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	Discontinued at Digi-Key
Core Processor	eZ80
Core Size	8-Bit
Speed	50MHz
Connectivity	EBI/EMI, Ethernet MAC, I²C, IrDA, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	32
Program Memory Size	256KB (256K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	16K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 3.6V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	144-LQFP
Supplier Device Package	144-LQFP (20x20)
Purchase URL	https://www.e-xfl.com/product-detail/zilog/ez80f91gaz0aeg

- 144-pin LQFP and BGA packages
- 3.0V–3.6V supply voltage with 5V tolerant inputs
- Operating Temperature Range:
  - Standard: 0ºC to +70ºC
  - Extended: –40ºC to +105ºC

Note: All signals with an $\overline{\text{overline}}$ are active Low. For example, the signal $\overline{\text{DCD1}}$ is active when it is a logic 0 (Low) state.

Power connections follow these conventional descriptions:

Connection	Circuit	Device
Power	V_{CC}	V_{DD}
Ground	GND	V_{SS}

Block Diagram

Figure 1 shows a block diagram of the eZ80F91 ASSP device.

Table 2. Pin Identification on the eZ80F91 ASSP Device (Continued)

LQFP Pin No	BGA Pin No	Symbol	Function	Signal Direction	Description
120	F7	PA6	GPIO Port A	Bidirectional	This pin is used for GPIO. It is individually programmed as input or output and is also used individually as an interrupt input. Each Port A pin, when programmed as output is selected to be an open-drain or open-source output.
		PWM2	PWM Output 2 Inverted	Output	This pin is used by Timer 3 for negative PWM 2. This signal is multiplexed with PA6.
		EC1	Event Counter	Input	Event Counter Signal to Timer 2. This signal is multiplexed with PA6.
121	A8	PA7	GPIO Port A	Bidirectional	This pin is used for GPIO. It is individually programmed as input or output and is also used individually as an interrupt input. Each Port A pin, when programmed as output is selected to be an open-drain or open-source output.
		PWM3	PWM Output 3 Inverted	Output	This pin is used by Timer 3 for negative PWM 3. This signal is multiplexed with PA7.
122	B8	V_{DD}	Power Supply		Power Supply.
123	C8	V_{SS}	Ground		Ground.
124	D8	CRS	MII Carrier Sense	Input	This pin is used by the EMAC for the MII Interface to the PHY (physical layer). Carrier Sense is an asynchronous signal.
125	A7	COL	MII Collision Detect	Input	This pin is used by the EMAC for the MII Interface to the PHY. Collision Detect is an asynchronous signal.
126	B7	TxD3	MII Transmit Data	Output	This pin is used by the EMAC for the MII Interface to the PHY. Transmit Data is synchronous to the rising-edge of Tx_CLK.

Table 3. Register Map (Continued)

Address (hex)	Mnemonic	Name	Reset (hex)	CPU Access	Page No
0040	EMAC_PTMR	EMAC Transmit Polling Timer Register	00	R/W	320
0041	EMAC_RST	EMAC Reset Control Register	20	R/W	321
0042	EMAC_TLBP_L	EMAC Transmit Lower Boundary Pointer Low Byte	00	R/W	322
0043	EMAC_TLBP_H	EMAC Transmit Lower Boundary Pointer High Byte	00	R/W	322
0044	EMAC_BP_L	EMAC Boundary Pointer Low Byte	00	R/W	323
0045	EMAC_BP_H	EMAC Boundary Pointer High Byte	C0	R/W	323
0046	EMAC_BP_U	EMAC Boundary Pointer Upper Byte	FF	R/W	323
0047	EMAC_RHBP_L	EMAC Receive High Boundary Pointer Low Byte	00	R/W	324
0048	EMAC_RHBP_H	EMAC Receive High Boundary Pointer High Byte	00	R/W	325
0049	EMAC_RRP_L	EMAC Receive Read Pointer Low Byte	00	R/W	325
004A	EMAC_RRP_H	EMAC Receive Read Pointer High Byte	00	R/W	326
004B	EMAC_BUFSZ	EMAC Buffer Size Register	00	R/W	326
004C	EMAC_IEN	EMAC Interrupt Enable Register	00	R/W	327
004D	EMAC_ISTAT	EMAC Interrupt Status Register	00	R/W	329
004E	EMAC_PRSD_L	EMAC PHY Read Status Data Low Byte	00	R/W	330
004F	EMAC_PRSD_H	EMAC PHY Read Status Data High Byte	00	R/W	331
0050	EMAC_MIISTAT	EMAC MII Status Register	00	R/W	331
0051	EMAC_RWP_L	EMAC Receive Write Pointer Low Byte	00	R/W	332
0052	EMAC_RWP_H	EMAC Receive Write Pointer High Byte	00	R/W	333
Ethernet Media Access Controller, continued					
0053	EMAC_TRP_L	EMAC Transmit Read Pointer Low Byte	00	R/W	333
0054	EMAC_TRP_H	EMAC Transmit Read Pointer High Byte	00	R/W	334
0055	EMAC_BLKSLFT_L	EMAC Receive Blocks Left Low Byte Register	20	R/W	334
0056	EMAC_BLKSLFT_H	EMAC Receive Blocks Left High Byte Register	00	R/W	335
0057	EMAC_FDATA_L	EMAC FIFO Data Low Byte	XX	R/W	336

Table 3. Register Map (Continued)

Address (hex)	Mnemonic	Name	Reset (hex)	CPU Access	Page No
007F	PWM1R_H	PWM 1 Rising-Edge High Byte Register	XX	R/W	157
	TMR3_CAPB_H	Timer 3 Capture Value B High Byte Register	XX	R/W	142
0080	PWM2R_L	PWM 2 Rising-Edge Low Byte Register	XX	R/W	157
	TMR3_OC_CTL1	Timer 3 Output Compare Control Register 1	00	R/W	132
0081	PWM2R_H	PWM 2 Rising-Edge High Byte Register	XX	R/W	157
	TMR3_OC_CTL2	Timer 3 Output Compare Control Register 2	00	R/W	132
0082	PWM3R_L	PWM 3 Rising-Edge Low Byte Register	XX	R/W	157
	TMR3_OC0_L	Timer 3 Output Compare 0 Value Low Byte Register	XX	R/W	144
0083	PWM3R_H	PWM 3 Rising-Edge High Byte Register	XX	R/W	157
	TMR3_OC0_H	Timer 3 Output Compare 0 Value High Byte Register	XX	R/W	145
0084	PWM0F_L	PWM 0 Falling-Edge Low Byte Register	XX	R/W	158
	TMR3_OC1_L	Timer 3 Output Compare 1 Value Low Byte Register	XX	R/W	144
0085	PWM0F_H	PWM 0 Falling-Edge High Byte Register	XX	R/W	158
	TMR3_OC1_H	Timer 3 Output Compare 1 Value High Byte Register	XX	R/W	145
0086	PWM1F_L	PWM 1 Falling-Edge Low Byte Register	XX	R/W	158
	TMR3_OC2_L	Timer 3 Output Compare 2 Value Low Byte Register	XX	R/W	144
0087	PWM1F_H	PWM 1 Falling-Edge High Byte Register	XX	R/W	158
	TMR3_OC2_H	Timer 3 Output Compare 2 Value High Byte Register	XX	R/W	145
0088	PWM2F_L	PWM 2 Falling-Edge Low Byte Register	XX	R/W	158
	TMR3_OC3_L	Timer 3 Output Compare 3 Value Low Byte Register	XX	R/W	144

Table 3. Register Map (Continued)

Address (hex)	Mnemonic	Name	Reset (hex)	CPU Access	Page No
Universal Asynchronous Receiver/Transmitter 0 (UART0)					
00C0	UART0_RBR	UART 0 Receive Buffer Register	XX	R	184
	UART0_THR	UART 0 Transmit Holding Register	XX	W	184
	UART0_BRG_L	UART 0 Baud Rate Generator Low Byte Register	02	R/W	182
00C1	UART0_IER	UART 0 Interrupt Enable Register	00	R/W	185
	UART0_BRG_H	UART 0 Baud Rate Generator High Byte Register	00	R/W	183
00C2	UART0_IIR	UART 0 Interrupt Identification Register	01	R	186
	UART0_FCTL	UART 0 FIFO Control Register	00	W	187
00C3	UART0_LCTL	UART 0 Line Control Register	00	R/W	188
00C4	UART0_MCTL	UART 0 Modem Control Register	00	R/W	191
00C5	UART0_LSR	UART 0 Line Status Register	60	R	192
00C6	UART0_MSR	UART 0 Modem Status Register	XX	R	194
00C7	UART0_SPR	UART 0 Scratch Pad Register	00	R/W	195
I^{2}C					
00C8	I2C_SAR	I^{2}C Slave Address Register	00	R/W	226
00C9	I2C_XSAR	I^{2}C Extended Slave Address Register	00	R/W	227
00CA	I2C_DR	I^{2}C Data Register	00	R/W	227
00CB	I2C_CTL	I^{2}C Control Register	00	R/W	228
General-Purpose Input/Output Ports					
00CE	PC_ALT0	Port C Alternate Register 0	00	W	56
00CF	PD_ALT0	Port D Alternate Register 0	00	W	56
00CC	I2C_SR	I^{2}C Status Register	F8	R	230
	I2C_CCR	I^{2}C Clock Control Register	00	W	232
00CD	I2C_SRR	I^{2}C Software Reset Register	XX	W	233

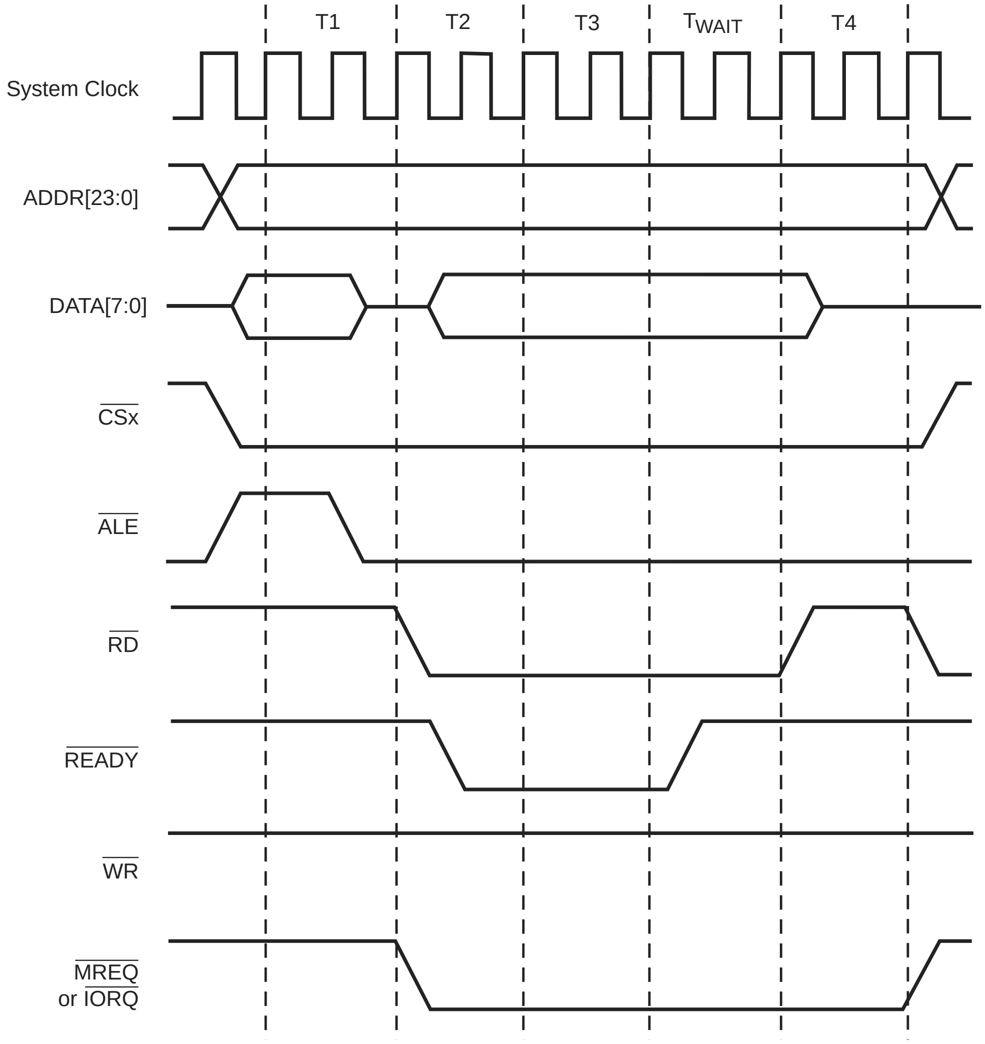

Figure 15. Example: Intel Bus Mode Read Timing: Multiplexed Address and Data Bus

Bit	Description (Continued)
[2] RP_TMO	**Row Program Time-Out Error Flag** 0: Flag is not set. 1: Flag is set.
[1] PG_VIO	**Page Erase Violation Error Flag** 0: The page erase violation error flag is not set. 1: The page erase violation error flag is set.
[0] MASS_VIO	**Mass Erase Violation Error Flag** 0: The mass erase violation error flag is not set. 1: The mass erase violation error flag is set.

Note: The lower 32KB block (00000h to 07FFFh) is called the boot block and is protected using the external $\overline{\text{WP}}$ pin. Attempts to page erase BLK0 or mass erase Flash when $\overline{\text{WP}}$ is asserted result in failure and signal an erase violation.

Flash Page Select Register

The msb of this register is used to select whether I/O Flash access and page erase operations are directed to the 512-byte information page or to the main Flash memory array, and also whether the information page is included in mass erase operations. The lower 7 bits are used to select one of the main 128 pages for page erase or I/O operations.

To perform a page erase, the software must set the proper page value prior to setting the page erase bit in the Flash Control Register. In addition, each access to the FLASH_DATA Register causes an autoincrement of the Flash address stored in the Flash Address registers (FLASH_PAGE, FLASH_ROW, FLASH_COL). See Table 43.

Table 43. Flash Page Select Register (FLASH_PAGE)

Bit	7	6	5	4	3	2	1	0
Field	INFO_EN	FLASH_PAGE						
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	00FCh							

Note: R/W = read/write, R = read only.

Bit	Description
[7] INFO_EN	**Flash I/O Access to Page Erase Operations** 0: Directed to main Flash memory. Info page is not affected by a mass erase operation. 1: Directed to the information page. Page erase operations only affect the information page. Info page is included during a mass erase operation
[6:0] FLASH_PAGE	**Flash Page Address** 00h–7Fh: Page address of Flash memory to be used during a page erase or I/O access of main Flash memory. When INFO_EN is set to 1, this field is ignored.

Real-Time Clock Minutes Register

This register contains the current minutes count. The value in the RTC_MIN Register is unchanged by a RESET. The current setting of BCD_EN determines whether the values in this register are binary (BCD_EN = 0) or binary-coded decimal (BCD_EN = 1). Access to this register is read-only if the RTC is locked, and read/write if the RTC is unlocked. See Table 81.

Table 81. Real-Time Clock Minutes Register (RTC_MIN)

Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Field	TEN_MIN				MIN			
Reset	U	U	U	U	U	U	U	U
R/W	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*
Address	00E1h							

Note: U = unchanged by RESET; R/W* = read only if RTC locked, read/write if RTC unlocked.

Binary-Coded Decimal Operation (BCD_EN = 1)

Bit	**Description**
[7:4] TEN_MIN	**Minutes: Tens** 0–5: The tens digit of the current minutes count.
[3:0] MIN	**Minutes: Ones** 0–9: The ones digit of the current minutes count.

Binary Operation (BCD_EN = 0)

Bit	**Description**
[7:0] MIN	**Minutes** 00h–3Bh: The current minutes count.

Bit	Description (Continued)
[4] TCIE	**Transmission Complete Interrupt** 0: Transmission complete interrupt is disabled 1: Transmission complete interrupt is generated when both the transmit hold register and the transmit shift register are empty
[3] MIIE	**Modem Interrupt Input Enable** 0: Modem interrupt on edge detect of status inputs is disabled. 1: Modem interrupt on edge detect of status inputs is enabled.
[2] LSIE	**Line Status Interrupt Input Enable** 0: Line status interrupt is disabled. 1: Line status interrupt is enabled for receive data errors: incorrect parity bit received, framing error, overrun error, or break detection.
[1] TIE	**Transmit Interrupt Input Enable** 0: Transmit interrupt is disabled. 1: Transmit interrupt is enabled. Interrupt is generated when the transmit FIFO/buffer is empty indicating no more bytes available for transmission.
[0] RIE	**Receive Interrupt Input Enable** 0: Receive interrupt is disabled. 1: Receive interrupt and receiver time-out interrupt are enabled. Interrupt is generated if the FIFO/buffer contains data ready to be read or if the receiver times out.

UART Interrupt Identification Register

The read-only UARTx_IIR Register allows you to check whether the FIFO is enabled and the status of interrupts. These registers share the same I/O addresses as the UARTx_FCTL registers. See Tables 99 and 100.

Table 99. UART Interrupt Identification Registers (UART*x*_IIR)

Bit	7	6	5	4	3	2	1	0
Field	FSTS	Reserved			INSTS			INTBIT
Reset	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R
Address	UART0_IIR = 00C2h, UART1_IIR = 00D2h							
Note: *x* indicates UART[1:0]; R = read only.								

Bit	Description
[7] FSTS	**FIFO Enable** 0: FIFO is disabled. 1: FIFO is enabled.
[6:4]	**Reserved** These bits are reserved and must be programmed to 000.

SPI Transmit Shift Register

The SPI Transmit Shift Register (SPI_TSR) is used by the SPI master to transmit data over an SPI serial bus to a slave device. A write to the SPI_TSR Register places data directly into the shift register for transmission. A write to this register within an SPI device configured as a master initiates transmission of the byte of the data loaded into the register. At the completion of transmitting a byte of data, the SPI Flag (SPI_SR[7]) is set to 1 in both the master and slave devices.

The write-only SPI Transmit Shift Register shares the same address space as the read-only SPI Receive Buffer Register. See Table 116.

Table 116. SPI Transmit Shift Register (SPI_TSR)

Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Field	Tx_DATA							
Reset	U	U	U	U	U	U	U	U
R/W	W	W	W	W	W	W	W	W
Address	00BCh							

Note: U = undefined; W = write only.

Bit	Description
[7:0] Tx_DATA	**SPI Transmit Data** 00h–FFh: SPI transmit data.

Table 121. I^{2}C Master Receive Status Codes

Code	I^{2}C State	ASSP Response	Next I^{2}C Action
78h	Arbitration lost, general call addr received, ACK transmitted	Same as code 68h	Same as code 68h
B0h	Arbitration lost, SLA+R received, ACK transmitted	Write byte to DATA, clear IFLG, clear AAK = 0	Transmit last byte, receive ACK
		Or write byte to DATA, clear IFLG, set AAK = 1	Transmit data byte, receive ACK

Notes:
1. AAK is an I^{2}C control bit that identifies which ACK signal to transmit.
2. R is defined as the read bit; that is, the lsb is set to 1.
3. W is defined as the write bit; that is, the lsb is cleared to 0.

If 10-bit addressing is being used, the slave is first addressed using the full 10-bit address, plus the write bit. The master then issues a restart followed by the first part of the 10-bit address again, this time with the read bit. The status code then becomes `40h` or `48h`. It is the responsibility of the slave to remember that it had been selected prior to the restart.

If a repeated start condition is received, the status code is `10h` instead of `08h`.

After each data byte is received, the IFLG is set to 1 and one of the status codes listed in Table 122 is loaded into the I^{2}C_SR Register.

Table 122. I^{2}C Master Receive Status Codes For Data Bytes

Code	I^{2}C State	ASSP Response	Next I^{2}C Action
50h	Data byte received, ACK transmitted	Read data, clear IFLG, clear AAK = 0*	Receive data byte, transmit NACK
		Or read data, clear IFLG, set AAK = 1	Receive data byte, transmit ACK
58h	Data byte received, NACK transmitted	Read data, set STA, clear IFLG	Transmit repeated start
		Or read data, set STP, clear IFLG	Transmit stop
		Or read data, set STA and STP, clear IFLG	Transmit stop, then start
38h	Arbitration lost in NACK bit	Same as master transmit	Same as master transmit

Note: *AAK is an I^{2}C control bit that identifies which ACK signal to transmit.

Table 152. Pin to Boundary Scan Cell Mapping (Continued)

Pin	Direction	Scan Cell No	Pin	Direction	Scan Cell No
WR	Output	11	PA5	Output	118
WR	OEN	12	PA5	OEN	119
RD	Output	13	PA4	Input	120
MREQ	Input	14	PA4	Output	121
MREQ	Output	15	PA4	OEN	122
IORQ	Input	16	PA3	Input	123
IORQ	Output	17	PA3	Output	124
D7	Input	18	PA3	OEN	125
D7	Output	19	PA2	Input	126
D6	Input	20	PA2	Output	127
D6	Output	21	PA2	OEN	128
D5	Input	22	PA1	Input	129
D5	Output	23	PA1	Output	130
D4	Input	24	PA1	OEN	131
D4	Output	25	PA0	Input	132
D3	Input	26	PA0	Output	133
D3	Output	27	PA0	OEN	134
D2	Input	28	PHI	Output	135
D2	Output	29	PHI	OEN	136
D1	Input	30	SCL	Input	137
D1	Output	31	SCL	Output	138
D0	Input	32	SDA	Input	139
D0	Output	33	SDA	Output	140
D0	OEN	34	PB7	Input	141
CS3	Output	35	PB7	Output	142
CS2	Output	36	PB7	OEN	143

Notes:

1. The address bits 0–7, 8–15, and 16–23 each share a single output enable. In this table, the output enables are associated with the least-significant bit that they control.
2. Direction on the data bus is controlled by a single output enable. It is associated in this table with D[0].
3. MREQ, IORQ, INSTRDN, RD, and WR share an output enable; it is associated in this table with WR.

Table 152. Pin to Boundary Scan Cell Mapping (Continued)

Pin	Direction	Scan Cell No	Pin	Direction	Scan Cell No
CS1	Output	37	PB6	Input	144
CS0	Output	38	PB6	Output	145
A23	Input	39	PB6	OEN	146
A23	Output	40	PB5	Input	147
A22	Input	41	PB5	Output	148
A22	Output	42	PB5	OEN	149
A21	Input	43	PB4	Input	150
A21	Output	44	PB4	Output	151
A20	Input	45	PB4	OEN	152
A20	Output	46	PB3	Input	153
A19	Input	47	PB3	Output	154
A19	Output	48	PB3	OEN	155
A18	Input	49	PB2	Input	156
A18	Output	50	PB2	Output	157
A17	Input	51	PB2	OEN	158
A17	Output	52	PB1	Input	159
A16	Input	53	PB1	Output	160
A16	Output	54	PB1	OEN	161
A16	OEN	55	PB0	Input	162
A15	Input	56	PB0	Output	163
A15	Output	57	PB0	OEN	164
A14	Input	58	PC7	Input	165
A14	Output	59	PC7	Output	166
A13	Input	60	PC7	OEN	167
A13	Output	61	PC6	Input	168
A12	Input	62	PC6	Output	169

Notes:

1. The address bits 0–7, 8–15, and 16–23 each share a single output enable. In this table, the output enables are associated with the least-significant bit that they control.
2. Direction on the data bus is controlled by a single output enable. It is associated in this table with D[0].
3. $\overline{\text{MREQ}}$, $\overline{\text{IORQ}}$, $\overline{\text{INSTRDN}}$, $\overline{\text{RD}}$, and $\overline{\text{WR}}$ share an output enable; it is associated in this table with $\overline{\text{WR}}$.

Bit	Description (Continued)
[2] INT_LOCK_EN	**PLL Lock Interrupt Enable** 0: Interrupt generation for PLL locked condition (Bit 4) is disabled. 1: Interrupt generation for PLL locked condition is enabled.
[1] INT_UNLOCK_EN	**PLL Unlock Interrupt Enable** 0: Interrupt generation for PLL unlocked condition (Bit 3) is disabled. 1: Interrupt generation for PLL unlocked condition is enabled.
[0] PLL_ENABLE	**PLL Enable** 0: PLL is disabled.* 1: PLL is enabled.

Note: *PLL cannot be disabled if the CLK_MUX bit of PLL_CTL0[1:0] is set to 01, because the PLL is selected as the clock source.

PLL Characteristics

The operating and testing characteristics for the PLL are described in Table 157.

Table 157. PLL Characteristics

Symbol	Parameter	Test Condition	Min	Typ	Max	Units
I_{OHCP_OUT}	High level output current for CP_OUT pin (programmed value ±42%)	$3.0 < V_{DD} < 3.6$ $0.6 < PD\_OUT < V_{DD} – 0.6$ PLL_CTL0[7:6] = 11	–0.86	–1.50	–2.13	mA
I_{OLCP_OUT}	Low level output current for CP_OUT pin (programmed value ±42%)	$3.0 < V_{DD} <3.6$ $0.6 < PD\_OUT < V_{DD} – 0.6$ PLL_CTL0[7:6] = 11	0.86	1.50	2.13	mA
I_{OHCP_OUT}	High level output current for CP_OUT pin (programmed value ±42%)	$3.0 < V_{DD} <3.6$ $0.6 < PD\_OUT < V_{DD} – 0.6$ PLL_CTL0[7:6] = 10	–0.42	–1.0	–1.42	mA
I_{OLCP_OUT}	Low level output current for CP_OUT pin (programmed value ±42%)	$3.0 < V_{DD} <3.6$ $0.6 < PD\_OUT <V_{DD} – 0.6$ PLL_CTL0[7:6] = 10	0.42	1.0	1.42	mA
I_{OHCP_OUT}	High level output current for CP_OUT pin (programmed value ±42%)	$3.0 < V_{DD} <3.6$ $0.6 < PD\_OUT <V_{DD} – 0.6$ PLL_CTL0[7:6] = 01	–210	–500	–710	µA
I_{OLCP_OUT}	Low level output current for CP_OUT pin (programmed value ±42%)	$3.0 < V_{DD} <3.6$ $0.6 < PD\_OUT <V_{DD} – 0.6$ PLL_CTL0[7:6] = 01	210	500	710	µA
I_{OHCP_OUT}	High level output current for CP_OUT pin (programmed value ±42%)	$3.0 < V_{DD} <3.6$ $0.6 < PD\_OUT <V_{DD} – 0.6$ PLL_CTL0[7:6] = 00	–42	–100	–142	µA

TxFIFO is more than half full. Similarly, the RxDMA offers two levels of priority: a high priority when the RxFIFO is more than half full and a Low priority when the RxFIFO is less than half full.

The arbiter determines resolution between the CPU, the RxDMA, and the TxDMA requests to access EMAC memory. Post writing for CPU writes results in *zero wait state* write access timing when the CPU assumes the highest priority. CPU reads require a minimum of 1 wait state and takes more when the CPU does not hold the highest priority. The CPU read wait state is not a user-controllable operation, because it is controlled by the arbiter. The RxDMA and TxDMA requests are not allowed to occur back-to-back. Therefore, the maximum throughput rate for the two Direct Memory Access (DMA) ports is 25 MBps each (one byte every 2 clocks) when the system clock is running at 50MHz. The rate is reduced to 20 MBps for a 40MHz system clock. The arbiter uses the internal WAIT signal to add wait states to CPU access when required. See Table 175.

Table 175. Arbiter Priority

Priority Level	Device Serviced	Flags
0	RxDMA High	RxFIFO > half full (FAF)
1	TxDMA High	TxFIFO < half full (FAE)
2	eZ80® CPU	
3	RxDMA Low	RxFIFO < half full (FAE)
4	TxDMA Low	TxFIFO > half full (FAF)

TxDMA

The TxDMA module moves the next packet to be transmitted from EMAC memory into the TxFIFO. Whenever the polling timer expires, the TxDMA reads the High status byte from the Tx descriptor table pointed to by the Transmit Read Pointer, TRP. Polling continues until the High status read reaches bit 7, when the Emac_Owns ownership semaphore, bit 15 of the descriptor table (see Table 179) is set to 1. The TxDMA then initializes the packet length counter with the size of the packet from descriptor table bytes 3 and 4. The TxDMA moves the data into the TxFIFO until the packet length counter downcounts to zero. The TxDMA then waits for Transmission Complete signal to be asserted to indicate that the packet is sent and that the Transmit status from the EMAC is valid. The TxDMA updates the descriptor table status and resets the ownership semaphore, bit 15. Finally, the Tx_DONE_STAT bit of the EMAC Interrupt Status Register is set to 1, the address field, DMA_Address, is updated from the descriptor table next pointer, NP (see Figure 62). The high byte of the status is read to determine if the next packet is ready to be transmitted.

While the TxDMA is filling the TxFIFO, it monitors two signals from the Transmit FIFO State Machine (TxFifoSM) to detect error conditions and to determine if the packet is to be retransmitted (TxDMA_Retry asserted) or the packet is aborted (TxDMA_Abort

EMAC Receive Blocks Left High and Low Byte Registers

This register reports the number of buffers left in Receive EMAC shared memory. The hardware uses this information along with the block-level set in the EMAC_BUFSZ Register to determine when to transmit a pause control frame. Software uses this information to determine when it must request that a pause control frame be transmitted (by setting bit 6 of the EMAC_CFG4 Register). For the BlksLeft logic to operate properly, the Receive buffer must contain at least one more packet buffer than the number of packet buffers required for the largest packet. That is, one packet cannot fill the entire Receive buffer. Otherwise, BlksLeft will be in error. See Tables 225 and 226.

Table 225. EMAC Receive Blocks Left Low Byte Register (EMAC_BLKSLFT_L)

Bit	7	6	5	4	3	2	1	0
Field	EMAC_BLKSLFT_L							
Reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R
Address	0055h							

Note: R = read only.

Bit	Description
[7:0] EMAC_BLKSLFT_L	**Receive Blocks Left Low Byte** 00h–FFh: These bits represent the low byte of the two-byte EMAC Receive Blocks Left value, {EMAC_BLKSLFT_H, EMAC_BLKSLFT_L}. Bit 7 is bit 7 of the 16-bit value. Bit 0 is bit 0 (lsb) of the 16-bit value.

Table 226. EMAC Receive Blocks Left High Byte Register (EMAC_BLKSLFT_H)

Bit	7	6	5	4	3	2	1	0
Field	EMAC_BLKSLFT_H							
Reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R
Address	0056h							

Note: R = read only.

Bit	Description
[7:0] EMAC_BLKSLFT_H	**Receive Blocks Left High Byte** 00h–FFh: These bits represent the high byte of the two-byte EMAC Receive Blocks Left value, {EMAC_BLKSLFT_H, EMAC_BLKSLFT_L}. Bit 7 is bit 15 (msb) of the 16-bit value. Bit 0 is bit 8 of the 16-bit value.

EMAC FIFO Flags Register

The FIFO Flags value is set in the EMAC hardware to *half full*, or 16 bytes. See Table 229.

Table 229. EMAC FIFO Flags Register (EMAC_FFLAGS)

Bit	7	6	5	4	3	2	1	0
Field	TFF		TFAE	TFE	RFF	RFAF	RFAE	RFE
Reset	0	0	1	1	0	0	1	1
R/W	R	R	R	R	R	R	R	R
Address	0059h							

Note: R = read only.

Bit	Description
[7] TFF	**Transmit FIFO Full** 0: Transmit FIFO not full. 1: Transmit FIFO full.
[6]	**Reserved** This bit is reserved and must be programmed to 0.
[5] TFAE	**Transmit FIFO Almost Empty** 0: Transmit FIFO not almost empty. 1: Transmit FIFO almost empty.
[4] TFE	**Transmit FIFO Empty** 0: Transmit FIFO not empty. 1: Transmit FIFO empty.
[3] RFF	**Receive FIFO Full** 0: Receive FIFO not full. 1: Receive FIFO full.
[2] RFAF	**Receive FIFO Almost Full** 0: Receive FIFO not almost full. 1: Receive FIFO almost full.
[1] RFAE	**Receive FIFO Almost Empty** 0: Receive FIFO not almost empty. 1: Receive FIFO almost empty.
[0] RFE	**Receive FIFO Empty** 0: Receive FIFO not empty. 1: Receive FIFO empty.

Current Consumption Under Various Operating Conditions

Figure 65 shows the typical current consumption of the eZ80F91 ASSP device versus V_{DD} while operating at 25ºC, with zero wait states, and with either a 10 MHz, 20 MHz, or 50 MHz system clock.

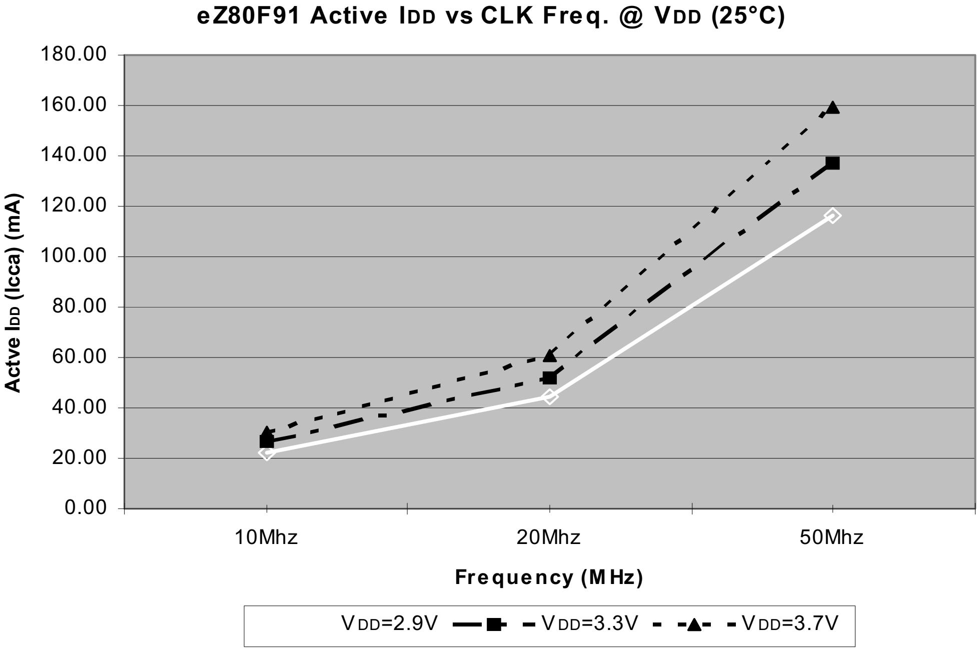

Figure 65. I_{CC} vs. System Clock Frequency During ACTIVE Mode