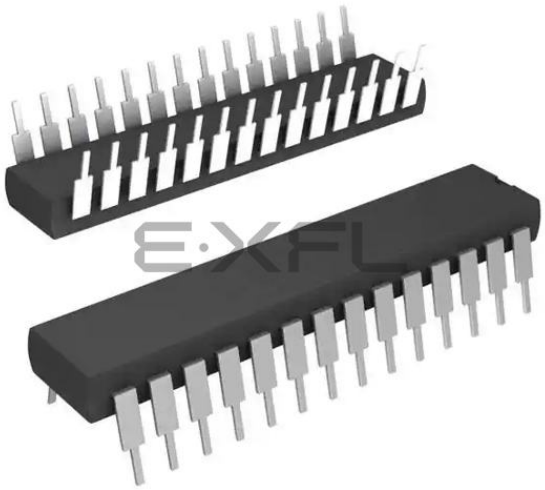

Welcome to E-XFL.COM

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	Active
Core Processor	eZ8
Core Size	8-Bit
Speed	20MHz
Connectivity	-
Peripherals	Brown-out Detect/Reset, LED, POR, PWM, WDT
Number of I/O	25
Program Memory Size	12KB (12K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	256 x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 3.6V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Through Hole
Package / Case	28-DIP (0.600", 15.24mm)
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/zilog/z8f1233pj020eg

PA0 and PA6 contain two different Timer functions, a timer input and a complementary timer output. Both of these functions require the same GPIO configuration, the selection between the two is based on the TIMER mode. For more details, see the Timers chapter on page 68.

Direct LED Drive

The Port C pins provide a sinked current output, capable of driving an LED without requiring an external resistor. The output sinks current at programmable levels, 3mA, 7mA, 13mA and 20mA. This mode is enabled through the LED Control registers.

For proper function, the LED anode must be connected to V_{DD} and the cathode to the GPIO pin.

Using all Port C pins in LED drive mode with maximum current may result in excessive total current. See the Electrical Characteristics chapter on page 184 for the maximum total current for the applicable package.

Shared Reset Pin

On the 20- and 28-pin devices, the Port D0 pin shares function with a bidirectional reset pin. Unlike all other I/O pins, this pin does not default to GPIO function on power-up. This pin acts as a bidirectional input/output open-drain reset with an internal pull-up until the user software reconfigures it as a GPIO PD0. When in GPIO mode, the Port D0 pin functions as output only, and must be configured as an output. PD0 supports the high drive feature, but not the stop-mode recovery feature.

Crystal Oscillator Override

For systems using a crystal oscillator, the pins PA0 and PA1 are connected to the crystal. When the crystal oscillator is enabled, the GPIO settings are overridden and PA0 and PA1 are disabled. See the Oscillator Control Register Definitions section on page 154.

5V Tolerance

In the 20- and 28-pin versions of this device, any pin, which shares functionality with an ADC, crystal or comparator port is not 5V-tolerant, including PA[1:0], PB[5:0] and PC[2:0]. All other signal pins are 5V-tolerant and can safely handle inputs higher than V_{DD} even with the pull-ups enabled, but with excess power consumption on pull-up resistor.

reload. For the timer output to make a state change at a ONE-SHOT time-out (rather than a single cycle pulse), first set the TPOL bit in the Timer Control Register to the start value before enabling ONE-SHOT Mode. After starting the timer, set TPOL to the opposite bit value.

Observe the following steps for configuring a timer for ONE-SHOT Mode and for initiating the count:

1. Write to the Timer Control Register to:
   - Disable the timer
   - Configure the timer for ONE-SHOT Mode
   - Set the prescale value
   - Set the initial output level (High or Low) if using the timer output Alternate function
2. Write to the Timer High and Low Byte registers to set the starting count value.
3. Write to the Timer Reload High and Low Byte registers to set the reload value.
4. If appropriate, enable the timer interrupt and set the timer interrupt priority by writing to the relevant interrupt registers.
5. If using the timer output function, configure the associated GPIO port pin for the timer output alternate function.
6. Write to the Timer Control Register to enable the timer and initiate counting.

In ONE-SHOT Mode, the system clock always provides the timer input. The timer period is calculated with the following equation:

$$\text{One-Shot Mode Time-Out Period (s)} = \frac{(\text{Reload Value} - \text{Start Value}) \times \text{Prescale}}{\text{System Clock Frequency (Hz)}}$$

CONTINUOUS Mode

In CONTINUOUS Mode, the timer counts up to the 16-bit reload value stored in the Timer Reload High and Low Byte registers. The timer input is the system clock. Upon reaching the reload value, the timer generates an interrupt, the count value in the Timer High and Low Byte registers is reset to 0001H and the counting resumes. Additionally, if the timer output alternate function is enabled, the timer output pin changes state (from Low to High or from High to Low) at timer reload.

Observe the following steps for configuring a timer for CONTINUOUS Mode and for initiating the count:

1. Write to the Timer Control Register to:

- Disable the timer
- Configure the timer for CONTINUOUS Mode
- Set the prescale value
- If using the timer output Alternate function, set the initial output level (High or Low)

2. Write to the Timer High and Low Byte registers to set the starting count value (usually 0001H). This action only affects the first pass in CONTINUOUS Mode. After the first timer reload in CONTINUOUS Mode, counting always begins at the reset value of 0001H.
3. Write to the Timer Reload High and Low Byte registers to set the reload value.
4. Enable the timer interrupt (if appropriate) and set the timer interrupt priority by writing to the relevant interrupt registers.
5. Configure the associated GPIO port pin (if using the timer output function) for the timer output alternate function.
6. Write to the Timer Control Register to enable the timer and initiate counting.

In CONTINUOUS Mode, the system clock always provides the timer input. The timer period is calculated with the following equation:

$$\text{Continuous Mode Time-Out Period (s)} = \frac{\text{Reload Value} \times \text{Prescale}}{\text{System Clock Frequency (Hz)}}$$

If an initial starting value other than 0001H is loaded into the Timer High and Low Byte registers, use the ONE-SHOT Mode equation to determine the first time-out period.

COUNTER Mode

In COUNTER Mode, the timer counts input transitions from a GPIO port pin. The timer input is taken from the GPIO port pin: timer input alternate function. The TPOL bit in the Timer Control Register determines whether the count occurs on the rising edge or the falling edge of the timer input signal. In COUNTER Mode, the prescaler is disabled.

Caution: The input frequency of the timer input signal must not exceed one-fourth the system clock frequency.

Upon reaching the reload value stored in the Timer Reload High and Low Byte registers, the timer generates an interrupt, the count value in the Timer High and Low Byte registers is reset to 0001H and counting resumes. Additionally, if the timer output alternate function

Timer 0–1 PWM High and Low Byte Registers

The Timer 0–1 PWM High and Low Byte (TxPWMH and TxPWML) registers, shown in Tables 54 and 55, control PWM operations. These registers also store the capture values for the CAPTURE and CAPTURE/COMPARE modes.

Table 54. Timer 0–1 PWM High Byte Register (TxPWMH)

Bit	7	6	5	4	3	2	1	0
Field	PWMH							
RESET	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Address	F04H, F0CH							

Table 55. Timer 0–1 PWM Low Byte Register (TxPWML)

Bit	7	6	5	4	3	2	1	0
Field	PWML							
RESET	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Address	F05H, F0DH							

Bit	Description
[7:0] PWMH, PWML	**Pulse Width Modulator High and Low Bytes** These two bytes, {PWMH[7:0], PWML[7:0]}, form a 16-bit value that is compared to the current 16-bit timer count. When a match occurs, the PWM output changes state. The PWM output value is set by the TPOL bit in the Timer Control Register (TxCTL1). The TxPWMH and TxPWML registers also store the 16-bit captured timer value when operating in capture or CAPTURE/COMPARE modes.

Flash Memory

The products in the Z8 Encore! F0830 Series features either 1KB (1024 bytes with NVDS), 2KB (2048 bytes with NVDS), 4KB (4096 bytes with NVDS), 8KB (8192 bytes with NVDS) or 12KB (12288 bytes with no NVDS) of nonvolatile Flash memory with read/write/erase capability. Flash memory can be programmed and erased in-circuit by either user code or through the On-Chip Debugger.

The Flash memory array is arranged in pages with 512 bytes per page. The 512-byte page is the minimum Flash block size that can be erased. Each page is divided into eight rows of 64 bytes.

For program/data protection, Flash memory is also divided into sectors. In the Z8 Encore! F0830 Series, each sector maps to one page (for 1KB, 2KB and 4KB devices), two pages (8KB device) or three pages (12KB device).

The first two bytes of Flash program memory is used as Flash option bits. For more information, see *the* Flash Option Bits chapter on page 124.

Table 69 lists the Flash memory configuration for each device in the Z8 Encore! F0830 Series. Figures 14 through 18 display the memory arrangements for each Flash memory size.

Table 69. Z8 Encore! F0830 Series Flash Memory Configuration

Part Number	Flash Size KB (Bytes)	Flash Pages	Program Memory Addresses	Flash Sector Size (bytes)
Z8F123x	12 (12,288)	24	0000H–2FFFH	1536
Z8F083x	8 (8196)	16	0000H–1FFFH	1024
Z8F043x	4 (4096)	8	0000H–0FFFH	512
Z8F023x	2 (2048)	4	0000H–07FFH	512
Z8F013x	1 (1024)	2	0000H–03FFH	512

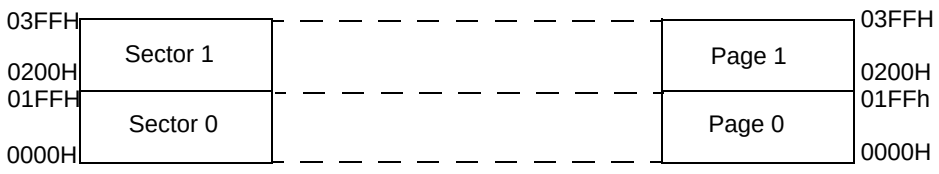

Figure 14. 1K Flash with NVDS

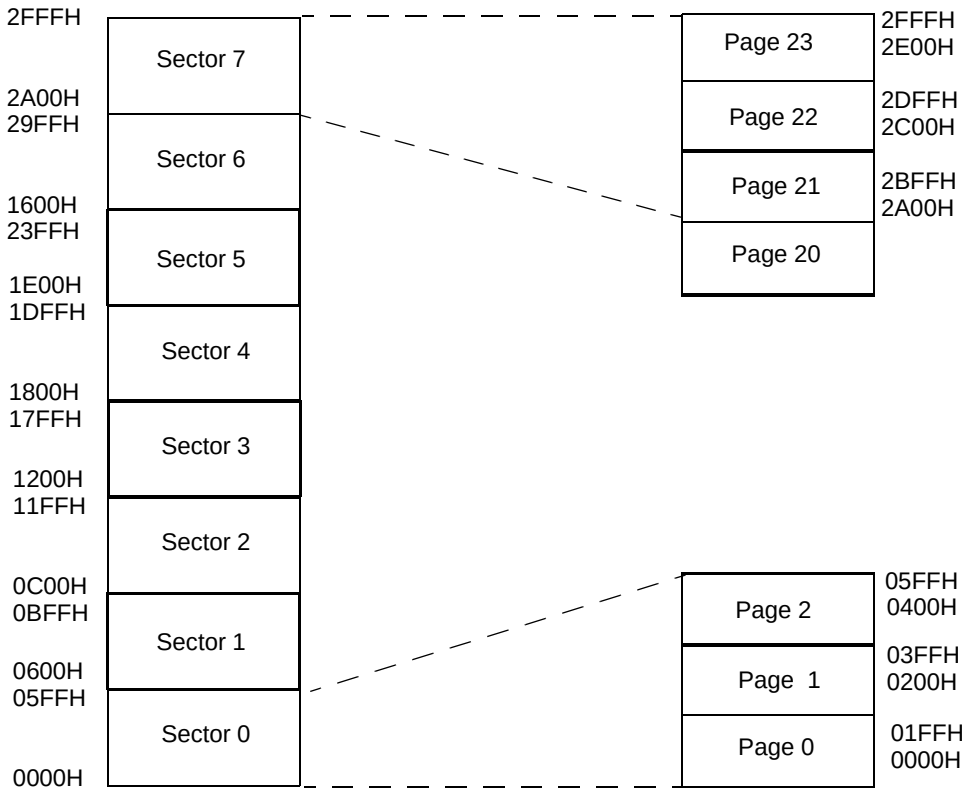

Figure 18. 12K Flash without NVDS

Data Memory Address Space

The Flash information area, including Zilog Flash option bits, are located in the data memory address space. The Z8 Encore! MCU is configured by these proprietary Flash option bits to prevent the user from writing to the eZ8 CPU data memory address space.

Flash Information Area

The Flash information area is physically separate from program memory and is mapped to the address range FE00H to FE7FH. Not all of these addresses are user-accessible. Factory trim values for the VBO, Internal Precision Oscillator and factory calibration data for the ADC are stored here.

Table 70 describes the Flash information area. This 128-byte information area is accessed by setting the bit 7 of the Flash Page Select Register to 1. When access is enabled, the

Byte Read

To read a byte from the NVDS array, user code must first push the address onto the stack. User code issues a `CALL` instruction to the address of the byte-read routine (`0x2000`). At the return from the subroutine, the read byte resides in working register R0 and the read status byte resides in working register R1. The bit fields of this status byte are defined in Table 92. Additionally, the user code should pop the address byte off the stack.

The read routine uses 16 bytes of stack space in addition to the one byte of address pushed by the user code. Sufficient memory must be available for this stack usage.

Due to the Flash memory architecture, NVDS reads exhibit a nonuniform execution time. A read operation takes between 71µs and 258µs (assuming a 20MHz system clock). Slower system clock speeds result in proportionally higher execution times.

NVDS byte reads from invalid addresses (those exceeding the NVDS array size) return `0xff`. Illegal read operations have a 6µs execution time.

The status byte returned by the NVDS read routine is zero for a successful read. If the status byte is nonzero, there is a corrupted value in the NVDS array at the location being read. In this case, the value returned in R0 is the byte most recently written to the array that does not have an error.

Table 92. Read Status Byte

Bit	7	6	5	4	3	2	1	0
Field	Reserved			DE	Reserved	FE	IGADDR	Reserved
Default Value	0	0	0	0	0	0	0	0

Bit	Description
[7:5]	**Reserved** These bits are reserved and must be programmed to 000.
[4] DE	**Data Error** When reading an NVDS address, if an error is found in the latest data corresponding to this NVDS address, this bit is set to 1. NVDS source code steps forward until it finds valid data at this address.
[3]	**Reserved** This bit is reserved and must be programmed to 0.
[2] FE	**Flash Error** If a Flash error is detected, this bit is set to 1.
[1] IGADDR	**Illegal Address** When NVDS byte reads from invalid addresses (those exceeding the NVDS array size) occur, this bit is set to 1.
[0]	**Reserved** This bit is reserved and must be programmed to 0.

If the OCD receives a serial break (nine or more continuous bits low), the autobaud detector/generator resets. Reconfigure the autobaud detector/generator by sending 80H.

OCD Serial Errors

The OCD can detect any of the following error conditions on the DBG pin:

- Serial break (a minimum of nine continuous bits Low)
- Framing error (received Stop bit is Low)
- Transmit collision (simultaneous transmission by OCD and host detected by the OCD)

When the OCD detects one of these errors, it aborts any command currently in progress, transmits a four character long serial break back to the host and resets the autobaud detector/generator. A framing error or transmit collision may be caused by the host sending a serial break to the OCD. As a result of the open-drain nature of the interface, returning a serial break back to the host only extends the length of the serial break if the host releases the serial break early.

The host transmits a serial break on the DBG pin when first connecting to the Z8 Encore! F0830 Series devices or when recovering from an error. A serial break from the host resets the autobaud generator/detector, but does not reset the OCD Control Register. A serial break leaves the device in DEBUG Mode, if that is the current mode. The OCD is held in reset until the end of the serial break when the DBG pin returns high. Because of the open-drain nature of the DBG pin, the host can send a serial break to the OCD even if the OCD is transmitting a character.

Breakpoints

Execution breakpoints are generated using the BRK instruction (opcode 00H). When the eZ8 CPU decodes a BRK instruction, it signals the OCD. If breakpoints are enabled, the OCD enters DEBUG Mode and idles the eZ8 CPU. If breakpoints are not enabled, the OCD ignores the BRK signal and the BRK instruction operates as an NOP instruction.

Breakpoints in Flash Memory

The BRK instruction is opcode 00H, which corresponds to the fully programmed state of a byte in Flash memory. To implement a breakpoint, write 00H to the required break address overwriting the current instruction. To remove a breakpoint, the corresponding page of Flash memory must be erased and reprogrammed with the original data.

Runtime Counter

The OCD contains a 16-bit runtime counter. It counts system clock cycles between breakpoints. The counter starts counting when the OCD leaves DEBUG Mode and stops counting when it enters DEBUG Mode again or when it reaches the maximum count of FFFFH.

On-Chip Debugger Commands

The host communicates to the On-Chip Debugger by sending OCD commands using the DBG interface. During normal operation, only a subset of the OCD commands are available. In DEBUG Mode, all OCD commands become available unless the user code and control registers are protected by programming the Flash read protect option bit (FRP). The FRP prevents the code in memory from being read out of the Z8 Encore! F0830 Series products. When this option is enabled, several of the OCD commands are disabled.

Table 95 summarizes the On-Chip Debugger commands. This table indicates the commands that operate when the device is not in DEBUG Mode (normal operation) and the commands that are disabled by programming the FRP.

Table 95. On-Chip Debugger Command Summary

Debug Command	Command Byte	Enabled when not in DEBUG Mode?	Disabled by Flash Read Protect Option Bit
Read OCD Revision	00H	Yes	–
Reserved	01H	–	–
Read OCD Status Register	02H	Yes	–
Read Runtime Counter	03H	–	–
Write OCD Control Register	04H	Yes	Cannot clear DBGMODE bit
Read OCD Control Register	05H	Yes	–
Write Program Counter	06H	–	Disabled
Read Program Counter	07H	–	Disabled
Write Register	08H	–	Only writes of the Flash Memory Control registers are allowed. Additionally, only the Mass Erase command is allowed to be written to the Flash Control register.
Read Register	09H	–	Disabled
Write Program Memory	0AH	–	Disabled
Read Program Memory	0BH	–	Disabled
Write Data Memory	0CH	–	Yes
Read Data Memory	0DH	–	–

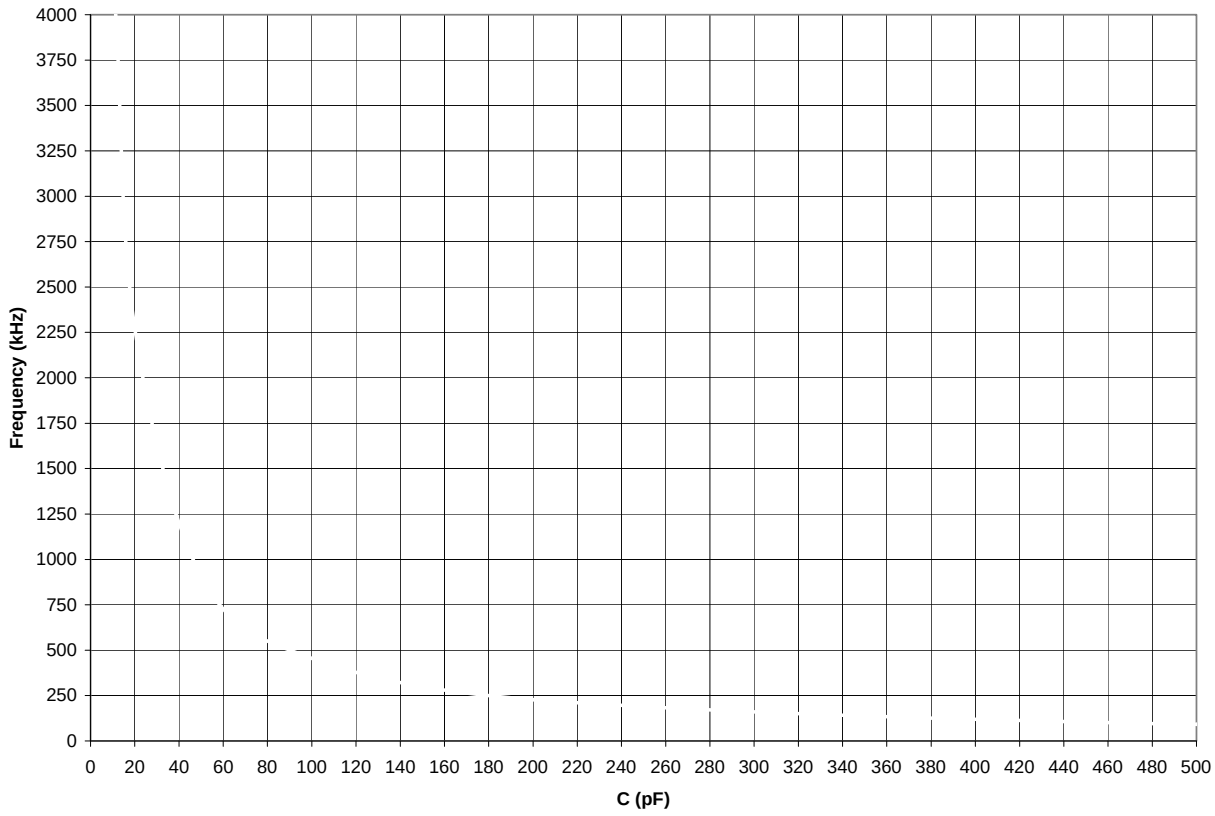

Figure 27. Typical RC Oscillator Frequency as a Function of External Capacitance with a 45 kΩ Resistor

Caution: When using the external RC OSCILLATOR Mode, the oscillator can stop oscillating if the power supply drops below 2.7V but before it drops to the Voltage Brown-Out threshold. The oscillator resumes oscillation when the supply voltage exceeds 2.7V.

This example indicates that the source data is added to the destination data; the result is stored in the destination location.

eZ8 CPU Instruction Classes

eZ8 CPU instructions can be divided functionally into the following groups:

- Arithmetic
- Bit manipulation
- Block transfer
- CPU control
- Load
- Logical
- Program control
- Rotate and shift

Tables 105 through 112 contain the instructions belonging to each group and the number of operands required for each instruction. Some instructions appear in more than one table as these instructions can be considered as a subset of more than one category. Within these tables, the source operand is identified as *src*, the destination operand is *dst* and a condition code is *cc*.

Table 105. Arithmetic Instructions

Mnemonic	Operands	Instruction
ADC	dst, src	Add with Carry
ADCX	dst, src	Add with Carry using Extended Addressing
ADD	dst, src	Add
ADDX	dst, src	Add using Extended Addressing
CP	dst, src	Compare
CPC	dst, src	Compare with Carry
CPCX	dst, src	Compare with Carry using Extended Addressing
CPX	dst, src	Compare using Extended Addressing
DA	dst	Decimal Adjust
DEC	dst	Decrement
DECW	dst	Decrement Word
INC	dst	Increment

eZ8 CPU Instruction Summary

Table 113 summarizes the eZ8 CPU instructions. The table identifies the addressing modes employed by the instruction, the effect upon the Flags register, the number of CPU clock cycles required for the instruction fetch and the number of CPU clock cycles required for the instruction execution.

Table 113. eZ8 CPU Instruction Summary

Assembly Mnemonic	Symbolic Operation	Address Mode dst	Address Mode src	Op Code(s) (Hex)	Flags C	Z	S	V	D	H	Fetch Cycles	Instr. Cycles
ADC dst, src	dst ← dst + src + C	r	r	12	*	*	*	*	0	*	2	3
		r	Ir	13							2	4
		R	R	14							3	3
		R	IR	15							3	4
		R	IM	16							3	3
		IR	IM	17							3	4
ADCX dst, src	dst ← dst + src + C	ER	ER	18	*	*	*	*	0	*	4	3
		ER	IM	19							4	3
ADD dst, src	dst ← dst + src	r	r	02	*	*	*	*	0	*	2	3
		r	Ir	03							2	4
		R	R	04							3	3
		R	IR	05							3	4
		R	IM	06							3	3
		IR	IM	07							3	4
ADDX dst, src	dst ← dst + src	ER	ER	08	*	*	*	*	0	*	4	3
		ER	IM	09							4	3

Note: Flags Notation:
* = Value is a function of the result of the operation.
– = Unaffected.
X = Undefined.
0 = Reset to 0.
1 = Set to 1.

Op Code Maps

A description of the opcode map data and the abbreviations are provided in Figure 28. Table 114 on page 181 lists opcode map abbreviations.

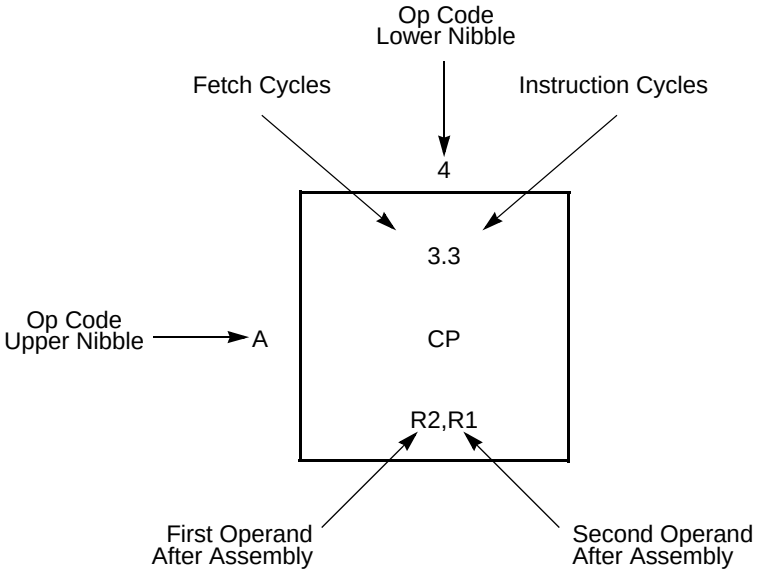

Figure 28. Op Code Map Cell Description

Table 117. AC Characteristics (Continued)

		V_{DD} = 2.7 to 3.6V T_A = 0°C to +70°C		V_{DD} = 2.7 to 3.6V T_A = –40°C to +105°C			
Symbol	**Parameter**	**Min**	**Max**	**Min**	**Max**	**Units**	**Conditions**
T_{XINR}	System Clock Rise Time			–	3	ns	T_{CLK} = 50 ns
T_{XINF}	System Clock Fall Time			–	3	ns	T_{CLK} = 50 ns
$T_{XTALSET}$	Crystal Oscillator Setup Time			–	30,000	cycle	Crystal oscillator cycles
T_{IPOSET}	Internal Precision Oscillator Startup Time			–	25	µs	Startup time after enable
T_{WDTSET}	WDT Startup Time			–	50	µs	Startup time after reset

On-Chip Peripheral AC and DC Electrical Characteristics

Table 118. Power-On Reset and Voltage Brown-Out Electrical Characteristics and Timing

		T_A = 0°C to +70°C			T_A = –40°C to +105°C				
Symbol	**Parameter**	**Min**	**Typ**	**Max**	**Min**	**Typ[1]**	**Max**	**Units**	**Conditions**
V_{POR}	Power-On Reset Voltage Threshold				2.20	2.45	2.70	V	$V_{DD} = V_{POR}$ (default VBO trim)
V_{VBO}	Voltage Brown-Out Reset Voltage Threshold				2.15	2.40	2.65	V	$V_{DD} = V_{VBO}$ (default VBO trim)
	V_{POR} to V_{VBO} hysteresis					50	75	mV	
	Starting V_{DD} voltage to ensure valid Power-On Reset.				–	V_{SS}	–	V	
T_{ANA}	Power-On Reset Analog Delay				–	50	–	µs	$V_{DD} > V_{POR}$; T_{POR} Digital Reset delay follows T_{ANA}

Note: [1]Data in the typical column is from characterization at 3.3V and 0°C. These values are provided for design guidance only and are not tested in production.

Hex Address: F0D

Table 143. Timer 1 PWM Low Byte Register (T1PWML)

Bit	7	6	5	4	3	2	1	0
Field	PWML							
RESET	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Address	F0DH							

Hex Address: F0E

Table 144. Timer 1 Control Register 0 (T1CTL0)

Bit	7	6	5	4	3	2	1	0
Field	TMODEHI	TICONFIG		Reserved	PWMD			INPCAP
RESET	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Address	F0EH							

Hex Address: F0F

Table 145. Timer 1 Control Register 1 (T1CTL1)

Bit	7	6	5	4	3	2	1	0
Field	TEN	TPOL	PRES			TMODE		
RESET	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Address	F0FH							

Hex Addresses: F10–F6F

This address range is reserved.

Hex Address: FC1

Table 158. IRQ0 Enable High Bit Register (IRQ0ENH)

Bit	7	6	5	4	3	2	1	0
Field	Reserved	T1ENH	T0ENH	Reserved	Reserved	Reserved	Reserved	ADCENH
RESET	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Address	FC1H							

Hex Address: FC2

Table 159. IRQ0 Enable Low Bit Register (IRQ0ENL)

Bit	7	6	5	4	3	2	1	0
Field	Reserved	T1ENL	T0ENL	Reserved	Reserved	Reserved	Reserved	ADCENL
RESET	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R	R	R/W
Address	FC2H							

Hex Address: FC3

Table 160. Interrupt Request 1 Register (IRQ1)

Bit	7	6	5	4	3	2	1	0
Field	PA7I	PA6CI	PA5I	PA4I	PA3I	PA2I	PA1I	PA0I
RESET	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Address	FC3H							

Hex Address: FC4

Table 161. IRQ1 Enable High Bit Register (IRQ1ENH)

Bit	7	6	5	4	3	2	1	0
Field	PA7ENH	PA6CENH	PA5ENH	PA4ENH	PA3ENH	PA2ENH	PA1ENH	PA0ENH
RESET	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Address	FC4H							

G

H

I

Customer Support

To share comments, get your technical questions answered or report issues you may be experiencing with our products, please visit Zilog's Technical Support page at http://support.zilog.com.

To learn more about this product, find additional documentation or to discover other facets about Zilog product offerings, please visit the Zilog Knowledge Base at http://zilog.com/kb or consider participating in the Zilog Forum at http://zilog.com/forum.

This publication is subject to replacement by a later edition. To determine whether a later edition exists, please visit the Zilog website at http://www.zilog.com.