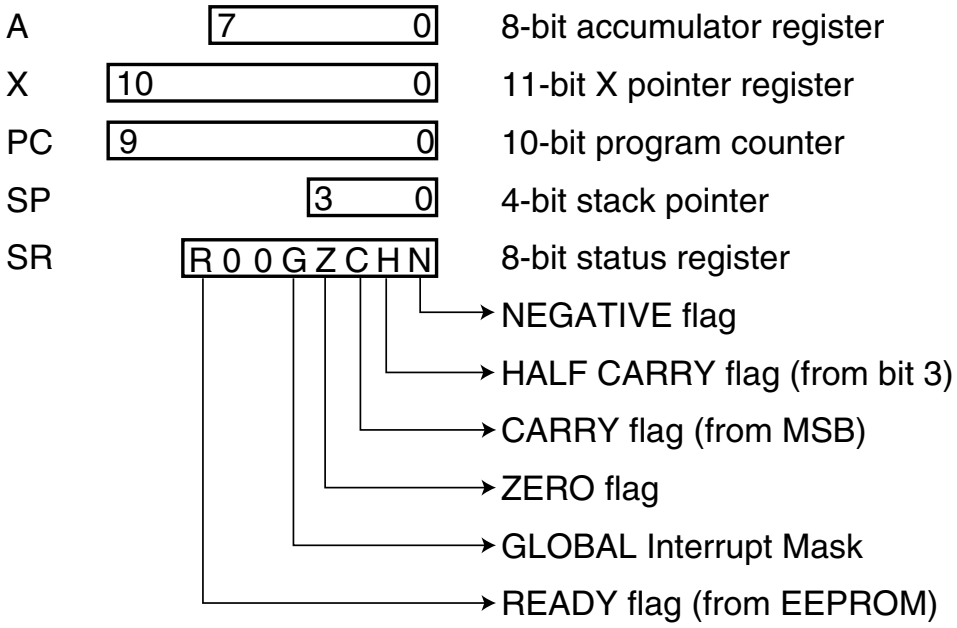


Table 9: Instruction Addressing Modes

Instruction	Immediate			Direct	Indexed	Indirect		Inherent	Relative	Absolute
ADC	A, #			A, M		A, [X]				
ADD	A, #			A, M		A, [X]				
AND	A, #			A, M		A, [X]				
OR	A, #			A, M		A, [X]				
SUBC	A, #			A, M		A, [X]				
XOR	A, #			A, M		A, [X]				
CLR				M		A	X			
INC				M		A	X			
DEC				M		A	X			
IFEQ	A, #	X, #	M,#	A, M		A, [X]				
IFGT	A, #	X, #		A, M		A, [X]				
IFNE	A, #			A, M		A, [X]				
IFLT		X, #								
SC								no-op		
RC								no-op		
IFC								no-op		
IFNC								no-op		
INVC								no-op		
LDC				#, M						
STC				#, M						
RLC				M				A		
RRC				M				A		
LD	A, #	X, #	M, #	A, M	A, [00,X]	A, [X]				
ST				A, M	A, [00,X]	A, [X]				
LD				M, M						
NOP								no-op		
IFBIT	#, A			#, M						
SBIT				#, M		#, [X]				
RBIT				#, M		#, [X]				
JP									Rel	
JSR										M
JMP										M
RET								no-op		
RETI								no-op		
INTR								no-op		

4.4 Memory Map

All I/O ports, peripheral registers and core registers, except the accumulator and the program counter are mapped into memory space.

Table 11: Memory Map

Address	Memory Space	Block	Contents
0x00 - 0x3F	Data	SRAM	Data RAM
0x40 - 0x7F	Data	EEPROM	Data EEPROM
0xAA	Data	Timer1	T1RALO register
0xAB	Data	Timer1	T1RAHI register
0xAC	Data	Timer1	TMR1LO register
0xAD	Data	Timer1	TMR1HI register
0xAE	Data	Timer1	T1CNTRL register
0xAF	Data	MIW	WKEDG register
0xB0	Data	MIW	WKPND register
0xB1	Data	MIW	WKEN register
0xB2	Data	I/O	PORTGD register
0xB3	Data	I/O	PORTGC register
0xB4	Data	I/O	PORTGP register
0xB5	Data	Timer0	WDSVR register
0xB6	Data	Timer0	T0CNTRL register
0xB7	Data	Clock	HALT mode register
0xB8 - 0xBC			Reserved
0xBD	Data	LBD	LBD register
0xBE	Data	Core	XHI register
0xBF	Data	Core	XLO register
0xC0	Data	Core	Power mode clear (PMC) register
0xCE	Data	Core	SP register
0xCF	Data	Core	Status register (SR)
0xC00 - 0xFF5	Program	EEPROM	Code EEPROM
0xFF6 - 0xFF7	Program	Core	Timer0 Interrupt vector
0xFF8 - 0xFF9	Program	Core	Timer1 Interrupt vector
0xFFA - 0xFFB	Program	Core	MIW Interrupt vector
0xFFC - 0xFFD	Program	Core	Software Interrupt vector
0xFFE - 0xFFF			Reserved

4.5 Memory

The ACEx microcontroller device has 64 bytes of SRAM and 64 bytes of EEPROM available for data storage. The device also has 1K bytes of EEPROM for program storage. Software can read and write to SRAM and data EEPROM but can only read from the code EEPROM. While in normal mode, the code EEPROM is protected from any writes. The code EEPROM can only be rewritten when the device is in program mode and if the write disable (WDIS) bit of the initialization register is not set to 1.

While in normal mode, the user can write to the data EEPROM array by 1) polling the ready (R) flag of the SR, then 2) executing the appropriate instruction. If the R flag is 1, the data EEPROM block is ready to perform the next write. If the R flag is 0, the data EEPROM is busy. The data EEPROM array will reset the R flag after the completion of a write cycle. Attempts to read, write, or enter HALT/IDLE mode while the data EEPROM is busy (R = 0) can affect the current data being written.

4.6 Initialization Registers

The ACEx microcontroller has two 8-bit wide initialization registers. These registers are read from the memory space on power-up to initialize certain on-chip peripherals. Figure 14 provides a detailed description of Initialization Register 1. The Initialization Register 2 is used to trim the internal oscillator to its appropriate frequency. This register is pre-programmed in the factory to yield an internal instruction clock of 1MHz.

Both Initialization Registers 1 and 2 can be read from and written to during programming mode. However, re-trimming the internal oscillator (writing to the Initialization Register 2) once it has left the factory is *discouraged*.

Figure 14: Initialization Register 1

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CMODE[0]	CMODE[1]	WDEN	BOREN	BLSEL [9]	UBD [7,8]	WDIS [7,8]	RDIS [7,8]

(0) RDIS [7,8] — If set, disables attempts to read the contents from the memory while in programming mode

(1) WDIS [7,8] — If set, disables attempts to write new contents to the memory while in programming mode

(2) UBD [7,8] — If set, the device will not allow any writes to occur in the upper block of data EEPROM (0x60-0x7F)

(3) BLSEL [9] — If set, the Brown-out Reset (BOR) voltage reference level is set to its higher range for the ACE1101
If not set, the BOR voltage reference level is set to its lower range for the ACE1101L

(4) BOREN — If set, allows a BOR to occur if V_{CC} falls below the voltage reference level

(5) WDEN — If set, enables the on-chip processor watchdog circuit

(6) CMODE[1] — Clock mode select bit 1 (See Table 17)

(7) CMODE[0] — Clock mode select bit 0 (See Table 17)

[7] If both the WDIS and RDIS bits are set, the device will no longer be able to be placed into program mode.

[8] If the RDIS or UBD bits are not set while the WDIS bit is not set, then the RDIS and UBD bits can be reset.

[9] The BLSEL bit is set to its appropriate level in the factory. If writing to the initialization register is necessary, be sure to maintain BLSEL set value.

5.0 Timer 1

Timer 1 is a versatile 16-bit timer that can operate in one of three modes:

- **Pulse Width Modulation** (PWM) mode, which generates pulses of a specified width and duty cycle
- **External Event Counter** mode, which counts occurrences of an external event
- **Standard Input Capture** mode, which measures the elapsed time between occurrences of external events

Timer 1 contains a 16-bit timer/counter register (TMR1), a 16-bit auto-reload/capture register (T1RA), and an 8-bit control register (T1CNTRL). All register are memory-mapped for simple access through the core with both the 16-bit registers organized as a pair of 8-bit register bytes {TMR1HI, TMR1LO} and {T1RAHI, T1RALO}. Depending on the operating mode, the timer contains an external input or output (T1) that is multiplexed with the I/O pin G2. By default, the TMR1 is reset to 0xFFFF, T1RA is reset to 0x0000, and T1CNTRL is reset to 0x00.

The timer can be started or stopped through the T1CNTRL register bit T1C0. When running, the timer counts down (decrements) every clock cycle. Depending on the operating mode, the timer's clock is either the instruction clock or a transition on the T1 input. In addition, occurrences of timer underflow (transitions from 0x0000 to 0xFFFF/T1RA value) can either generate an interrupt and/or toggle the T1 output pin.

Timer 1's interrupt (TMRI1) can be enabled by interrupt enable (T1EN) bit in the T1CNTRL register. When the timer interrupt is enabled, depending on the operating mode, the source of the interrupt is a timer underflow and/or a timer capture.

5.1 Timer control bits

Reading and writing to the T1CNTRL register controls the timer's operation. By writing to the control bits, the user can enable or disable the timer interrupts, set the mode of operation, and start or stop the timer. The T1CNTRL register bits are described in Tables 12 and 13.

Table 12: TIMER1 Control Register Definition (T1CNTRL)

T1CNTRL Register	Name	Function
Bit 7	T1C3	Timer TIMER1 control bit 3 (see Table 13)
Bit 6	T1C2	Timer TIMER1 control bit 2 (see Table 13)
Bit 5	T1C1	Timer TIMER1 control bit 1 (see Table 13)
Bit 4	T1C0	Timer TIMER1 run: 1 = Start timer, 0 = Stop timer; or Timer TIMER1 underflow interrupt pending flag in input capture mode
Bit 3	T1PND	Timer1 interrupt pending flag: 1 = Timer1 interrupt pending, 0 = Timer1 interrupt not pending
Bit 2	T1EN	Timer1 interrupt enable bit: 1 = Timer1 interrupt enabled, 0 = Timer1 interrupt disabled
Bit 1	----------	Reserved
Bit 0	----------	Reserved

Table 13: TIMER1 Operating Mode Selection

T1 C3	T1 C2	T1 C1	Timer Mode	Interrupt A Source	Timer Counts On
0	0	0	MODE 2	TIMER1 Underflow	T1 Pos. Edge
0	0	1	MODE 2	TIMER1 Underflow	T1 Neg. Edge
1	0	1	MODE 1 T1 Toggle	Autoreload T1RA	Instruction Clock
1	0	0	MODE 1 No T1 Toggle	Autoreload T1RA	Instruction Clock
0	1	0	MODE 3 Captures: T1 Pos. edge	Pos. T1 Edge	Instruction Clock
0	1	1	MODE 3 Captures: T1 Neg. Edge	Neg. T1 Edge	Instruction Clock

5.3 Mode 2: External Event Counter Mode

The External Event Counter mode operates similarly to the PWM mode; however, the timer is not clocked by the instruction clock but by transitions of the T1 input signal. The edge is selectable through the T1C1 bit of the T1CNTRL register. A block diagram of the timer's External Event Counter mode of operation is shown in Figure 16.

The T1 input should be connected to an external device that generates a positive/negative-going pulse for each event. By clocking the timer through T1, the number of positive/negative transitions can be counted therefore allowing software to capture the number of events that occur. The input signal on T1 must have a pulse width equal to or greater than one instruction clock cycle.

The counter can be configured to sense either positive-going or negative-going transitions on the T1 pin. The maximum frequency at which transitions can be sensed is one-half the frequency of the instruction clock.

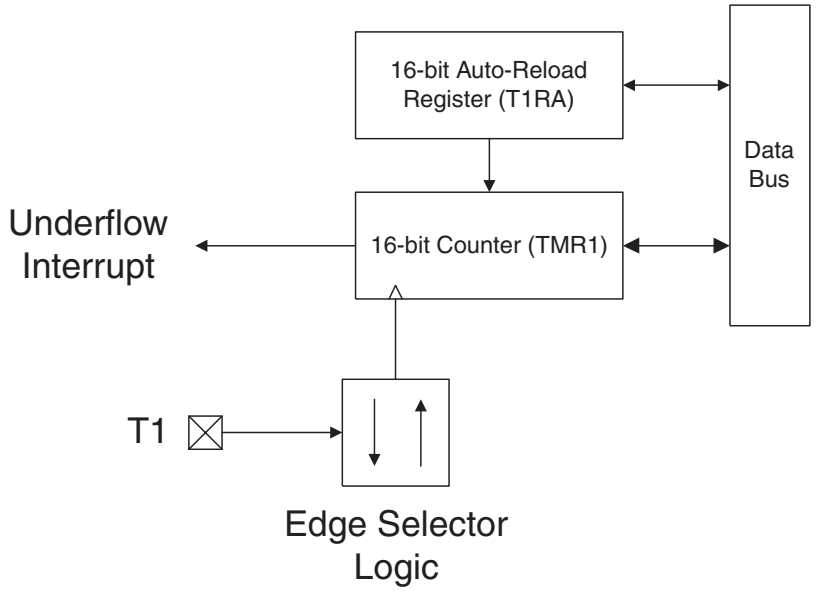
As with the PWM mode, when the counter underflows the counter is reloaded from the T1RA register and the count down proceedsfrom the loaded value. At every underflow, a pending flag (T1PND) located in the T1CNTRL register is set. Software must then clear the T1PND flag and can then load the T1RA register with an alternate value.

The counter has one interrupt (TMRI1) that is maskable through the T1EN bit of the T1CNTRL register. However, the core is only interrupted if the T1EN bit and the G (Global Interrupt enable) bit of the SR is set. If interrupts are enabled, the counter will generate an interrupt each time the T1PND flag is set (whenever timer underflows provided that the pending flag was cleared.) The interrupt service routine is responsible for proper handling of the T1PND flag and the T1EN bit.

The following steps show how to properly configure Timer 1 to operate in the External Event Counter mode. For this example, the counter is clocked every falling edge of the T1 input signal. Follow the instructions in parentheses to clock the counter every rising edge.

1. Configure T1 as an input by clearing bit 2 of PORTGC.
   - RBIT 2, PORTGC ; Configure G2 as an input
2. Initialize T1 to input with pull-up by setting bit 2 of PORTGD.
   - SBIT 2, PORTGD ; Set G2 high
3. Enable the global interrupt enable bit.
   - SBIT 4, STATUS
4. Load the initial count into the TMR1 and T1RA registers. When the number of external events is detected, the counter will reach zero; however, it will not underflow until the next event is detected. To count N pulses, load the value N-1 into the registers. If it is only necessary to count the number of occurrences and no action needs to be taken at a particular count, load the value 0xFFFF into the registers.
   - LD TMR1LO, #0FFH
   - LD TMR1HI, #00H
   - LD T1RALO, #0FFH
   - LD T1RAHI, #00H
5. Write the appropriate control value to the T1CNTRL register to select External Event Counter mode, to clock every falling edge, to set the enable bit, to clear the pending flag, and to start the counter. (See Table 12 and Table 13)
   - LD T1CNTRL, #34H (#00h) ; Setting the T1C0 bit starts the timer
6. When the counter underflows, the interrupt service routine must clear the T1PND flag and take whatever action is required once the number of events occurs. If the software wishes to merely count the number of events and the anticipated number may exceed 65,536, the interrupt service routine should record the number of underflows by incrementing a counter in memory. Software can then calculate the correct event count.
   - RBIT T1PND, T1CNTRL ; T1PND equals 3

Figure 16: External Event Counter Mode Block Diagram



6.0 Timer 0

Timer 0 is a 12-bit free running idle timer. Upon power-up or any reset, the timer is reset to 0x000 and then counts up continuously based on the instruction clock of 1MHz (1 μs). Software cannot read from or write to this timer. However, software can monitor the timer's pending (T0PND) bit that is set every 8192 cycles (initially 4096 cycles after a reset). The T0PND flag is set every other time the timer overflows (transitions from 0xFFF to 0x000) through a divide-by-2 circuit. After an overflow, the timer will reset and restart its counting sequence.

Software can either poll the T0PND bit or vector to an interrupt subroutine. In order to interrupt on a T0PND, software must be sure to enable the Timer 0 interrupt enable (T0INTEN) bit in the Timer 0 control (T0CNTRL) register and also make sure the G bit is set in SR. Once the timer interrupt is serviced, software should reset the T0PND bit before exiting the routine. Timer 0 supports the following functions:

1. Exiting from IDLE mode (See Section 16.0 for details.)
2. Start up delay from HALT mode
3. Watchdog pre-scaler (See Section 7.0 for details.)

The T0INTEN bit is a read/write bit. If set to 0, interrupt requests from the Timer 0 are ignored. If set to 1, interrupt requests are accepted. Upon reset, the T0INTEN bit is reset to 0.

The T0PND bit is a read/write bit. If set to 1, it indicates that a Timer 0 interrupt is pending. This bit is set by a Timer 0 overflow and is reset by software or system reset.

The WKINTEN bit is used in the Multi-input Wakeup/Interrupt block. See Section 8 for details.

7.0 Watchdog

The Watchdog timer is used to reset the device and safely recover in the rare event of a processor "runaway condition." The 12-bit Timer 0 is used as a pre-scaler for Watchdog timer. The Watchdog timer must be serviced before every 61,440 cycles but no sooner than 4096 cycles since the last Watchdog reset. The Watchdog is serviced through software by writing the value 0x1B to the Watchdog Service (WDSVR) register (see Figure 19). The part resets automatically if the Watchdog is serviced too frequent, or not frequent enough.

The Watchdog timer must be enabled through the Watchdog enable bit (WDEN) in the initialization register. The WDEN bit can only be set while the device is in programming mode. Once set, the Watchdog will always be powered-up enabled. Software cannot disable the Watchdog. The Watchdog timer can only be disabled in programming mode by resetting the WDEN bit as long as the memory write protect (WDIS) feature is not enabled.

WARNING

Ensure that the Watchdog timer has been serviced before entering IDLE mode because it remains operational during this time.

Figure 18: Timer 0 Control Register Definition (T0CNTRL)

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
WKINTEN	x	x	x	x	x	T0PND	T0INTEN

Figure 19: Watchdog Server Register (WDSVR)

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

8.0 Multi-Input Wakeup/interrupt Block

The Multi-Input Wakeup (MIW)/Interrupt contains three memory-mapped registers associated with this circuit: WKEDG (Wakeup Edge), WKEN (Wakeup Enable), and WKPND (Wakeup Pending). Each register has 8-bits with each bit corresponding to an input pins as shown in Figure 20. All three registers are initialized to zero upon reset.

The WKEDG register establishes the edge sensitivity for each of the wake-up input pin: either positive going-edge (0) or negative-going edge (1).

The WKEN register enables (1) or disables (0) each of the port pins for the Wakeup/Interrupt function. The wakeup I/Os used for the Wakeup/Interrupt function must also be configured as an input pin in its associated port configuration register. However, an interrupt of the core will not occur unless interrupts are enabled for the block via bit 7 of the T0CNTRL register (see Figure 18) and the G (global interrupt enable) bit of the SR is set.

The WKPND register contains the pending flags corresponding to each of the port pins (1 for wakeup/interrupt pending, 0 for wakeup/interrupt not pending).

To use the Multi-Input Wakeup/Interrupt circuit, perform the steps listed below. Performing the steps in the order shown will prevent false triggering of a Wakeup/Interrupt condition. This same procedure should be used following any type of reset because the wakeup inputs are left floating after resets resulting in unknown data on the port inputs.

1. Clear the WKEN register.
   - CLR WKEN
2. If necessary, write to the port configuration register to select the desired port pins to be configured as inputs.
   - RBIT 4, PORTGC ; G4
3. If necessary, write to the port data register to select the desired port pins input state.
   - SBIT 4, PORTGD ; Pull-up
4. Write the WKEDG register to select the desired type of edge sensitivity for each of the pins used.
   - LD WKEDG, #0FFH ; All negative-going edges
5. Clear the WKPND register to cancel any pending bits.
   - CLR WKPND
6. Set the WKEN bits associated with the pins to be used, thus enabling those pins for the Wakeup/Interrupt function.
   - LD WKEN, #10H ; Enabling G4

Once the Multi-Input Wakeup/Interrupt function has been configured, a transition sensed on any of the I/O pins will set the corresponding bit in the WKPND register. The WKPND bits , where the corresponding enable (WKEN) bits are set, will bring the device out of the HALT/IDLE mode and can also trigger an interrupt if interrupts are enabled. The interrupt service routine can read the WKPND register to determine which pin sensed the interrupt.

The interrupt service routine or other software should clear the pending bit. The device will not enter HALT/IDLE mode as long as a WKPND pending bit is pending and enabled. The user has the responsibility of clearing the pending flags before attempting to enter the HALT/IDLE mode.

Upon reset, the WKEDG register is configured to select positive-going edge sensitivity for all wakeup inputs. If the user wishes to change the edge sensitivity of a port pin, use the following procedure to avoid false triggering of a Wakeup/Interrupt condition.

1. Clear the WKEN bit associated with the pin to disable that pin.
2. Write the WKEDG register to select the new type of edge sensitivity for the pin.
3. Clear the WKPND bit associated with the pin.
4. Set the WKEN bit associated with the pin to re-enable it.

PORTG provides the user with three fully selectable, edge sensitive interrupts that are all vectored into the same service subroutine. The interrupt from PORTG shares logic with the wakeup circuitry. The WKEN register allows interrupts from PORTG to be individually enabled or disabled. The WKEDG register specifies the trigger condition to be either a positive or a negative edge. The WKPND register latches in the pending trigger conditions.

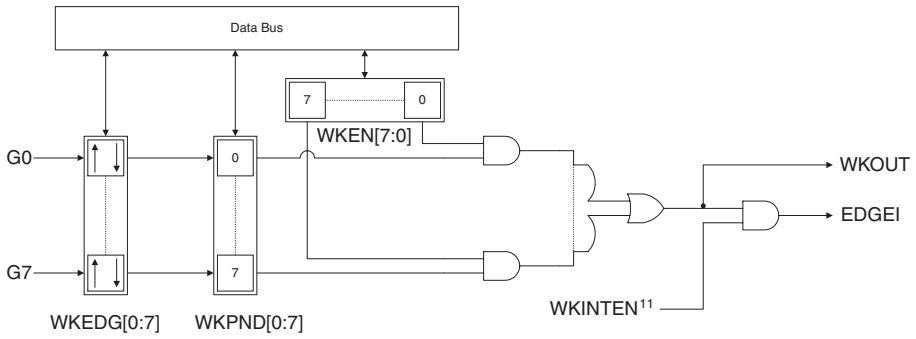
Since PORTG is also used for exiting the device from the HALT/IDLE mode, the user can elect to exit the HALT/IDLE mode either with or without the interrupt enabled. If the user elects to disable the interrupt, then the device restarts execution from the point at which it was stopped (first instruction cycle of the instruction following HALT/IDLE mode entrance instruction). In the other case, the device finishes the instruction that was being executed when the part was stopped and then branches to the interrupt service routine. The device then reverts to normal operation.

Figure 20: Multi-input Wakeup (MIW) Register Definition

WKEDG, WKEN, WKPND							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
[10]G7	[10]G6	G5	G4	G3	G2	G1	G0

[10] Available only on the 14-pin package option

Figure 21: Multi-input Wakeup (MIWU) Block Diagram



[11] WKINTEN: Bit 7 of T0CNTRL

9.0 I/O Port

The six I/O pins (eight on 14-pin package option) are bi-directional (see Figure 22) with the exception of G3 which is always an input with weak pull-up. The bi-directional I/O pins can be individually configured by software to operate as high-impedance inputs, as inputs with weak pull-up, or as push-pull outputs. The operating state is determined by the contents of the corresponding bits in the data and configuration registers. Each bi-directional I/O pin can be used for general purpose I/O, or in some cases, for a specific alternate function determined by the on-chip hardware.

9.1 I/O registers

The I/O pins (G0-G7) have three memory-mapped port registers associated with the I/O circuitry: a port configuration register (PORTGC), a port data register (PORTGD), and a port input register (PORTGP). PORTGC is used to configure the pins as inputs or outputs. A pin may be configured as an input by writing a 0 or as an output by writing a 1 to its corresponding PORTGC bit. If a pin is configured as an output, its PORTGD bit represents the state of the pin (1 = logic high, 0 = logic low). If the pin is configured as an input, its PORTGD bit selects whether the pin is a weak pull-up or a high-impedence input. Table 14 provides details of the port configuration options. The port configuration and data registers can both be read from or written to. Reading PORTGP returns the value of the port pins regardless of how the pins are configured. Since this device supports MIW, PORTG inputs have Schmitt triggers.

Figure 22: PORTG Logic Diagram

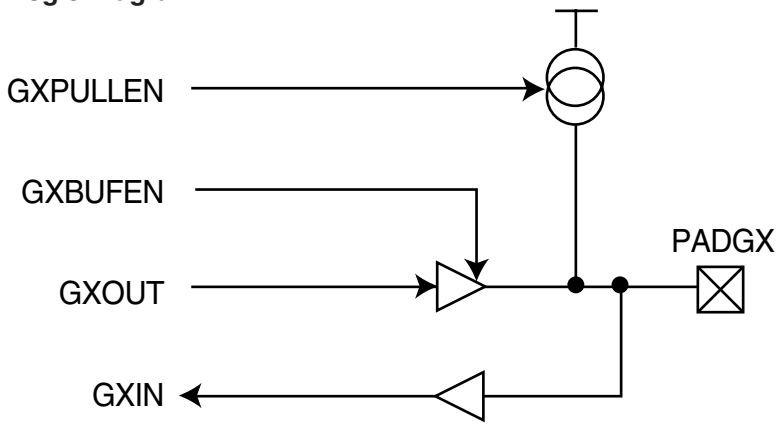


Figure 23: I/O Register bit assignments (PORTGC, PORTGD, PORTGD)

PORTGC, PORGD, PORTGD							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
G7[12]	G6[12]	G5	G4	G3[13]	G2	G1	G0

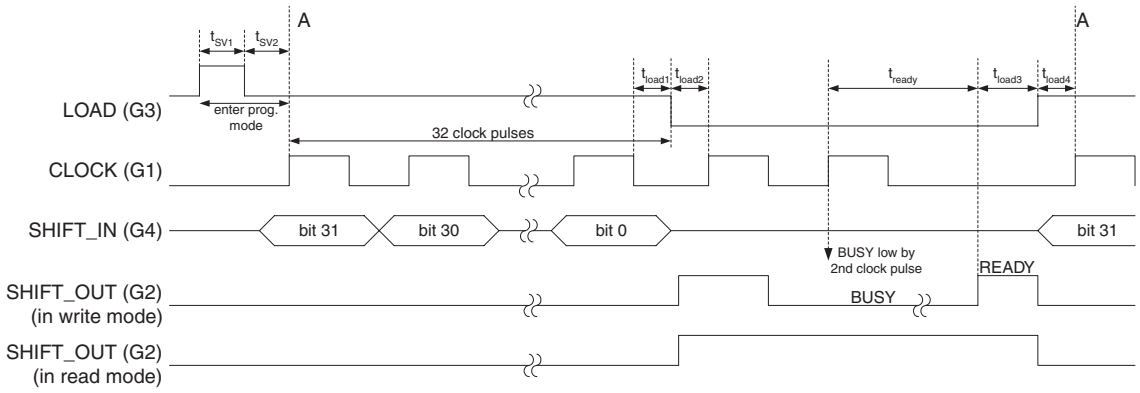
[12] Available only on the 14-pin package option

[13] G3 is always an input with weak pull-up.

Table 14: I/O configuration options

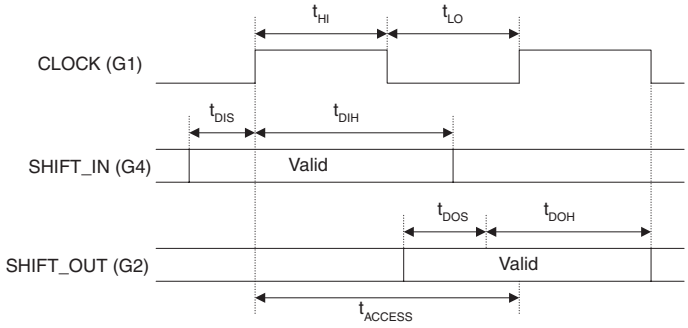
Configuration Bit	Data Bit	Port Pin Configuration
0	0	High-impedence input (TRI-STATE input)
0	1	Input with pull-up (weak one input)
1	0	Push-pull zero output
1	1	Push-pull one output

Figure 24- Programming Protocol[15]



A: start of programming cycle

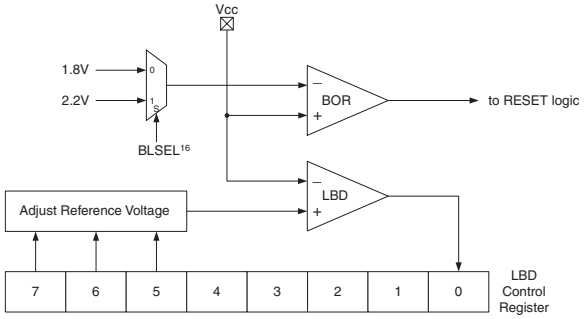
Figure 25- Serial Data Timing



11.0 Brown-out/Low Battery Detect Circuit

The Brown-out Reset (BOR) and Low Battery Detect (LBD) circuits on the ACEx microcontroller have been designed to offer two types of voltage reference comparators. The sections below will describe the functionality of both circuits.

Figure 26: BOR/LBD Block Diagram



11.1 Brown-out Reset [17]

The Brown-out Reset (BOR) function is used to hold the device in reset when V_{CC} drops below a fixed threshold. (See BOR Electrical Characteristics for threshold voltage.) While in reset, the device is held in its initial condition until V_{CC} rises above the threshold value. Shortly after V_{CC} rises above the threshold value, an internal reset sequence is started. After the reset sequence, the core fetches the first instruction and starts normal operation.

On the devices, the BOR should be used in situations when V_{CC} rises and falls slowly and in situations when V_{CC} does not fall to zero before rising back to operating range. The Brown-out Reset can be thought of as a supplement function to the Power-on Reset when V_{CC} does not fall below ~1.5V. The Power-on Reset circuit works best when V_{CC} starts from zero and rises sharply. So in applications where V_{CC} is not constant, the BOR will give added device stability.

The BOR circuit must be enabled through the BOR enable bit (BOREN) in the initialization register. The BOREN bit can only be set while the device is in programming mode. Once set, the BOR will always be powered-up enabled. Software cannot disable the BOR. The BOR can only be disabled in programming mode by resetting the BOREN bit as long as the global write protect (WDIS) feature is not enabled.

11.2 Low Battery Detect

The Low Battery Detect (LBD) circuit allows software to monitor the V_{CC} level at the lower voltage ranges. LBD has an eight level software programmable voltage reference threshold that can be changed on the fly. Once V_{CC} falls below the selected threshold, the LBD flag in the LBD control register is set. The LBD flag will hold its value until V_{CC} rises above the threshold. (See Table 16)

The LBD bit is read only. If LBD is 0, it indicates that the V_{CC} level is higher than the selected threshold. If LBD is 1, it indicates that the V_{CC} level is below the selected threshold. The threshold level can be adjusted up to eight levels using the three trim bits (Bat_trim[2:0]) of the LBD control register. The LBD flag does not cause any hardware actions or an interruption of the processor. It is for software monitoring only.

The LBD function is disabled during HALT/IDLE mode. After exiting HALT/IDLE, software must wait at lease 10 µs before reading the LBD bit to ensure that the internal circuit has stabilized.

[16] See Figure 14 for information on BLSEL.

[17] BOR is not available on the ACE1101B device.

Table 16: LBD Control Register Definition

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Bat_trim[2:0]			X	X	X	X	LBD

Level	Bat_trim[2]	Bat_trim[1]	Bat_trim[0]	Voltage Reference Range (±20%)
1	0	0	0	2.9 - 3.0
2	0	0	1	2.8 - 2.9
3	0	1	0	2.7 - 2.8
4	0	1	1	2.6 - 2.7
5	1	0	0	2.5 - 2.6
6	1	0	1	2.4 - 2.5
7	1	1	0	2.3 - 2.4
8	1	1	1	2.2 - 2.3

12.0 RESET block

When a RESET sequence is initiated, all I/O registers will be reset setting all I/Os to high-impedence inputs. The system clock is restarted after the required clock start-up delay. A reset is generated by any one of the following three conditions:

- Power-on Reset (as described in Section 13.0)
- Brown-out Reset (as described in Section 11.1)
- Watchdog Reset (as described in Section 7.0)
- External Reset[18] (as described in Section 13.0)

13.0 Power-On-Reset

The Power-On Reset (POR) circuit is guaranteed to work if the rate of rise of V_{CC} is no slower than 10ms/1volt. The POR circuit was designed to respond to fast low to high transitions between 0V and V_{CC}. The circuit will not work if V_{CC} does not drop to 0V before the next power-up sequence. In applications where 1) the V_{CC} rise is slower than 10ms/1 volt or 2) V_{CC} does not drop to 0v before the next power-up sequence the external reset option should be used.

The external reset provides a way to properly reset the ACEx microcontroller if POR cannot be used in the application. The external reset pin contains an internal pull-up resistor. Therefore, to reset the device the reset pin should be held low for at least 2ms so that the internal clock has enough time to stabilize.

14.0 CLOCK

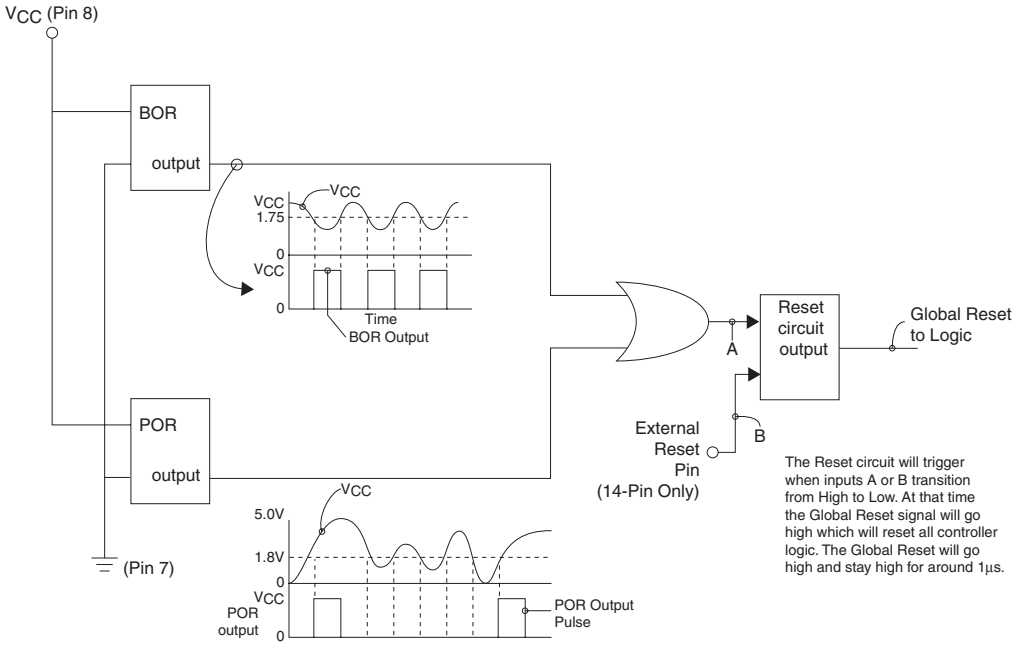
The ACEx microcontroller has an on-board oscillator trimmed to a frequency of 2MHz who is divided down by two yielding a 1MHz frequency. (See AC Electrical Characteristics.) Upon power-up, the on-chip oscillator runs continuously unless entering HALT mode or using an external clock source.

If required, an external oscillator circuit may be used depending on the states of the CMODE bits of the initialization register. (See Table 17) When the device is driven using an external clock, the clock input to the device (G1/CKI) can range between DC to 4MHz. For external crystal configuration, the output clock (CKO) is on the G0 pin. (See Figure 28) If an external crystal or RC is used, internally the input frequency (CKI) is divided-down by four to yield the corresponding instruction clock. If the device is configured for an external square clock, it will not be divided.

Table 17: CMODEx Bit Definition

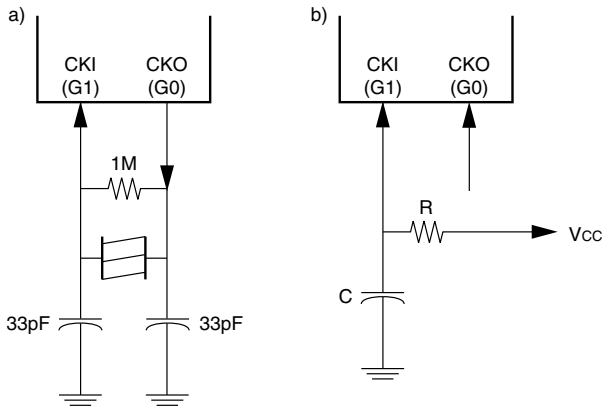
CMODE[1]	CMODE[0]	Clock Type
0	0	Internal 1 MHz clock
0	1	External square clock
1	0	External crystal/resonator
1	1	External RC clock

Figure 27: BOR and POR Circuit Relationship Diagram



[18] Available only on the 14-pin package option.

Figure 28: Crystal (a) and RC (b) Oscillator Diagrams



15.0 HALT Mode

The HALT mode is a power saving feature that almost completely shuts down the device for current conservation. The device is placed into HALT mode by setting the HALT enable bit (EHALT) of the HALT register through software using only the "LD M, #" instruction. EHALT is a write only bit and is automatically cleared upon exiting HALT. When entering HALT, the internal oscillator and all the on-chip systems including the LBD and the BOR circuits are shut down.

The device can exit HALT mode only by the MIW circuit. Therefore, prior to entering HALT mode, software must configure the MIW circuit accordingly. (See Section 8) After a wakeup from HALT, a 1ms start-up delay is initiated to allow the internal oscillator to stabilize before normal execution resumes. Immediately after exiting HALT, software must clear the Power Mode Clear (PMC) register by only using the "LD M, #" instruction. (See Figure 30)

16.0 IDLE Mode

In addition to the HALT mode power saving feature, the device also supports an IDLE mode operation. The device is placed into IDLE mode by setting the IDLE enable bit (EIDLE) of the HALT register through software using only the "LD M, #" instruction. EIDLE is a write only bit and is automatically cleared upon exiting IDLE. The IDLE mode operation is similar to HALT except the internal oscillator, the Watchdog, and the Timer 0 remain active while the other on-chip systems including the LBD and the BOR circuits are shut down.

The device can exit IDLE by a Timer 0 overflow every 8192 cycles or/and by the MIW circuit. If exiting IDLE mode with the MIW, prior to entering, software must configure the MIW circuit accordingly. (See Section 8) Once a wake from IDLE mode is triggered, the core will begin normal operation by the next clock cycle. Immediately after exiting IDLE mode, software must clear the Power Mode Clear (PMC) register by using only the "LD M, #" instruction. (See Figure 31)

Figure 29: HALT Register Definition

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
undefined	undefined	undefined	undefined	undefined	undefined	EIDLE	EHALT

Figure 30: Recommended HALT Flow

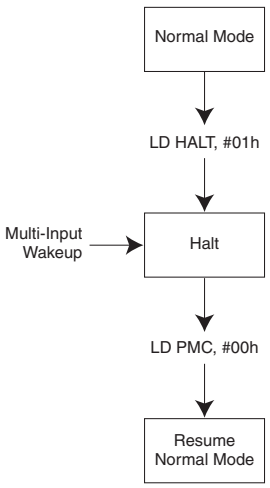
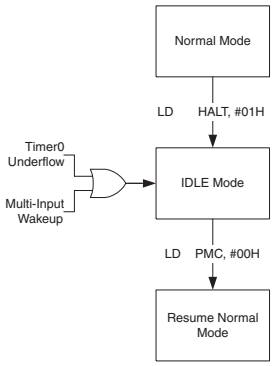
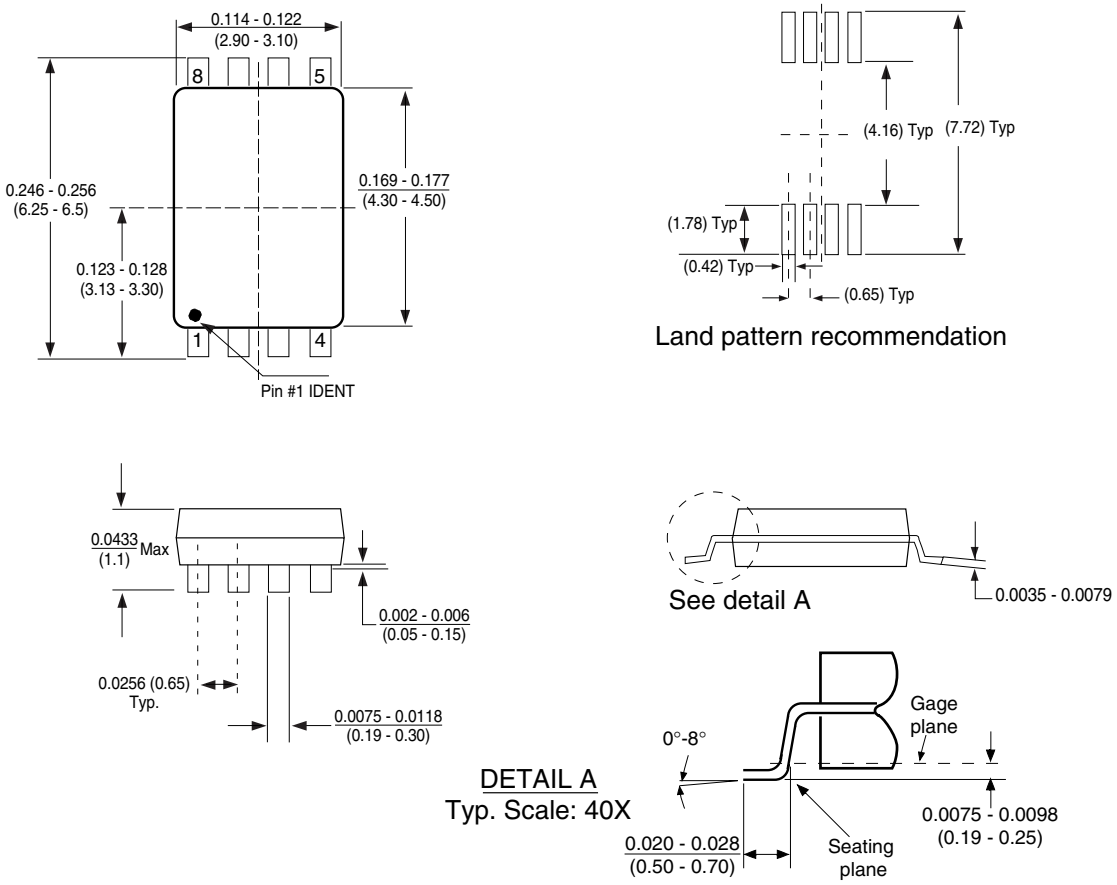


Figure 31: Recommended IDLE Flow



Physical Dimensions inches (millimeters) unless otherwise noted

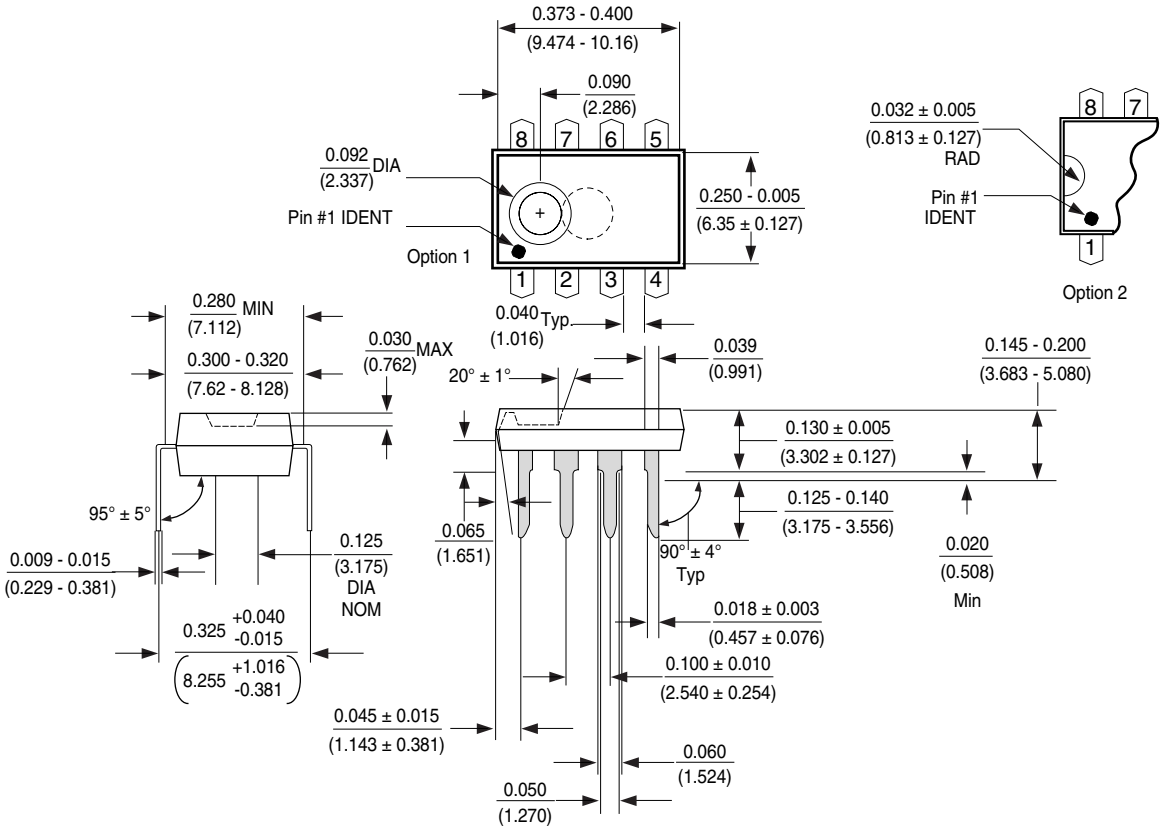


Notes: Unless otherwise specified

1. Reference JEDEC registration MO153. Variation AA. Dated 7/93

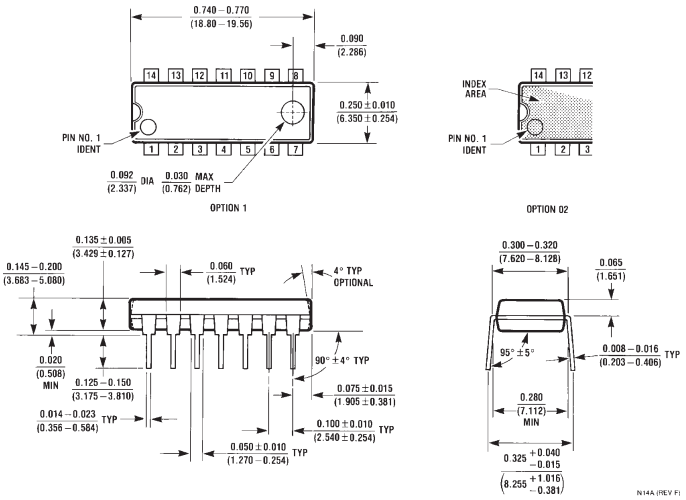
8-Pin Molded TSSOP (MT8)
Order Number ACE1101(1101L)MT8/ACE1101EMT8/ACE1101VMT8
ACE1101BMT8/ACE1101BEMT8/ACE1101BVMT8
Package Number MTC08

Physical Dimensions inches (millimeters) unless otherwise noted



8-Pin DIP (N)
Order Number ACE1101(1101L)N/ACE1101EN/ACE1101VN
ACE1101BN/ACE1101BEN/ACE1101BVN
Package Number N08E

Physical Dimensions inches (millimeters) unless otherwise noted



14-Pin DIP (N14)
Order Number ACE1101(1101L)N14/ACE1101EN14/ACE1101VN14
ACE1101BN14/ACE1101BEN14/ACE1101BVNT14
Package Number N014A