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	1KB (1K 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	Surface Mount
Package / Case	28-SOIC (0.295", 7.50mm Width)
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/zilog/z8f0131sj020eg

Pin Description

The Z8 Encore! F0830 Series products are available in a variety of package styles and pin configurations. This chapter describes the signals and the pin configurations for each of the package styles. For information about the physical package specifications, see the Packaging chapter on page 199.

Available Packages

Table 3 lists the package styles that are available for each device in the Z8 Encore! F0830 Series product line.

Table 3. Z8 Encore! F0830 Series Package Options

Part Number	ADC	20-pin QFN	20-pin SOIC	20-pin SSOP	20-pin PDIP	28-pin QFN	28-pin SOIC	28-pin SSOP	28-pin PDIP
Z8F1232	Yes	X	X	X	X	X	X	X	X
Z8F1233	No	X	X	X	X	X	X	X	X
Z8F0830	Yes	X	X	X	X	X	X	X	X
Z8F0831	No	X	X	X	X	X	X	X	X
Z8F0430	Yes	X	X	X	X	X	X	X	X
Z8F0431	No	X	X	X	X	X	X	X	X
Z8F0230	Yes	X	X	X	X	X	X	X	X
Z8F0231	No	X	X	X	X	X	X	X	X
Z8F0130	Yes	X	X	X	X	X	X	X	X
Z8F0131	No	X	X	X	X	X	X	X	X

Pin Configurations

Figures 2 and 3 display the pin configurations of all of the packages available in the Z8 Encore! F0830 Series. See Table 4 on page 11 for a description of the signals. Analog input alternate functions (ANAx) are not available on the following devices:

- Z8F0831
- Z8F0431
- Z8F0131
- Z8F0231
- Z8F1233

Address Space

The eZ8 CPU can access the following three distinct address spaces:

- The register file addresses access for the general purpose registers and the eZ8 CPU, peripheral and general purpose I/O port control registers
- The program memory addresses access for all memory locations having executable code and/or data
- The data memory addresses access for all memory locations containing only the data

The following sections describe these three address spaces. For more information about the eZ8 CPU and its address space, refer to the eZ8 CPU Core User Manual (UM0128) which is available for download at www.zilog.com.

Register File

The register file address space in the Z8 Encore! MCU is 4 KB (4096 bytes). The register file consists of two sections—control registers and general-purpose registers. When instructions are executed, registers defined as *Source* are read and registers defined as *Destinations* are written. The architecture of the eZ8 CPU allows all general purpose registers to function as accumulators, address pointers, index registers, stack areas or scratch pad memory.

The upper 256 bytes of the 4 KB register file address space are reserved for controlling the eZ8 CPU, on-chip peripherals and the I/O ports. These registers are located at addresses from F00H to FFFH. Some of the addresses within 256 B Control Register section are reserved (unavailable). Reading from a reserved register file address returns an undefined value. Writing to reserved register file addresses is not recommended and can produce unpredictable results.

The on-chip RAM always begins at address 000H in the register file address space. The Z8 Encore! F0830 Series devices contain up to 2 KB of on-chip RAM. Reading from register file addresses outside the available RAM addresses (and not within the Control Register address space), returns an undefined value and writing to these register file addresses has no effect.

Port A–D Pull-up Enable Subregisters

The Port A–D Pull-Up Enable Subregister is accessed through the Port A–D Control Register by writing 06H to the Port A–D Address Register. See Table 26. Setting the bits in the Port A–D Pull-Up Enable subregisters enables a weak internal resistive pull-up on the specified port pins.

Table 26. Port A–D Pull-Up Enable Subregisters (PxPUE)

Bit	7	6	5	4	3	2	1	0
Field	PPUE7	PPUE6	PPUE5	PPUE4	PPUE3	PPUE2	PPUE1	PPUE0
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	If 06H in Port A–D Address Register, accessible through the Port A–D Control Register							

Bit	Description
[7:0] PxPUE	Port Pull-Up Enable 0 = The weak pull-up on the port pin is disabled. 1 = The weak pull-up on the port pin is enabled.

Note: x indicates the specific GPIO port pin number (7–0).

- 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 for (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 using 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

WDT Reset in Normal Operation

If configured to generate a reset when a time-out occurs, the Watchdog Timer forces the device into the System Reset state. The WDT status bit in the Watchdog Timer Control Register is set to 1. See the Reset and Stop Mode Recovery chapter on page 21 for more information about system reset operations.

WDT Reset in STOP Mode

If configured to generate a reset when a time-out occurs and the device is in STOP Mode, the Watchdog Timer initiates a Stop Mode Recovery. Both the WDT status bit and the STOP bit in the Watchdog Timer Control Register are set to 1 following WDT time-out in STOP Mode. See the Reset and Stop Mode Recovery chapter on page 21 for more information about Stop Mode Recovery operations.

Watchdog Timer Reload Unlock Sequence

Writing the unlock sequence to the Watchdog Timer (WDTCTL) Control Register address, unlocks the three Watchdog Timer Reload Byte registers (WDTU, WDTH and WDTL) to allow changes to the time-out period. These write operations to the WDTCTL Register address produce no effect on the bits in the WDTCTL Register. The locking mechanism prevents spurious writes to the reload registers.

The following sequence is required to unlock the Watchdog Timer Reload Byte registers (WDTU, WDTH and WDTL) for write access:

1. Write 55H to the Watchdog Timer Control Register (WDTCTL).
2. Write AAH to the Watchdog Timer Control Register (WDTCTL).
3. Write the Watchdog Timer Reload Upper Byte Register (WDTU).
4. Write the Watchdog Timer Reload High Byte Register (WDTH).
5. Write the Watchdog Timer Reload Low Byte Register (WDTL).

All three Watchdog Timer Reload registers must be written in the order listed above. There must be no other register writes between each of these operations. If a register write occurs, the lock state machine resets and no further writes can occur unless the sequence is restarted. The value in the Watchdog Timer Reload registers is loaded into the counter when the Watchdog Timer is first enabled and every time a WDT instruction is executed.

ADC Interrupt

The ADC can generate an interrupt request when a conversion has been completed. An interrupt request that is pending when the ADC is disabled is not cleared automatically.

Reference Buffer

The reference buffer, RBUF, supplies the reference voltage for the ADC. When enabled, the internal voltage reference generator supplies the ADC. When RBUF is disabled, the ADC must have the reference voltage supplied externally through the V_{REF} pin in 28-pin package. RBUF is controlled by the REFEN bit in the ADC Control Register.

Internal Voltage Reference Generator

The internal voltage reference generator provides the voltage VR2, for the RBUF. VR2 is 2 V.

Calibration and Compensation

A user can perform calibration and store the results into Flash or the user code can perform a manual offset calibration. There is no provision for manual gain calibration.

ADC Control Register Definitions

The ADC Control registers are defined in this section.

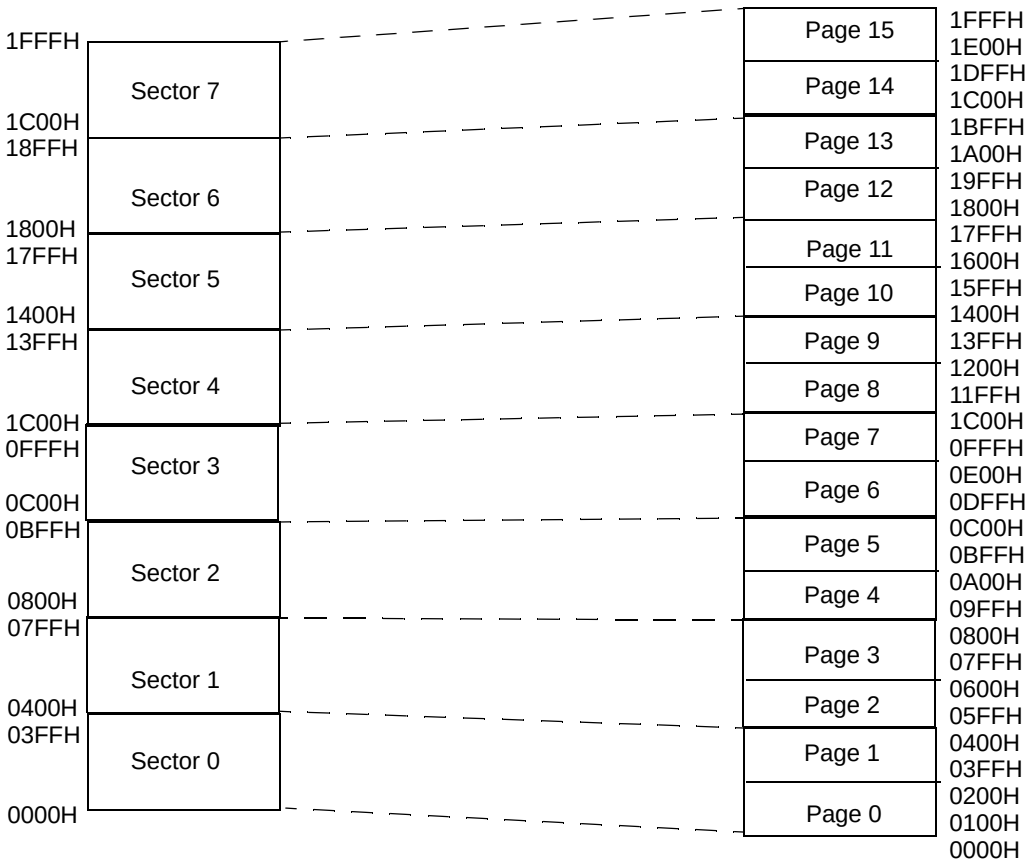

Figure 17. 8K Flash with NVDS

Flash Control Register

The Flash Controller must be unlocked using the Flash Control Register before programming or erasing Flash memory. Writing the sequence 73H-8CH, sequentially, to the Flash Control Register unlocks the Flash Controller. When the Flash Controller is unlocked, Flash memory can be enabled for mass erase or page erase by writing the appropriate enable command to the FCTL. Page erase applies only to the active page selected in Flash Page Select Register. Mass erase is enabled only through the On-Chip Debugger. Writing an invalid value or an invalid sequence returns the Flash Controller to its Locked state. The write-only Flash Control Register shares its register file address with the read-only Flash Status Register.

Table 72. Flash Control Register (FCTL)

Bit	7	6	5	4	3	2	1	0
Field	FCMD							
RESET	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W
Address	FF8H							

Bit	Description
[7:0] FCMD	Flash Command 73H = First unlock command. 8CH = Second unlock command. 95H = Page erase command (must be third command in sequence to initiate page erase). 63H = Mass erase command (must be third command in sequence to initiate mass erase). 5EH = Enable Flash Sector Protect Register access.

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 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 71 µs and 258 µs (assuming a 20 MHz 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 6 µs 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.

! Caution: It is possible to disable the clock failure detection circuitry as well as all functioning clock sources. In this case, the Z8 Encore! F0830 Series device ceases functioning and can only be recovered by power-on-reset.

Oscillator Control Register Definitions

The following section provides the bit definitions for the Oscillator Control Register.

Oscillator Control Register

The Oscillator Control Register (OSCCTL) enables/disables the various oscillator circuits, enables/disables the failure detection/recovery circuitry and selects the primary oscillator, which becomes the system clock.

The Oscillator Control Register must be unlocked before writing. Writing the two step sequence E7H followed by 18H to the Oscillator Control Register unlocks it. The register is locked at successful completion of a register write to the OSCCTL.

Figure 24 displays the oscillator control clock switching flow. See Table 117 on page 189 to review the waiting times of various oscillator circuits.

Table 99. Oscillator Control Register (OSCCTL)

Bit	7	6	5	4	3	2	1	0
Field	INTEN	XTLEN	WDTEN	POFEN	WDFEN	SCKSEL		
RESET	1	0	1	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Address	F86H							

Bit	Description
[7] INTEN	Internal Precision Oscillator Enable 1 = Internal Precision Oscillator is enabled. 0 = Internal Precision Oscillator is disabled.
[6] XTLEN	Crystal Oscillator Enable This setting overrides the GPIO register control for PA0 and PA1. 1 = Crystal oscillator is enabled. 0 = Crystal oscillator is disabled.
[5] WDTEN	Watchdog Timer Oscillator Enable 1 = Watchdog Timer Oscillator is enabled. 0 = Watchdog Timer Oscillator is disabled.

ister, the user code must wait at least 5000 IPO cycles for the crystal to stabilize. After this period, the crystal oscillator may be selected as the system clock.

Figure 25 displays a recommended configuration for connection with an external fundamental-mode, parallel-resonant crystal operated at 20 MHz. Recommended 20 MHz crystal specifications are provided in Table 100. Resistor R_1 is optional and limits total power dissipation by the crystal. Printed circuit board layout must add no more than 4 pF of stray capacitance to either the X_{IN} or X_{OUT} pins. If oscillation does not occur, reduce the values of capacitors C_1 and C_2 to decrease loading.

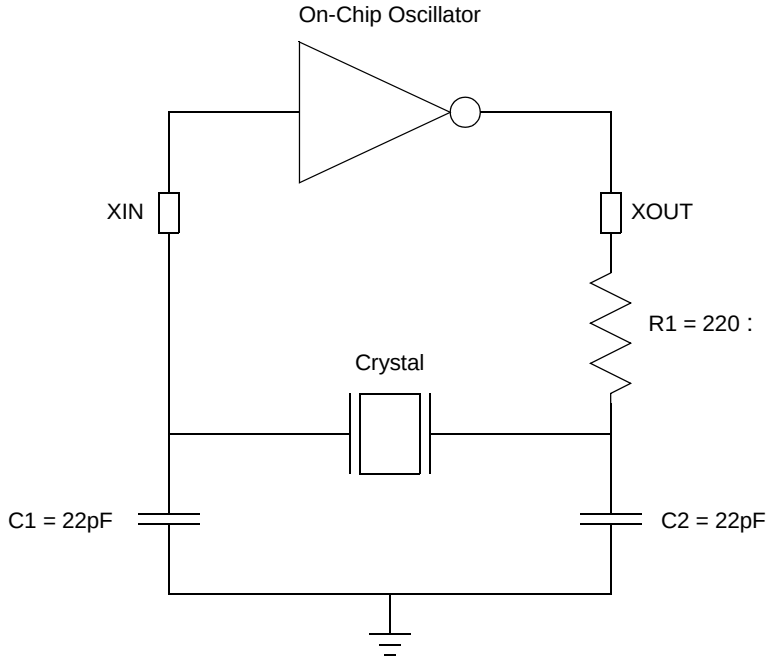

Figure 25. Recommended 20 MHz Crystal Oscillator Configuration

Table 100. Recommended Crystal Oscillator Specifications

Parameter	Value	Units	Comments
Frequency	20	MHz	
Resonance	Parallel		
Mode	Fundamental		
Series Resistance (R_S)	60	:	Maximum
Load Capacitance (C_L)	30	pF	Maximum
Shunt Capacitance (C_0)	7	pF	Maximum
Drive Level	1	mW	Maximum

eZ8 CPU Instruction Summary

Table 113 summarizes the eZ8 CPU instruction set. 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.

Table 113. eZ8 CPU Instruction Summary (Continued)

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
LD dst, rc	dst ← src	r	IM	0C–FC	–	–	–	–	–	–	2	2
		r	X(r)	C7							3	3
		X(r)	r	D7							3	4
		r	Ir	E3							2	3
		R	R	E4							3	2
		R	IR	E5							3	4
		R	IM	E6							3	2
		IR	IM	E7							3	3
		Ir	r	F3							2	3
		IR	R	F5							3	3
LDC dst, src	dst ← src	r	Irr	C2	–	–	–	–	–	–	2	5
		Ir	Irr	C5							2	9
		Irr	r	D2							2	5
LDCI dst, src	dst ← src r ← r + 1 rr ← rr + 1	Ir	Irr	C3	–	–	–	–	–	–	2	9
		Irr	Ir	D3							2	9
LDE dst, src	dst ← src	r	Irr	82	–	–	–	–	–	–	2	5
		Irr	r	92							2	5
LDEI dst, src	dst ← src r ← r + 1 rr ← rr + 1	Ir	Irr	83	–	–	–	–	–	–	2	9
		Irr	Ir	93							2	9
LDWX dst, src	dst ← src	ER	ER	1FE8	–	–	–	–	–	–	5	4

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

Table 113. eZ8 CPU Instruction Summary (Continued)

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
LDX dst, src	dst ← src	r	ER	84	–	–	–	–	–	–	3	2
		Ir	ER	85							3	3
		R	IRR	86							3	4
		IR	IRR	87							3	5
		r	X(rr)	88							3	4
		X(rr)	r	89							3	4
		ER	r	94							3	2
		ER	Ir	95							3	3
		IRR	R	96							3	4
		IRR	IR	97							3	5
		ER	ER	E8							4	2
		ER	IM	E9							4	2
LEA dst, X(src)	dst ← src + X	r	X(r)	98	–	–	–	–	–	–	3	3
		rr	X(rr)	99							3	5
MULT dst	dst[15:0] ← dst[15:8] * dst[7:0]	RR		F4	–	–	–	–	–	–	2	8
NOP	No operation			0F	–	–	–	–	–	–	1	2
OR dst, src	dst ← dst OR src	r	r	42	–	*	*	0	–	–	2	3
		r	Ir	43							2	4
		R	R	44							3	3
		R	IR	45							3	4
		R	IM	46							3	3
		IR	IM	47							3	4
ORX dst, src	dst ← dst OR src	ER	ER	48	–	*	*	0	–	–	4	3
		ER	IM	49							4	3
POP dst	dst ← @SP SP ← SP + 1	R		50	–	–	–	–	–	–	2	2
		IR		51							2	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.

Figure 32 displays the typical current consumption versus the system clock frequency in NORMAL Mode.

Figure 32. I_{CC} Versus System Clock Frequency (NORMAL Mode)

General Purpose I/O Port Output Timing

Figure 34 and Table 125 provide timing information for the GPIO port pins.

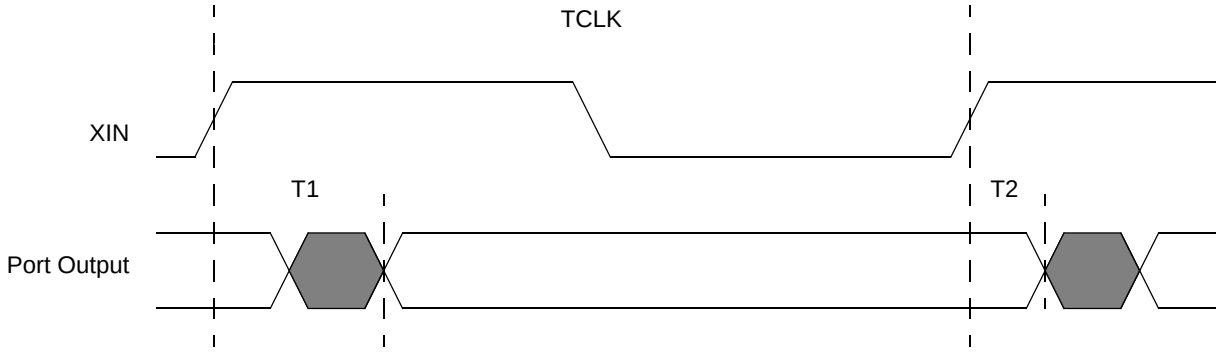

Figure 34. GPIO Port Output Timing

Table 125. GPIO Port Output Timing

		Delay (ns)	
Parameter	Abbreviation	Minimum	Maximum
GPIO Port Pins			
T_1	XIN Rise to Port Output Valid Delay	–	15
T_2	XIN Rise to Port Output Hold Time	2	–

Hex Address: F09

Table 139. Timer 1 Low Byte Register (T1L)

Bit	7	6	5	4	3	2	1	0
Field	TL							
RESET	0	0	0	0	0	0	0	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Address	F09H							

Hex Address: F0A

Table 140. Timer 1 Reload High Byte Register (T1RH)

Bit	7	6	5	4	3	2	1	0
Field	TRH							
RESET	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Address	F0AH							

Hex Address: F0B

Table 141. Timer 1 Reload Low Byte Register (T1RL)

Bit	7	6	5	4	3	2	1	0
Field	TRL							
RESET	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Address	F0BH							

Hex Address: F0C

Table 142. Timer 1 PWM High Byte Register (T1PWMH)

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	F0CH							

Hex Address: FC5

Table 162. IRQ1 Enable Low Bit Register (IRQ1ENL)

Bit	7	6	5	4	3	2	1	0
Field	PA7ENL	PA6CENL	PA5ENL	PA4ENL	PA3ENL	PA2ENL	PA1ENL	PA0ENL
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	FC5H							

Hex Address: FC6

Table 163. Interrupt Request 2 Register (IRQ2)

Bit	7	6	5	4	3	2	1	0
Field	Reserved				PC3I	PC2I	PC1I	PC0I
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	FC6H							

Hex Address: FC7

Table 164. IRQ2 Enable High Bit Register (IRQ2ENH)

Bit	7	6	5	4	3	2	1	0
Field	Reserved				C3ENH	C2ENH	C1ENH	C0ENH
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	FC7H							

Hex Address: FC8

Table 165. IRQ2 Enable Low Bit Register (IRQ2ENL)

Bit	7	6	5	4	3	2	1	0
Field	Reserved				C3ENL	C2ENL	C1ENL	C0ENL
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	FC8H							