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	eZ8
Core Size	8-Bit
Speed	5MHz
Connectivity	IrDA, UART/USART
Peripherals	Brown-out Detect/Reset, LED, POR, PWM, WDT
Number of I/O	16
Program Memory Size	4KB (4K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 3.6V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Surface Mount
Package / Case	20-SOIC (0.295", 7.50mm Width)
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/zilog/z8f0413sh005sg

List of Tables

Table 1. F0823 Series Family Part Selection Guide 2
Table 2. F0823 Series Package Options 7
Table 3. Signal Descriptions 9
Table 4. Pin Characteristics (20- and 28-pin Devices)* 11
Table 5. Pin Characteristics (8-Pin Devices) 12
Table 6. Z8 Encore! XP F0823 Series Program Memory Maps 14
Table 7. F0823 Series Flash Memory Information Area Map 15
Table 8. Register File Address Map 16
Table 9. Reset and Stop Mode Recovery Characteristics and Latency 21
Table 10. Reset Sources and Resulting Reset Type 23
Table 11. Stop Mode Recovery Sources and Resulting Action 27
Table 12. Reset Status Register (RSTSTAT) 28
Table 13. POR Indicator Values 29
Table 14. Power Control Register 0 (PWRCTL0) 32
Table 15. Port Availability by Device and Package Type 33
Table 16. Port Alternate Function Mapping (8-Pin Parts) 35
Table 17. Port Alternate Function Mapping (Non 8-Pin Parts) 36
Table 18. GPIO Port Registers and Subregisters 40
Table 19. Port A–C GPIO Address Registers (PxADDR) 41
Table 20. PADDR[7:0] Subregister Functions 41
Table 21. Port A–C Control Registers (PxCTL) 42
Table 22. Port A–C Data Direction Subregisters (PxDD) 43
Table 23. Port A–C Alternate Function Subregisters (PxAF) 44
Table 24. Port A–C Output Control Subregisters (PxOC) 44
Table 25. Port A–C High Drive Enable Subregisters (PHDEx) 45
Table 26. Port A–C Stop Mode Recovery Source Enable Subregisters (PSMREx) . . 46
Table 27. Port A–C Pull-Up Enable Subregisters (PPUEx) 47
Table 28. Port A–C Alternate Function Set 1 Subregisters (PAFS1x) 48
Table 29. Port A–C Alternate Function Set 2 Subregisters (PxAFS2) 49
Table 30. Port A–C Input Data Registers (PxIN) 50
Table 31. Port A–C Output Data Register (PxOUT) 51
Table 32. LED Drive Enable (LEDEN) 51
Table 33. LED Drive Level High Register (LEDLVLH) 52

Table 70. UART Address Compare Register (U0ADDR) 115
Table 71. UART Baud Rate High Byte Register (U0BRH) 115
Table 72. UART Baud Rate Low Byte Register (U0BRL) 115
Table 73. UART Baud Rates 116
Table 74. ADC Control Register 0 (ADCCTL0) 127
Table 75. ADC Control/Status Register 1 (ADCCTL1) 129
Table 76. ADC Data High Byte Register (ADCD_H) 130
Table 77. ADC Data Low Bits Register (ADCD_L) 131
Table 78. Comparator Control Register (CMP0) 133
Table 79. Z8 Encore! XP F0823 Series Flash Memory Configurations 134
Table 80. Flash Code Protection Using the Flash Option Bits 138
Table 81. Flash Control Register (FCTL) 141
Table 82. Flash Status Register (FSTAT) 142
Table 83. Flash Page Select Register (FPS) 143
Table 84. Flash Sector Protect Register (FPROT) 144
Table 85. Flash Frequency High Byte Register (FFREQH) 145
Table 86. Flash Frequency Low Byte Register (FFREQL) 145
Table 87. Trim Bit Address Register (TRMADR) 148
Table 88. Trim Bit Data Register (TRMDR) 149
Table 89. Flash Option Bits at Program Memory Address 0000H 149
Table 90. Flash Options Bits at Program Memory Address 0001H 150
Table 91. Trim Options Bits at Address 0000H 151
Table 92. Trim Option Bits at 0001H 152
Table 93. Trim Option Bits at 0002H (TIPO) 152
Table 94. ADC Calibration Bits 153
Table 95. ADC Calibration Data Location 153
Table 96. Serial Number at 001C–001F (S_NUM) 154
Table 97. Serialization Data Locations 154
Table 98. Lot Identification Number (RAND_LOT) 154
Table 99. Randomized Lot ID Locations 155
Table 100. OCD Baud-Rate Limits 160
Table 101. OCD Commands 162
Table 102. OCD Control Register (OCDCTL) 167
Table 103. OCD Status Register (OCDSTAT) 168
Table 104. Oscillator Configuration and Selection 169
Table 105. Oscillator Control Register (OSCCTL) 172

Pin Characteristics

Table 4 provides detailed information about the characteristics for each pin available on Z8 Encore! XP F0823 Series 20- and 28-pin devices. Data in Table 4 is sorted alphabetically by the pin symbol mnemonic.

Note: All six I/O pins on the 8-pin packages are 5V-tolerant (unless the pull-up devices are enabled). The right-most column in Table 4 describes 5V tolerance for the 20- and 28-pin packages only.

Table 4. Pin Characteristics (20- and 28-pin Devices)*

Symbol Mnemonic	Direction	Reset Direction	Active Low or Active High	Tristate Output	Internal Pull-up or Pull-down	Schmitt-Trigger Input	Open Drain Output	5V Tolerance
AVDD	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
AVSS	N/A	N/A	N/A	N/A	N/A	N/A	N/A	NA
DBG	I/O	I	N/A	Yes	No	Yes	Yes	Yes
PA[7:0]	I/O	I	N/A	Yes	Programmable Pull-up	Yes	Yes, Programmable	PA[7:2] only
PB[7:0]	I/O	I	N/A	Yes	Programmable Pull-up	Yes	Yes, Programmable	PB[7:6] only
PC[7:0]	I/O	I	N/A	Yes	Programmable Pull-up	Yes	Yes, Programmable	PC[7:3] only
RESET	I/O	I/O (defaults to RESET)	Low (in Reset mode)	Yes (PD0 only)	Always on for RESET	Yes	Always on for RESET	Yes
VDD	N/A	N/A	N/A	N/A			N/A	N/A
VSS	N/A	N/A	N/A	N/A			N/A	N/A

Note: PB6 and PB7 are available only in the devices without ADC.

Architecture

Figure 7 displays a simplified block diagram of a GPIO port pin. In this figure, the ability to accommodate alternate functions and variable port current drive strength is not displayed.

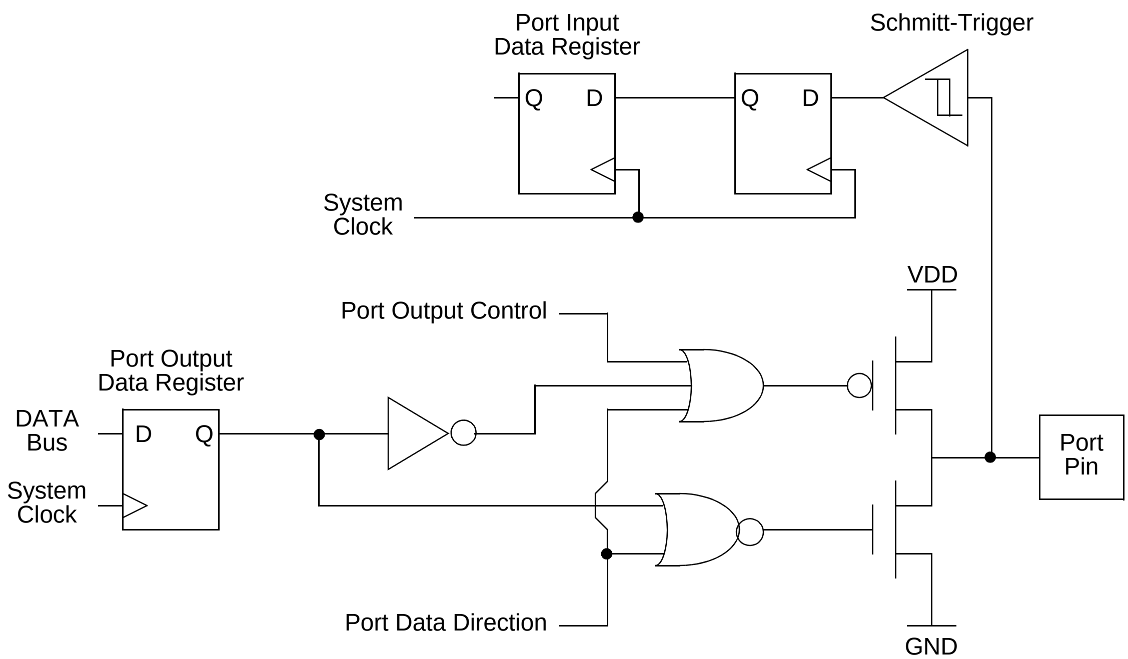

Figure 7. GPIO Port Pin Block Diagram

GPIO Alternate Functions

Many of the GPIO port pins are used for general-purpose I/O and access to on-chip peripheral functions such as the timers and serial communication devices. The port A–D Alternate Function subregisters configure these pins for either GPIO or alternate function operation. When a pin is configured for alternate function, control of the port pin direction (input/output) is passed from the Port A–D Data Direction registers to the alternate function assigned to this pin. Tables 16 and 17 list the alternate functions possible with each port pin for 8-pin and non-8-pin parts, respectively. The alternate function associated at a pin is defined through Alternate Function Sets subregisters AFS1 and AFS2.

The crystal oscillator functionality is not controlled by the GPIO block. When the crystal oscillator is enabled in the oscillator control block, the GPIO functionality of PA0 and PA1 is overridden. In that case, those pins function as input and output for the crystal oscillator.

Table 17. Port Alternate Function Mapping (Non 8-Pin Parts) (Continued)

Port	Pin	Mnemonic	Alternate Function Description	Alternate Function Set Register AFS1
Port C[4]	PC0	Reserved		AFS1[0]: 0
		ANA4/CINP	ADC or Comparator Input	AFS1[0]: 1
	PC1	Reserved		AFS1[1]: 0
		ANA5/CINN	ADC or Comparator Input	AFS1[1]: 1
	PC2	Reserved		AFS1[2]: 0
		ANA6/V_{REF}[6]	ADC Analog Input or ADC Voltage Reference	AFS1[2]: 1
	PC3	COUT	Comparator Output	AFS1[3]: 0
		Reserved		AFS1[3]: 1
	PC4	Reserved		AFS1[4]: 0
				AFS1[4]: 1
	PC5	Reserved		AFS1[5]: 0
				AFS1[5]: 1
	PC6	Reserved		AFS1[6]: 0
				AFS1[6]: 1
	PC7	Reserved		AFS1[7]: 0
				AFS1[7]: 1

Notes:
1. Because there is only a single alternate function for each Port A pin, the Alternate Function Set registers are not implemented for Port A. Enabling alternate function selections as described in the Port A–C Alternate Function Subregisters section on page 43 automatically enables the associated alternate function.
2. Whether PA0/PA6 take on the timer input or timer output complement function depends on the timer configuration as described in the Timer Pin Signal Operation section on page 83.
3. Because there are at most two choices of alternate function for any pin of Port B, the Alternate Function Set register AFS2 is implemented but not used to select the function. Also, alternate function selection as described in the Port A–C Alternate Function Subregisters section on page 43 must also be enabled.
4. V_{REF} is available on PB5 in 28-pin products only.
5. Because there are at most two choices of alternate function for any pin of Port C, the Alternate Function Set register AFS2 is implemented but not used to select the function. Also, Alternate Function selection as described in the Port A–C Alternate Function Subregisters section on page 43 must also be enabled.
6. V_{REF} is available on PC2 in 20-pin parts only.

Direct LED Drive

The Port C pins provide a current sinked output capable of driving an LED without requiring an external resistor. The output sinks current at programmable levels of 3mA, 7mA, 13mA, and 20mA. This mode is enabled through the LED control registers. The LED Drive Enable (LEDEN) register turns on the drivers. The LED Drive Level (LEDLVLH and LEDLVLL) registers select the sink current.

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.

External Clock Setup

For systems using an external TTL drive, PB3 is the clock source for 20- and 28-pin devices. In this case, configure PB3 for alternate function CLKIN. Write the Oscillator Control Register (see the Oscillator Control Register Definitions section on page 171) such that the external oscillator is selected as the system clock. For 8-pin devices, use PA1 instead of PB3.

GPIO Interrupts

Many of the GPIO port pins are used as interrupt sources. Some port pins are configured to generate an interrupt request on either the rising edge or falling edge of the pin input signal. Other port pin interrupt sources generate an interrupt when any edge occurs (both rising and falling). For more information about interrupts using the GPIO pins, see the Interrupt Controller chapter on page 54.

GPIO Control Register Definitions

Four registers for each port provide access to GPIO control, input data, and output data. Table 18 lists these port registers. Use the Port A–D Address and Control registers together to provide access to subregisters for port configuration and control.

Table 18. GPIO Port Registers and Subregisters

Port Register Mnemonic	Port Register Name
P*x*ADDR	Port A–C Address Register (Selects subregisters).
P*x*CTL	Port A–C Control Register (Provides access to subregisters).
P*x*IN	Port A–C Input Data Register.
P*x*OUT	Port A–C Output Data Register.
Port Subregister Mnemonic	**Port Register Name**
P*x*DD	Data Direction.
P*x*AF	Alternate Function.
P*x*OC	Output Control (Open-Drain).

Table 23. Port A–C Alternate Function Subregisters (PxAF)

Bit	7	6	5	4	3	2	1	0
Field	AF7	AF6	AF5	AF4	AF3	AF2	AF1	AF0
RESET	00H (Ports A–C); 04H (Port A of 8-pin device)							
R/W	R/W							
Address	If 02H in Port A–C Address Register, accessible through the Port A–C Control Register							

Bit	Description
[7:0] AFx	**Port Alternate Function enabled** 0 = The port pin is in NORMAL Mode and the `DDx` bit in the Port A–C Data Direction Subregister determines the direction of the pin. 1 = The alternate function selected through Alternate Function Set subregisters is enabled. Port pin operation is controlled by the alternate function.

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

Port A–C Output Control Subregisters

The Port A–C Output Control Subregister (Table 24) is accessed through the Port A–C Control Register by writing `03H` to the Port A–C Address Register. Setting the bits in the Port A–C Output Control subregisters to 1 configures the specified port pins for open-drain operation. These subregisters affect the pins directly and, as a result, alternate functions are also affected.

Table 24. Port A–C Output Control Subregisters (PxOC)

Bit	7	6	5	4	3	2	1	0
Field	POC7	POC6	POC5	POC4	POC3	POC2	POC1	POC0
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 03H in Port A–C Address Register, accessible through the Port A–C Control Register							

Bit	Description
[7:0] POCx	**Port Output Control** These bits function independently of the alternate function bit and always disable the drains if set to 1. 0 = The drains are enabled for any output mode (unless overridden by the alternate function). 1 = The drain of the associated pin is disabled (open-drain mode).

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

Port A–C Alternate Function Set 1 Subregisters

The Port A–C Alternate Function Set1 Subregister (Table 28) is accessed through the Port A–C Control Register by writing `07H` to the Port A–C Address Register. The Alternate Function Set 1 subregisters selects the alternate function available at a port pin. Alternate Functions selected by setting or clearing bits of this register are defined in **“GPIO Alternate Functions” on page 34**.

Note: Alternate function selection on port pins must also be enabled as described in the Port A–C Alternate Function Subregisters section on page 43.

Table 28. Port A–C Alternate Function Set 1 Subregisters (PAFS1x)

Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Field	PAFS17	PAFS16	PAFS15	PAFS14	PAFS13	PAFS12	PAFS11	PAFS10
RESET	00H (all ports of 20/28 pin devices); 04H (Port A of 8-pin device)							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Address	If 07H in Port A–C Address Register, accessible through the Port A–C Control Register							

Bit	**Description**
[7:0] PAFS1x	**Port Alternate Function Set to 1** 0 = Port Alternate Function selected as defined in Table 15 (see the GPIO Alternate Functions section on page 34). 1 = Port Alternate Function selected as defined in Table 15 (see the GPIO Alternate Functions section on page 34).

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

Table 47. 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							

Bit	Description
[7:4]	**Reserved** These bits are reserved and must be programmed to 0000.
[3] C3ENL	**Port C3 Interrupt Request Enable Low Bit**
[2] C2ENL	**Port C2 Interrupt Request Enable Low Bit**
[1] C1ENL	**Port C1 Interrupt Request Enable Low Bit**
[0] C0ENL	**Port C0 Interrupt Request Enable High Low**

Interrupt Edge Select Register

The Interrupt Edge Select (IRQES) Register (Table 48) determines whether an interrupt is generated for the rising edge or falling edge on the selected GPIO Port A or Port D input pin.

Table 48. Interrupt Edge Select Register (IRQES)

Bit	7	6	5	4	3	2	1	0
Field	IES7	IES6	IES5	IES4	IES3	IES2	IES1	IES0
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	FCDH							

Bit	Description
[7] IESx	**Interrupt Edge Select *x*** 0 = An interrupt request is generated on the falling edge of the PA*x* input or PDx. 1 = An interrupt request is generated on the rising edge of the PA*x* input PDx.
Note: *x* indicates the specific GPIO port pin number (7–0).	

Observe the following steps to configure a timer for PWM DUAL OUTPUT Mode and initiating the PWM operation:

1. Write to the Timer Control Register to:
   - Disable the timer
   - Configure the timer for PWM DUAL OUTPUT Mode. Setting the mode also involves writing to the TMODEHI bit in the TxCTL1 Register
   - Set the prescale value
   - Set the initial logic level (High or Low) and PWM High/Low transition for the Timer Output alternate function
2. Write to the Timer High and Low Byte registers to set the starting count value (typically `0001H`); this write only affects the first pass in PWM Mode. After the first timer reset in PWM Mode, counting always begins at the reset value of `0001H`.
3. Write to the PWM High and Low Byte registers to set the PWM value.
4. Write to the PWM Control Register to set the PWM dead band delay value. The deadband delay must be less than the duration of the positive phase of the PWM signal (as defined by the PWM high and low byte registers). It must also be less than the duration of the negative phase of the PWM signal (as defined by the difference between the PWM registers and the Timer Reload registers).
5. Write to the Timer Reload High and Low Byte registers to set the reload value (PWM period). The reload value must be greater than the PWM value.
6. If appropriate, enable the timer interrupt and set the timer interrupt priority by writing to the relevant interrupt registers.
7. Configure the associated GPIO port pin for the Timer Output and Timer Output Complement alternate functions. The Timer Output Complement function is shared with the Timer Input function for both timers. Setting the timer mode to Dual PWM automatically switches the function from Timer In to Timer Out Complement.
8. Write to the Timer Control Register to enable the timer and initiate counting.

The PWM period is represented by the following equation:

$$\text{PWM Period (s)} = \frac{\text{Reload Value} \times \text{Prescale}}{\text{System Clock Frequency (Hz)}}$$

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

If `TPOL` is set to 0, the ratio of the PWM output High time to the total period is represented by:

7. Return to Step 4 to receive additional data.

Receiving Data Using the Interrupt-Driven Method

The UART Receiver interrupt indicates the availability of new data (as well as error conditions). Observe the following steps to configure the UART receiver for interrupt-driven operation:

1. Write to the UART Baud Rate High and Low Byte registers to set the acceptable baud rate.
2. Enable the UART pin functions by configuring the associated GPIO port pins for alternate function operation.
3. Execute a `DI` instruction to disable interrupts.
4. Write to the Interrupt control registers to enable the UART Receiver interrupt and set the acceptable priority.
5. Clear the UART Receiver interrupt in the applicable Interrupt Request register.
6. Write to the UART Control 1 Register to enable Multiprocessor (9-bit) mode functions, if appropriate.
   - Set the Multiprocessor Mode Select (`MPEN`) to Enable MULTIPROCESSOR Mode
   - Set the Multiprocessor Mode Bits, `MPMD[1:0]`, to select the acceptable address matching scheme
   - Configure the UART to interrupt on received data and errors or errors only (interrupt on errors only is unlikely to be useful for Z8 Encore! XP devices without a DMA block)
7. Write the device address to the Address Compare Register (automatic MULTIPROCESSOR modes only).
8. Write to the UART Control 0 Register to:
   - Set the receive enable bit (`REN`) to enable the UART for data reception
   - Enable parity, if appropriate and if multiprocessor mode is not enabled, and select either even or odd parity
9. Execute an `EI` instruction to enable interrupts.

The UART is now configured for interrupt-driven data reception. When the UART Receiver interrupt is detected, the associated interrupt service routine (ISR) performs the following:

scheme is enabled, the UART Address Compare register holds the network address of the device.

MULTIPROCESSOR (9-bit) Mode Receive Interrupts

When MULTIPROCESSOR Mode is enabled, the UART only processes frames addressed to it. The determination of whether a frame of data is addressed to the UART can be made in hardware, software or some combination of the two, depending on the multiprocessor configuration bits. In general, the address compare feature reduces the load on the CPU, because it does not require access to the UART when it receives data directed to other devices on the multi-node network. The following three MULTIPROCESSOR modes are available in hardware:

- Interrupt on all address bytes
- Interrupt on matched address bytes and correctly framed data bytes
- Interrupt only on correctly framed data bytes

These modes are selected with `MPMD[1:0]` in the UART Control 1 Register. For all multiprocessor modes, bit `MPEN` of the UART Control 1 Register must be set to 1.

The first scheme is enabled by writing `01b` to MPMD[1:0]. In this mode, all incoming address bytes cause an interrupt, while data bytes never cause an interrupt. The interrupt service routine must manually check the address byte that caused triggered the interrupt. If it matches the UART address, the software clears MPMD[0]. Each new incoming byte interrupts the CPU. The software is responsible for determining the end of the frame. It checks for the end-of-frame by reading the `MPRX` bit of the UART Status 1 Register for each incoming byte. If `MPRX`=1, a new frame has begun. If the address of this new frame is different from the UART's address, MPMD[0] must be set to 1 causing the UART interrupts to go inactive until the next address byte. If the new frame's address matches the UART's, the data in the new frame is processed as well.

The second scheme requires the following: set MPMD[1:0] to `10B` and write the UART's address into the UART Address Compare register. This mode introduces additional hardware control, interrupting only on frames that match the UART's address. When an incoming address byte does not match the UART's address, it is ignored. All successive data bytes in this frame are also ignored. When a matching address byte occurs, an interrupt is issued and further interrupts now occur on each successive data byte. When the first data byte in the frame is read, the `NEWFRM` bit of the UART Status 1 Register is asserted. All successive data bytes have `NEWFRM`=0. When the next address byte occurs, the hardware compares it to the UART's address. If there is a match, the interrupts continues and the `NEWFRM` bit is set for the first byte of the new frame. If there is no match, the UART ignores all incoming bytes until the next address match.

The third scheme is enabled by setting MPMD[1:0] to `11b` and by writing the UART's address into the UART Address Compare Register. This mode is identical to the second passed to the UART. Communication is half-duplex, which means simultaneous data transmission and reception is not allowed.

The baud rate is set by the UART's baud rate generator and supports IrDA standard baud rates from 9600 baud to 115.2 kbaud. Higher baud rates are possible, but do not meet IrDA specifications. The UART must be enabled to use the infrared endec. The infrared endec data rate is calculated using the following equation:

$$\text{Infrared Data Rate (bits/s)} = \frac{\text{System Clock Frequency (Hz)}}{16 \times \text{UART Baud Rate Divisor Value}}$$

Transmitting IrDA Data

The data to be transmitted using the infrared transceiver is first sent to the UART. The UART's transmit signal (TXD) and baud rate clock are used by the IrDA to generate the modulation signal (IR_TXD) that drives the infrared transceiver. Each UART/Infrared data bit is 16 clocks wide. If the data to be transmitted is 1, the IR_TXD signal remains low for the full 16 clock period. If the data to be transmitted is 0, the transmitter first outputs a 7 clock low period, followed by a 3 clock high pulse. Finally, a 6 clock low pulse is output to complete the full 16 clock data period. Figure 17 displays IrDA data transmission. When the infrared endec is enabled, the UART's TXD signal is internal to Z8 Encore! XP F0823 Series products while the IR_TXD signal is output through the TXD pin.

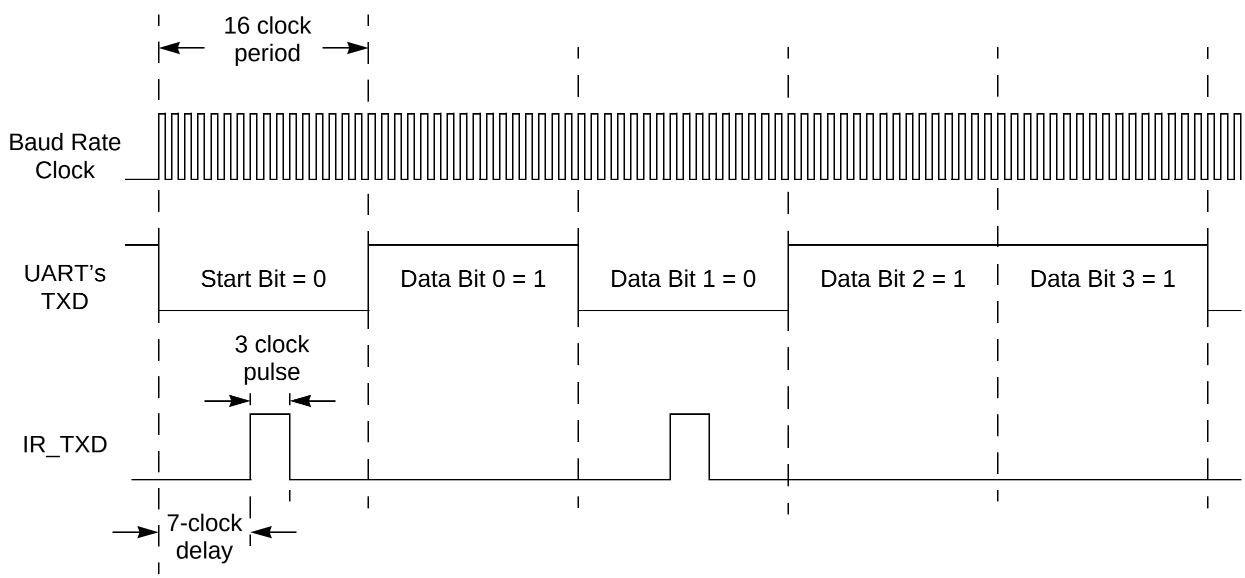

Figure 17. Infrared Data Transmission

bits), framed between High bits. The auto-baud detector measures this period and sets the OCD baud rate generator accordingly.

The auto-baud detector/generator is clocked by the system clock. The minimum baud rate is the system clock frequency divided by 512. For optimal operation with asynchronous datastreams, the maximum recommended baud rate is the system clock frequency divided by eight. The maximum possible baud rate for asynchronous datastreams is the system clock frequency divided by four, but this theoretical maximum is possible only for low noise designs with clean signals. Table 100 lists minimum and recommended maximum baud rates for sample crystal frequencies.

Table 100. OCD Baud-Rate Limits

System Clock Frequency (MHz)	Recommended Maximum Baud Rate (kbps)	Recommended Standard PC Baud Rate (bps)	Minimum Baud Rate (kbps)
5.5296	1382.4	691,200	1.08
0.032768 (32 kHz)	4.096	2400	0.064

If the OCD receives a Serial Break (nine or more continuous bits Low) the auto-baud detector/generator resets. Reconfigure the auto-baud detector/generator by sending `80H`.

OCD Serial Errors

The OCD detects 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 (OCD and host simultaneous transmission 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 auto-baud detector/generator. A Framing Error or Transmit Collision may be caused by the host sending a Serial Break to the OCD. Because of the open-drain nature of the interface, returning a Serial Break 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 F0823 Series devices or when recovering from an error. A Serial Break from the host resets the auto-baud 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 sends a Serial Break to the OCD even if the OCD is transmitting a character.

Table 118. eZ8 CPU Instruction Summary (Continued)

Assembly Mnemonic	Symbolic Operation	Address Mode dst	Address Mode src	Opcode(s) (Hex)	Flags C	Z	S	V	D	H	Fetch Cycles	Instr. Cycles
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
AND dst, src	dst ← dst AND src	r	r	52	–	*	*	0	–	–	2	3
		r	Ir	53							2	4
		R	R	54							3	3
		R	IR	55							3	4
		R	IM	56							3	3
		IR	IM	57							3	4
ANDX dst, src	dst ← dst AND src	ER	ER	58	–	*	*	0	–	–	4	3
		ER	IM	59							4	3
ATM	Block all interrupt and DMA requests during execution of the next 3 instructions			2F	–	–	–	–	–	–	1	2
BCLR bit, dst	dst[bit] ← 0	r		E2	–	–	–	–	–	–	2	2
BIT p, bit, dst	dst[bit] ← p	r		E2	–	–	–	0	–	–	2	2
BRK	Debugger Break			00	–	–	–	–	–	–	1	1
BSET bit, dst	dst[bit] ← 1	r		E2	–	–	–	0	–	–	2	2
BSWAP dst	dst[7:0] ← dst[0:7]	R		D5	X	*	*	0	–	–	2	2
BTJ p, bit, src, dst	if src[bit] = p PC ← PC + X		r	F6	–	–	–	–	–	–	3	3
			Ir	F7							3	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 128. Analog-to-Digital Converter Electrical Characteristics and Timing

Symbol	Parameter	V_{DD} = 3.0V to 3.6V T_A = 0°C to +70°C (unless otherwise stated) Minimum	Typical	Maximum	Units	Conditions
	Resolution	10		–	bits	
	Differential Nonlinearity (DNL)	–1.0	–	1.0	LSB[3]	External V_{REF} = 2.0V; $R_S \leftarrow$ 3.0 kΩ
	Integral Nonlinearity (INL)	–3.0	–	3.0	LSB[3]	External V_{REF} = 2.0V; $R_S \leftarrow$ 3.0 kΩ
	Offset Error with Calibration		$\pm$1		LSB[3]	
	Absolute Accuracy with Calibration		$\pm$3		LSB[3]	
V_{REF}	Internal Reference Voltage	1.0 2.0	1.1 2.2	1.2 2.4	V	REFSEL=01 REFSEL=10
V_{REF}	Internal Reference Variation with Temperature		$\pm$1.0		%	Temperature variation with V_{DD} = 3.0
V_{REF}	Internal Reference Voltage Variation with V_{DD}		$\pm$0.5		%	Supply voltage variation with T_A = 30°C
R_{REFOUT}	Reference Buffer Output Impedance		850		W	When the internal reference is buffered and driven out to the VREF pin (REFOUT = 1)
	Single-Shot Conversion Time	–	5129	–	System clock cycles	All measurements but temperature sensor
			10258			Temperature sensor measurement
	Continuous Conversion Time	–	256	–	System clock cycles	All measurements but temperature sensor
			512			Temperature sensor measurement

Notes:
1. Analog source impedance affects the ADC offset voltage (because of pin leakage) and input settling time.
2. Devices are factory calibrated at V_{DD} = 3.3 V and T_A = +30°C, so the ADC is maximally accurate under these conditions.
3. LSBs are defined assuming 10-bit resolution.
4. This is the maximum recommended resistance seen by the ADC input pin.
5. The input impedance is inversely proportional to the system clock frequency.

Table 135. Z8 Encore! XP F0823 Series Ordering Matrix (Continued)

Part Number	Flash	RAM	I/O Lines	Interrupts	16-Bit Timers w/PWM	10-Bit A/D Channels	UART with IrDA	Description
Z8 Encore! XP F0823 Series with 4 KB Flash								
Standard Temperature: 0°C to 70°C								
Z8F0413PB005SG	4 KB	1 KB	6	12	2	0	1	PDIP 8-pin package
Z8F0413QB005SG	4 KB	1 KB	6	12	2	0	1	QFN 8-pin package
Z8F0413SB005SG	4 KB	1 KB	6	12	2	0	1	SOIC 8-pin package
Z8F0413SH005SG	4 KB	1 KB	16	18	2	0	1	SOIC 20-pin package
Z8F0413HH005SG	4 KB	1 KB	16	18	2	0	1	SSOP 20-pin package
Z8F0413PH005SG	4 KB	1 KB	16	18	2	0	1	PDIP 20-pin package
Z8F0413SJ005SG	4 KB	1 KB	24	18	2	0	1	SOIC 28-pin package
Z8F0413HJ005SG	4 KB	1 KB	24	18	2	0	1	SSOP 28-pin package
Z8F0413PJ005SG	4 KB	1 KB	24	18	2	0	1	PDIP 28-pin package
Extended Temperature: –40°C to 105°C								
Z8F0413PB005EG	4 KB	1 KB	6	12	2	0	1	PDIP 8-pin package
Z8F0413QB005EG	4 KB	1 KB	6	12	2	0	1	QFN 8-pin package
Z8F0413SB005EG	4 KB	1 KB	6	12	2	0	1	SOIC 8-pin package
Z8F0413SH005EG	4 KB	1 KB	16	18	2	0	1	SOIC 20-pin package
Z8F0413HH005EG	4 KB	1 KB	16	18	2	0	1	SSOP 20-pin package
Z8F0413PH005EG	4 KB	1 KB	16	18	2	0	1	PDIP 20-pin package
Z8F0413SJ005EG	4 KB	1 KB	24	18	2	0	1	SOIC 28-pin package
Z8F0413HJ005EG	4 KB	1 KB	24	18	2	0	1	SSOP 28-pin package
Z8F0413PJ005EG	4 KB	1 KB	24	18	2	0	1	PDIP 28-pin package

LD 180
LDC 180
LDCI 179, 180
LDE 180
LDEI 179
LDX 180
LEA 180
load 180
logical 181
MULT 179
NOP 180
OR 181
ORX 181
POP 180
POPX 180
program control 181
PUSH 180
PUSHX 180
RCF 179, 180
RET 181
RL 181
RLC 181
rotate and shift 181
RR 182
RRC 182
SBC 179
SCF 179, 180
SRA 182
SRL 182
SRP 180
STOP 180
SUB 179
SUBX 179
SWAP 182
TCM 179
TCMX 179
TM 179
TMX 179
TRAP 181
Watchdog Timer refresh 180
XOR 181
XORX 181
instructions, eZ8 classes of 178
interrupt control register 68
Interrupt Controller 54
interrupt controller
architecture 54
interrupt assertion types 57
interrupt vectors and priority 57
operation 56
register definitions 59
software interrupt assertion 58
interrupt edge select register 66
interrupt request 0 register 59
interrupt request 1 register 60
interrupt request 2 register 61
interrupt return 181
interrupt vector listing 54
interrupts
UART 105
IR 176
Ir 176
IrDA
architecture 117
block diagram 117
control register definitions 120
operation 117
receiving data 119
transmitting data 118
IRET 181
IRQ0 enable high and low bit registers 61
IRQ1 enable high and low bit registers 63
IRQ2 enable high and low bit registers 65
IRR 176
Irr 176

J

JP 181
jump, conditional, relative, and relative conditional 181

L

LD 180
LDC 180
LDCI 179, 180
LDE 180

ONE-SHOT mode 70, 88
operating mode 70
PWM mode 75, 76, 89
reading the timer count values 83
reload high and low byte registers 84
timer control register definitions 83
timer output signal operation 83
timers 0-3
control registers 86, 87
high and low byte registers 83, 86
TM 179
TMX 179
tools, hardware and software 220
transmit
IrDA data 118
transmitting UART data-polled method 99
transmitting UART dat-interrupt-driven method 100
TRAP 181

U

UART 4
architecture 97
baud rate generator 108
control register definitions 108
controller signals 9
interrupts 105
MULTIPROCESSOR mode 103
receiving data using interrupt-driven method 102
receiving data using the polled method 101
transmitting data using the interrupt-driven method 100
transmitting data using the polled method 99
x baud rate high and low registers 115
x control 0 and control 1 registers 112
x status 0 and status 1 registers 110, 111
UxBRH register 115
UxBRL register 115
UxCTL0 register 112, 115
UxCTL1 register 113
UxRXD register 109
UxSTAT0 register 110
UxSTAT1 register 111
UxTXD register 109

V

vector 177
Voltage Brownout reset (VBR) 24

W

Watchdog Timer
approximate time-out delay 91
CNTL 24
control register 94, 171
electrical characteristics and timing 202, 204
interrupt in normal operation 92
interrupt in STOP mode 92
refresh 92, 180
reload unlock sequence 93
reload upper, high and low registers 94
reset 25
reset in normal operation 93
reset in STOP mode 93
time-out response 92
Watchdog Timer Control Register (WDTCTL) 94
WDTCTL register 94, 133, 172
WDTH register 95
WDTL register 95
WDTU register 95
working register 176
working register pair 177

X

X 177
XOR 181
XORX 181

Z

Z8 Encore!
block diagram 3
features 1

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.