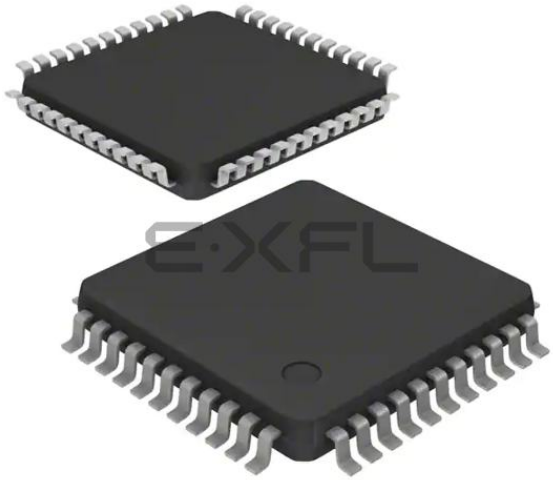


Welcome to E-XFL.COM

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	I²C, IrDA, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, DMA, POR, PWM, WDT
Number of I/O	31
Program Memory Size	64KB (64K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	4K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 3.6V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	44-LQFP
Supplier Device Package	44-LQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/zilog/z8f6421an020eg

DMA Controller

The Z8 Encore! XP F64xx Series feature three channels of DMA. Two of the channels are for register RAM to and from I/O operations. The third channel automatically controls the transfer of data from the ADC to the memory.

Pin Configurations

Figures 2 through 7 display the pin configurations for all of the packages available in the Z8 Encore! XP F64xx Series. For signal descriptions, see Table 3 on page 14.

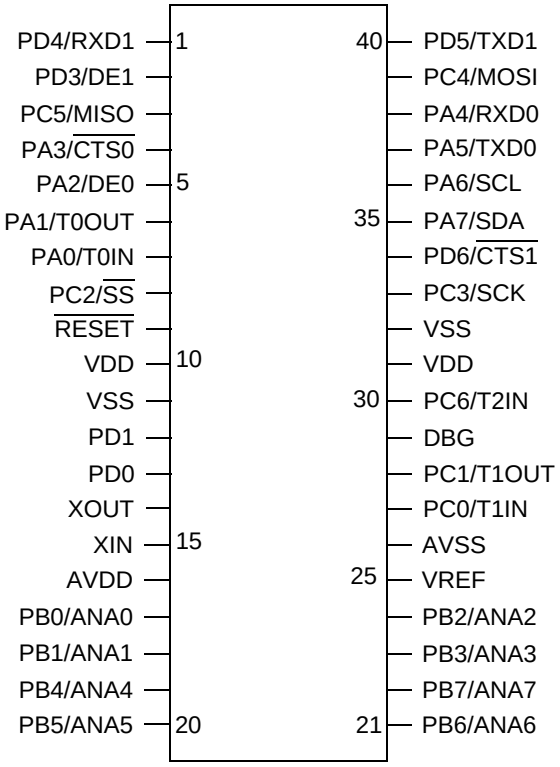


Figure 2. Z8 Encore! XP F64xx Series in 40-Pin Dual Inline Package (PDIP)

Note: Timer 3 and T2OUT are not supported in the 40-pin PDIP package.

Signal Descriptions

Table 3 lists the Z8 Encore! XP signals. To determine the available signals for a specific package style, see the Pin Configurations section on page 8.

Table 3. Signal Descriptions

Signal Mnemonic	I/O	Description
General-Purpose I/O Ports A–H		
PA[7:0]	I/O	Port A[7:0]. These pins are used for general-purpose I/O and support 5V-tolerant inputs.
PB[7:0]	I/O	Port B[7:0]. These pins are used for general-purpose I/O.
PC[7:0]	I/O	Port C[7:0]. These pins are used for general-purpose I/O. These pins are used for general-purpose I/O and support 5V-tolerant inputs
PD[7:0]	I/O	Port D[7:0]. These pins are used for general-purpose I/O. These pins are used for general-purpose I/O and support 5V-tolerant inputs
PE[7:0]	I/O	Port E[7:0]. These pins are used for general-purpose I/O. These pins are used for general-purpose I/O and support 5V-tolerant inputs.
PF[7:0]	I/O	Port F[7:0]. These pins are used for general-purpose I/O. These pins are used for general-purpose I/O and support 5V-tolerant inputs.
PG[7:0]	I/O	Port G[7:0]. These pins are used for general-purpose I/O. These pins are used for general-purpose I/O and support 5V-tolerant inputs.
PH[3:0]	I/O	Port H[3:0]. These pins are used for general-purpose I/O.
I^2C Controller		
SCL	O	Serial Clock. This is the output clock for the I^2C. This pin is multiplexed with a general-purpose I/O pin. When the general-purpose I/O pin is configured for alternate function to enable the SCL function, this pin is open-drain.
SDA	I/O	Serial Data. This open-drain pin transfers data between the I^2C and a slave. This pin is multiplexed with a general-purpose I/O pin. When the general-purpose I/O pin is configured for alternate function to enable the SDA function, this pin is open-drain.
SPI Controller		
$\overline{SS}$	I/O	Slave Select. This signal can be an output or an input. If the Z8 Encore! XP F64xx Series is the SPI master, this pin may be configured as the Slave Select output. If the Z8 Encore! XP F64xx Series is the SPI slave, this pin is the input slave select. It is multiplexed with a general-purpose I/O pin.
SCK	I/O	SPI Serial Clock. The SPI master supplies this pin. If the Z8 Encore! XP F64xx Series is the SPI master, this pin is an output. If the Z8 Encore! XP F64xx Series is the SPI slave, this pin is an input. It is multiplexed with a general-purpose I/O pin.

Power-On Reset

Each device in the Z8 Encore! XP F64xx Series contains an internal Power-On Reset circuit. The POR circuit monitors the supply voltage and holds the device in the Reset state until the supply voltage reaches a safe operating level. After the supply voltage exceeds the POR voltage threshold (V_{POR}), the POR Counter is enabled and counts 66 cycles of the Watchdog Timer oscillator. After the POR counter times out, the XTAL Counter is enabled to count a total of 16 system clock pulses. The devices are held in the Reset state until both the POR Counter and XTAL counter have timed out. After the Z8 Encore! XP F64xx Series devices exit the Power-On Reset state, the eZ8 CPU fetches the Reset vector. Following Power-On Reset, the POR status bit in the Watchdog Timer Control (WDTCTL) Register is set to 1.

Figure 8 displays Power-On Reset operation. For the POR threshold voltage (V_{POR}), see the Electrical Characteristics chapter on page 200.

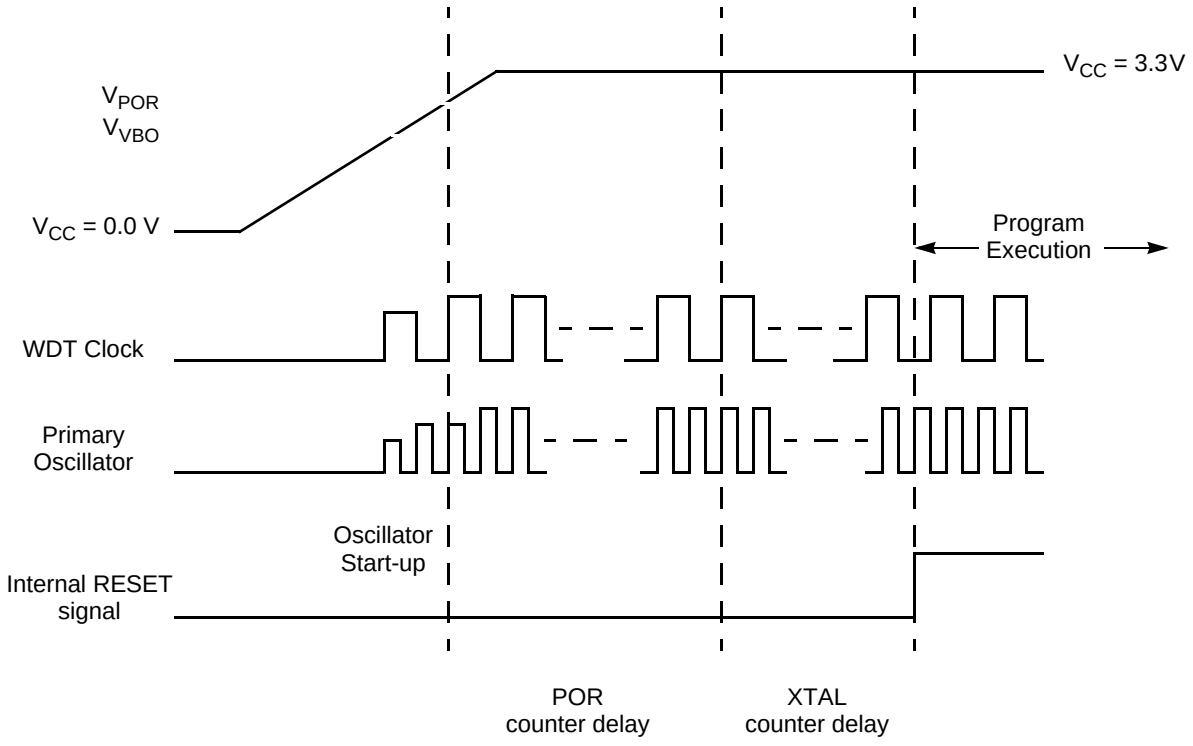


Figure 8. Power-On Reset Operation

Port A–H Input Data Registers

Reading from the Port A–H Input Data registers, shown in Table 21, returns the sampled values from the corresponding port pins. The Port A–H Input Data registers are read-only.

Table 21. Port A–H Input Data Registers (PxIN)

Bit	7	6	5	4	3	2	1	0
Field	PIN7	PIN6	PIN5	PIN4	PIN3	PIN2	PIN1	PIN0
RESET	X							
R/W	R							
Address	FD2H, FD6H, FDAH, FDEH, FE2H, FE6H, FEAH, FEEH							

Bit	Description
[7:0] PxIN	**Port Input Data** Sampled data from the corresponding port pin input. 0 = Input data is logical 0 (Low). 1 = Input data is logical 1 (High).

Note: x indicates register bits in the range [7:0].

Port A–H Output Data Register

The Port A–H Output Data Register, shown in Table 22, writes output data to the pins.

Table 22. Port A–H Output Data Register (P*x*OUT)

Bit	7	6	5	4	3	2	1	0
Field	POUT7	POUT6	POUT5	POUT4	POUT3	POUT2	POUT1	POUT0
RESET	0							
R/W	R/W							
Address	FD3H, FD7H, FDBH, FDFH, FE3H, FE7H, FEBH, FEFH							

Bit	Description
[7:0] PxOUT	**Port Output Data** These bits contain the data to be driven out from the port pins. The values are only driven if the corresponding pin is configured as an output and the pin is not configured for alternate function operation. 0 = Drive a logical 0 (Low). 1 = Drive a logical 1 (High). High value is not driven if the drain has been disabled by setting the corresponding Port Output Control Register bit to 1.

Note: x indicates register bits in the range [7:0].

Bit	Description (Continued)
[2] TDRE	**Transmitter Data Register Empty** This bit indicates that the UART Transmit Data Register is empty and ready for additional data. Writing to the UART Transmit Data Register resets this bit. 0 = Do not write to the UART Transmit Data Register. 1 = The UART Transmit Data Register is ready to receive an additional byte to be transmitted.
[1] TXE	**Transmitter Empty** This bit indicates that the Transmit Shift Register is empty and character transmission is finished. 0 = Data is currently transmitting. 1 = Transmission is complete.
[0] CTS	**$\overline{\text{CTS}}$ Signal** When this bit is read, it returns the level of the $\overline{\text{CTS}}$ signal.

UART Status 1 Register

The UART Status 1 Register, shown in Table 56, contains multiprocessor control and UART status bits.

Table 56. UART Status 1 Register (U*x*STAT1)

Bit	7	6	5	4	3	2	1	0
Field	Reserved						NEWFRM	MPRX
RESET	0							
R/W	R				R/W		R	
Address	F44H and F4CH							

Bit	Description
[7:2]	**Reserved** These bits are reserved and must be programmed to 000000.
[1] NEWFRM	**New Frame** Status bit denoting the start of a new frame. Reading the UART Receive Data Register resets this bit to 0. 0 = The current byte is not the first data byte of a new frame. 1 = The current byte is the first data byte of a new frame.
[0] MPRX	**Multiprocessor Receive** Returns the value of the last multiprocessor bit received. Reading from the UART Receive Data Register resets this bit to 0.

Table 62. UART Baud Rates

20.0 MHz System Clock				18.432 MHz System Clock			
Desired Rate (kHz)	BRG Divisor (Decimal)	Actual Rate (kHz)	Error (%)	Desired Rate (kHz)	BRG Divisor (Decimal)	Actual Rate (kHz)	Error (%)
1250.0	1	1250.0	0.00	1250.0	1	1152.0	–7.84%
625.0	2	625.0	0.00	625.0	2	576.0	–7.84%
250.0	5	250.0	0.00	250.0	5	230.4	–7.84%
115.2	11	113.6	–1.36	115.2	10	115.2	0.00
57.6	22	56.8	–1.36	57.6	20	57.6	0.00
38.4	33	37.9	–1.36	38.4	30	38.4	0.00
19.2	65	19.2	0.16	19.2	60	19.2	0.00
9.60	130	9.62	0.16	9.60	120	9.60	0.00
4.80	260	4.81	0.16	4.80	240	4.80	0.00
2.40	521	2.40	–0.03	2.40	480	2.40	0.00
1.20	1042	1.20	–0.03	1.20	960	1.20	0.00
0.60	2083	0.60	0.02	0.60	1920	0.60	0.00
0.30	4167	0.30	–0.01	0.30	3840	0.30	0.00
16.667 MHz System Clock				**11.0592 MHz System Clock**			
Desired Rate (kHz)	BRG Divisor (Decimal)	Actual Rate (kHz)	Error (%)	Desired Rate (kHz)	BRG Divisor (Decimal)	Actual Rate (kHz)	Error (%)
1250.0	1	1041.69	–16.67	1250.0	N/A	N/A	N/A
625.0	2	520.8	–16.67	625.0	1	691.2	10.59
250.0	4	260.4	4.17	250.0	3	230.4	–7.84
115.2	9	115.7	0.47	115.2	6	115.2	0.00
57.6	18	57.87	0.47	57.6	12	57.6	0.00
38.4	27	38.6	0.47	38.4	18	38.4	0.00
19.2	54	19.3	0.47	19.2	36	19.2	0.00
9.60	109	9.56	–0.45	9.60	72	9.60	0.00
4.80	217	4.80	–0.83	4.80	144	4.80	0.00
2.40	434	2.40	0.01	2.40	288	2.40	0.00
1.20	868	1.20	0.01	1.20	576	1.20	0.00
0.60	1736	0.60	0.01	0.60	1152	0.60	0.00
0.30	3472	0.30	0.01	0.30	2304	0.30	0.00

since the previous pulse was detected). This gives the endec a sampling window of minus four baud rate clocks to plus eight baud rate clocks around the expected time of an incoming pulse. If an incoming pulse is detected inside this window this process is repeated. If the incoming data is a logical 1 (no pulse), the endec returns to the initial state and waits for the next falling edge. As each falling edge is detected, the endec clock counter is reset, resynchronizing the endec to the incoming signal. This action allows the endec to tolerate jitter and baud rate errors in the incoming data stream. Resynchronizing the endec does not alter the operation of the UART, which ultimately receives the data. The UART is only synchronized to the incoming data stream when a start bit is received.

Infrared Encoder/Decoder Control Register Definitions

All infrared endec configuration and status information is set by the UART control registers as defined in the UART Control Register Definitions section on page 98.

Caution: To prevent spurious signals during IrDA data transmission, set the IREN bit in the UARTx Control 1 Register to 1 to enable the Infrared Encoder/Decoder before enabling the GPIO Port alternate function for the corresponding pin.

Bit	Description (Continued)
[5] COL	**Collision** 0 = A multimaster collision (mode fault) has not occurred. 1 = A multimaster collision (mode fault) has been detected.
[4] ABT	**Slave Mode Transaction Abort** This bit is set if the SPI is configured in slave mode, a transaction is occurring and $\overline{SS}$ deasserts before all bits of a character have been transferred as defined by the NUMBITS field of the SPIMODE Register. The IRQ bit also sets, indicating the transaction has completed. 0 = A slave mode transaction abort has not occurred. 1 = A slave mode transaction abort has been detected.
[3:2]	**Reserved** These bits are reserved and must be programmed to 00.
[1] TXST	**Transmit Status** 0 = No data transmission currently in progress. 1 = Data transmission currently in progress.
[0] SLAS	**Slave Select** If SPI enabled as a Slave, then the following conditions are true: 0 = $\overline{SS}$ input pin is asserted (Low). 1 = $\overline{SS}$ input is not asserted (High). If SPI enabled as a Master, this bit is not applicable.

Bit	Description (Continued)
[5] ACK	**Acknowledge** This bit indicates the status of the Acknowledge for the last byte transmitted or received. When set, this bit indicates that an Acknowledge occurred for the last byte transmitted or received. This bit is cleared when IEN = 0 or when a Not Acknowledge occurred for the last byte transmitted or received. It is not reset at the beginning of each transaction and is not reset when this register is read. **Caution:** When making decisions based on this bit within a transaction, software cannot determine when the bit is updated by hardware. In the case of write transactions, the I^2C pauses at the beginning of the Acknowledge cycle if the next transmit data or address byte has not been written (TDRE = 1) and stop and start = 0. In this case the ACK bit is not updated until the transmit interrupt is serviced and the Acknowledge cycle for the previous byte completes. For examples of how the ACK bit can be used, see the Address Only Transaction with a 7-bit Address section on page 133 and the Address Only Transaction with a 10-bit Address section on page 135.
[4] 10B	**10-Bit Address** This bit indicates whether a 10- or 7-bit address is being transmitted. After the start bit is set, if the five most significant bits of the address are 11110B, this bit is set. When set, it is reset once the first byte of the address has been sent.
[3] RD	**Read** This bit indicates the direction of transfer of the data. It is active High during a read. The status of this bit is determined by the least significant bit of the I^2C Shift Register after the start bit is set.
[2] TAS	**Transmit Address State** This bit is active High while the address is being shifted out of the I^2C Shift Register.
[1] DSS	**Data Shift State** This bit is active High while data is being shifted to or from the I^2C Shift Register.
[0] NCKI	**NACK Interrupt** This bit is set High when a Not Acknowledge condition is received or sent and neither the start nor the stop bit is active. When set, this bit generates an interrupt that can only be cleared by setting the start or stop bit, allowing you to specify whether to perform a stop or a repeated start.

On-Chip Debugger

The Z8 Encore! XP F64xx Series products contain an integrated On-Chip Debugger (OCD) that provides advanced debugging features including:

- Reading and writing of the Register File
- Reading and writing of Program and Data memory
- Setting of breakpoints
- Execution of eZ8 CPU instructions

Architecture

The On-Chip Debugger consists of four primary functional blocks: transmitter, receiver, autobaud generator, and debug controller. Figure 36 displays the architecture of the On-Chip Debugger.

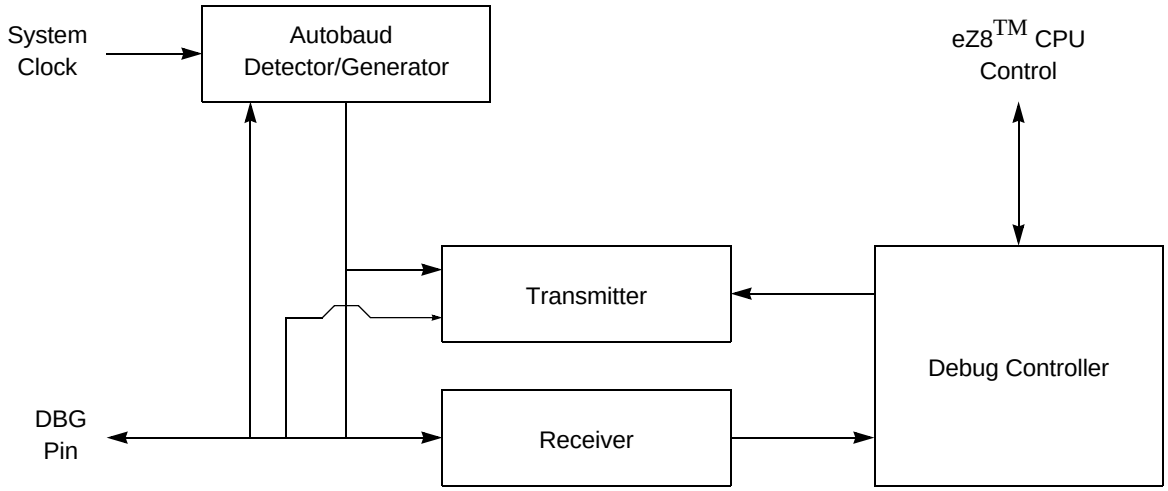


Figure 36. On-Chip Debugger Block Diagram

Table 102. On-Chip Debugger Commands

Debug Command	Command Byte	Enabled when NOT in DEBUG Mode?	Disabled by Read Protect Option Bit
Read OCD Revision	00H	Yes	—
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	—	Disabled
Read Data Memory	0DH	—	Disabled
Read Program Memory CRC	0EH	—	—
Reserved	0FH	—	—
Step Instruction	10H	—	Disabled
Stuff Instruction	11H	—	Disabled
Execute Instruction	12H	—	Disabled
Reserved	13H–FFH	—	—

In the following list of OCD commands, data and commands sent from the host to the On-Chip Debugger are identified by DBG ← Command/Data. Data sent from the On-Chip Debugger back to the host is identified by DBG → Data.

Read OCD Revision (00H). The Read OCD Revision command determines the version of the On-Chip Debugger. If OCD commands are added, removed, or changed, this revision number changes.

On-Chip Debugger Control Register Definitions

This section describes the features of the On-Chip Debugger Control and Status registers.

OCD Control Register

The OCD Control Register, shown in Table 103, controls the state of the On-Chip Debugger. This register enters or exits DEBUG Mode and enables the BRK instruction.

A *reset and stop* function can be achieved by writing 81H to this register. A *reset and go* function can be achieved by writing 41H to this register. If the device is operating in DEBUG Mode, a *run* function can be implemented by writing 40H to this register.

Table 103. OCD Control Register (OCDCTL)

Bit	7	6	5	4	3	2	1	0
Field	DBGMODE	BRKEN	DBGACK	BRKLOOP	Reserved			RST
RESET	0							
R/W	R/W			R				R/W

Bit	Description
[7] DBGMODE	**DEBUG Mode** Setting this bit to 1 causes the device to enter DEBUG Mode. When in DEBUG Mode, the eZ8 CPU stops fetching new instructions. Clearing this bit causes the eZ8 CPU to start running again. This bit is automatically set when a BRK instruction is decoded and breakpoints are enabled. If the Read Protect option bit is enabled, this bit can only be cleared by resetting the device, it cannot be written to 0. 0 = TheZ8 Encore! XP F64xx Series device is operating in NORMAL Mode. 1 = The Z8 Encore! XP F64xx Series device is in DEBUG Mode.
[6] BRKEN	**Breakpoint Enable** This bit controls the behavior of the BRK instruction (op code 00H). By default, breakpoints are disabled and the BRK instruction behaves like a NOP. If this bit is set to 1 and a BRK instruction is decoded, the OCD takes action dependent upon the BRKLOOP bit. 0 = BRK instruction is disabled. 1 = BRK instruction is enabled.
[5] DBGACK	**Debug Acknowledge** This bit enables the debug acknowledge feature. If this bit is set to 1, then the OCD sends an Debug Acknowledge character (FFH) to the host when a breakpoint occurs. 0 = Debug Acknowledge is disabled. 1 = Debug Acknowledge is enabled.

Figure 57 and Table 122 provide timing information for UART pins for the case where the Clear To Send input signal ($\overline{CTS}$) is not used for flow control. In this example, it is assumed that the Driver Enable polarity has been configured to be Active Low and is represented here by $\overline{DE}$. $\overline{DE}$ asserts after the UART Transmit Data Register has been written. $\overline{DE}$ remains asserted for multiple characters as long as the Transmit Data Register is written with the next character before the current character has completed.

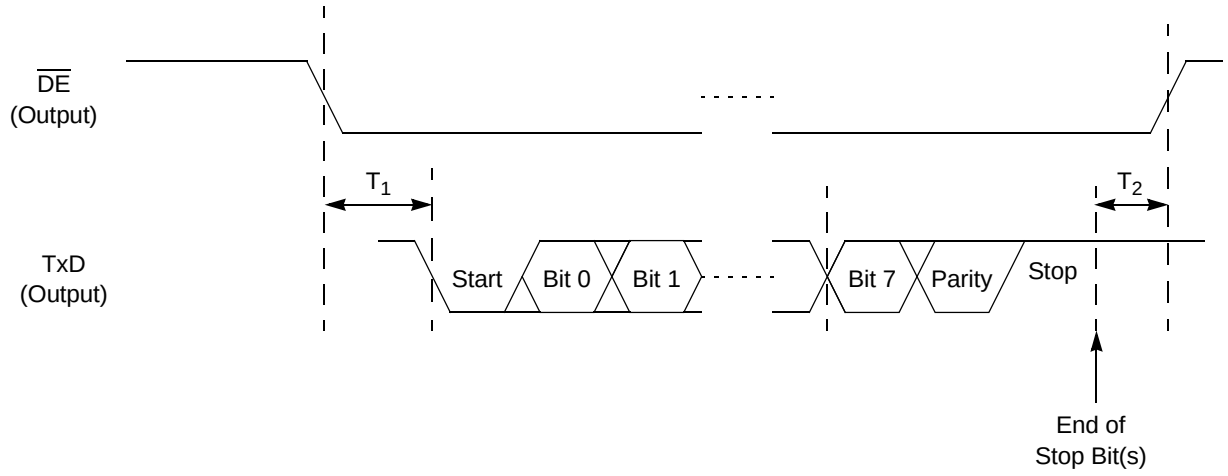


Figure 57. UART Timing without $\overline{CTS}$

Table 122. UART Timing without $\overline{CTS}$

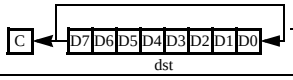
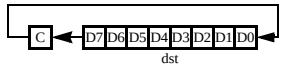
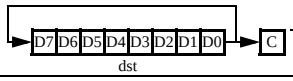
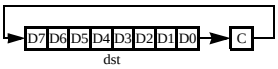
		Delay (ns)	
Parameter	**Abbreviation**	**Minimum**	**Maximum**
T_1	$\overline{DE}$ Assertion to TxD Falling Edge (Start) Delay	1 bit period	1 bit period + 1 * X_{IN} period
T_2	End of stop bit(s) to $\overline{DE}$ Deassertion Delay	1 * X_{IN} period	2 * X_{IN} period

Table 136. 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
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.

Table 136. 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
POPX dst	dst ← @SP SP ← SP + 1	ER		D8	–	–	–	–	–	–	3	2
PUSH src	SP ← SP – 1 @SP ← src	R		70	–	–	–	–	–	–	2	2
		IR		71							2	3
		IM		1F 70							3	2
PUSHX src	SP ← SP – 1 @SP ← src	ER		C8	–	–	–	–	–	–	3	2
RCF	C ← 0			CF	0	–	–	–	–	–	1	2
RET	PC ← @SP SP ← SP + 2			AF	–	–	–	–	–	–	1	4
RL dst		R		90	*	*	*	*	–	–	2	2
		IR		91							2	3
RLC dst		R		10	*	*	*	*	–	–	2	2
		IR		11							2	3
RR dst		R		E0	*	*	*	*	–	–	2	2
		IR		E1							2	3
RRC dst		R		C0	*	*	*	*	–	–	2	2
		IR		C1							2	3
SBC dst, src	dst ← dst – src – C	r	r	32	*	*	*	*	1	*	2	3
		r	Ir	33							2	4
		R	R	34							3	3
		R	IR	35							3	4
		R	IM	36							3	3
		IR	IM	37							3	4
SBCX dst, src	dst ← dst – src – C	ER	ER	38	*	*	*	*	1	*	4	3
		ER	IM	39							4	3
SCF	C ← 1			DF	1	–	–	–	–	–	1	2

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

Hex Address: F47

Table 178. UART Baud Rate Low Byte Register (U*x*BRL)

Bit7	7	6	5	4	3	2	1	0
Field	BRL							
RESET	1							
R/W	R/W							
Address	F47H and F4FH							

Hex Address: F48

Table 179. UART Transmit Data Register (U*x*TXD)

Bit	7	6	5	4	3	2	1	0
Field	TXD							
RESET	X							
R/W	W							
Address	F40H and F48H							

Table 180. UART Receive Data Register (U*x*RXD)

Bit	7	6	5	4	3	2	1	0
Field	RXD							
RESET	X							
R/W	R							
Address	F40H and F48H							

Hex Address: F49

Table 181. UART Status 0 Register (U*x*STAT0)

Bit	7	6	5	4	3	2	1	0
Field	RDA	PE	OE	FE	BRKD	TDRE	TXE	CTS
RESET	0						1	X
R/W	R							
Address	F41H and F49H							

General-Purpose Input/Output (GPIO)

For more information about these GPIO Control registers, see the GPIO Control Register Definitions section on page 39.

Hex Address: FD0

Table 229. Port A–H GPIO Address Registers (P*x*ADDR)

Bit	7	6	5	4	3	2	1	0
Field	PADDR[7:0]							
RESET	00H							
R/W	R/W							
Address	FD0H, FD4H, FD8H, FDCH, FE0H, FE4H, FE8H, FECH							

Hex Address: FD1

Table 230. Port A–H Control Registers (P*x*CTL)

Bit	7	6	5	4	3	2	1	0
Field	PCTL							
RESET	00H							
R/W	R/W							
Address	FD1H, FD5H, FD9H, FDDH, FE1H, FE5H, FE9H, FEDH							

Hex Address: FD2

Table 231. Port A–H Input Data Registers (PxIN)

Bit	7	6	5	4	3	2	1	0
Field	PIN7	PIN6	PIN5	PIN4	PIN3	PIN2	PIN1	PIN0
RESET	X							
R/W	R							
Address	FD2H, FD6H, FDAH, FDEH, FE2H, FE6H, FEAH, FEEH							

Index

Numerics

A

B

C