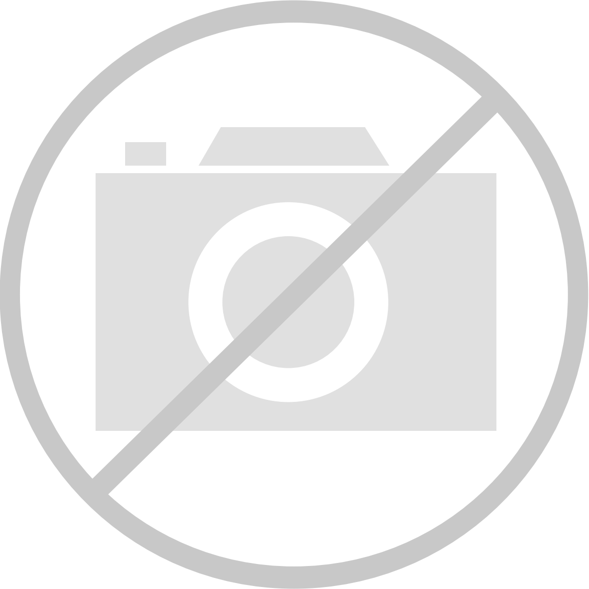
Zilog - EZ80F916050MODG Datasheet



Details	
Product Status	Obsolete
Module/Board Type	MCU, Ethernet Core
Core Processor	eZ80F91
Co-Processor	-
Speed	50MHz
Flash Size	256KB (Internal), 1MB (External)
RAM Size	16KB (Internal), 512KB (External)
Connector Type	Header 2x30
Size / Dimension	2.5" x 3.1" (63.5mm x 78.7mm)
Operating Temperature	0°C ~ 70°C
Purchase URL	https://www.e-xfl.com/product-detail/zilog/ez80f916050modg

eZ80F91 Development Kit Overview

The purpose of the eZ80F91 Development Kit is to provide the developer with a set of tools for evaluating the features of the eZ80F91 microcontroller and to be able to develop a new application before building application hardware.

The eZ80® Development Platform is designed to accept a number of application-specific modules and eZ80®-based add-on modules, including the eZ80F91 Module featured in this kit.

The eZ80® Development Platform, together with its plugged-in eZ80F91 Module, can operate in stand-alone mode with Flash memory, or interface via the ZPAKII Debug Tool to a host PC running ZiLOG Developer Studio II Integrated Development Environment (ZDS IDE) software.

The address bus, data bus, and all eZ80F91 Module control signals are buffered on the eZ80® Development Platform to provide sufficient drive capability.

A block diagram of the eZ80® Development Platform and the eZ80F91 Module is shown in Figure 1.

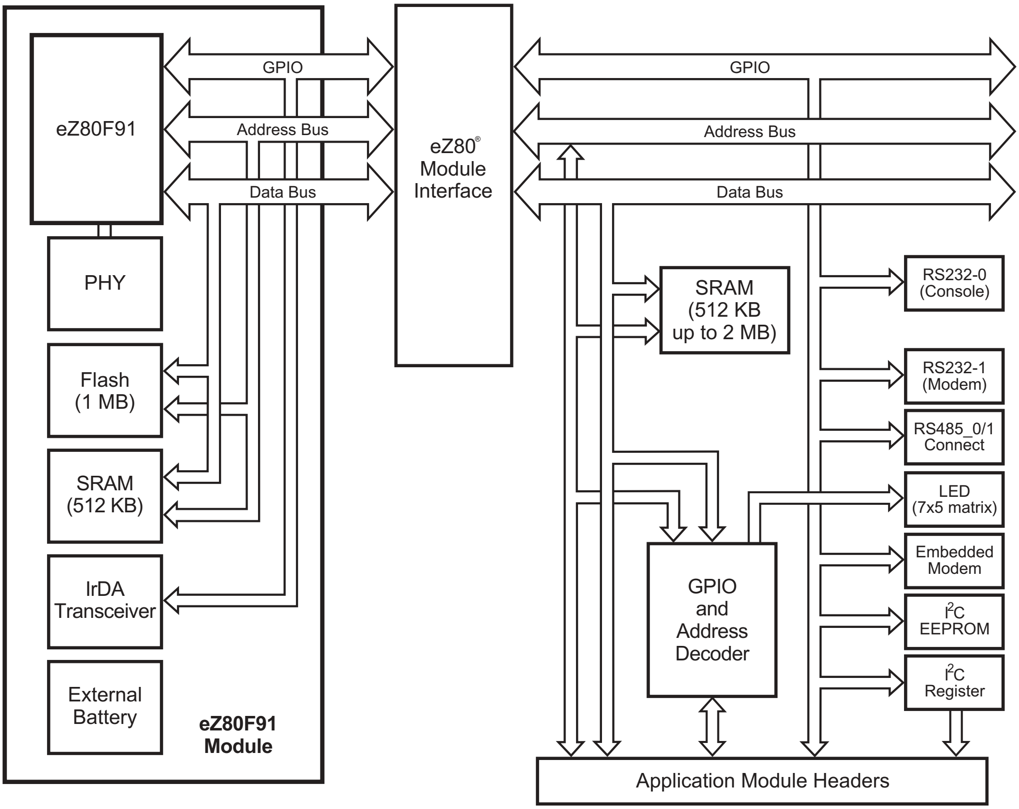


Figure 1. eZ80® Development Platform Block Diagram with eZ80F91 Module

Physical Dimensions

The dimensions of the eZ80® Development Platform PCB is 177.8 mm x 182.9 mm. The overall height is 38.1 mm. See Figure 5.

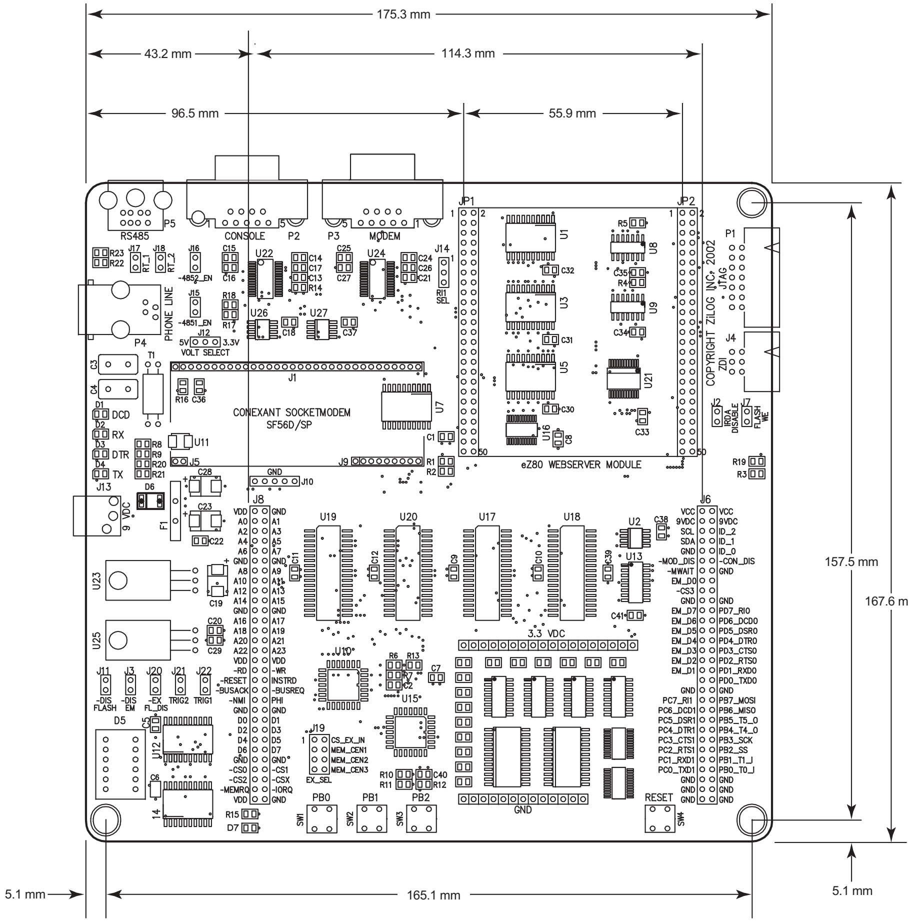


Figure 5. Physical Dimensions of the eZ80® Development Platform

Table 3. eZ80® Development Platform I/O Connector Identification—JP2[1]

Pin #	Symbol	Signal Direction	Active Level	eZ80F91 Signal[2]
1	PB7	Bidirectional		Yes
2	PB6	Bidirectional		Yes
3	PB5	Bidirectional		Yes
4	PB4	Bidirectional		Yes
5	PB3	Bidirectional		Yes
6	PB2	Bidirectional		Yes
7	PB1	Bidirectional		Yes
8	PB0	Bidirectional		Yes
9	GND			
10	PC7	Bidirectional		Yes
11	PC6	Bidirectional		Yes
12	PC5	Bidirectional		Yes
13	PC4	Bidirectional		Yes
14	PC3	Bidirectional		Yes
15	PC2	Bidirectional		Yes
16	PC1	Bidirectional		Yes
17	PC0	Bidirectional		Yes
18	PD7	Bidirectional		Yes

Notes:

1. For the sake of simplicity in describing the interface, Power and Ground nets are omitted from this table. The entire interface is represented in the eZ80F91 Module Schematics on pages 64 through 66.
2. The Power and Ground nets are connected directly to the eZ80F91 device.

Table 4. GPIO Connector J6* (Continued)

Signal	Pin #	Function	Direction	Notes
$\overline{\text{CS3}}$	17	Chip Select 3 of the CPU	OUT	This signal is also present on the J8.
EM_D[7:1]	21,23,25, 27,29,31, 33	Emulated, Bit [7:1]	IN/OUT	
Reserved	35			
PC[7:0]	39,41,43, 45,47,49, 51,53	Port C, Bit [7:0]	IN/OUT	
ID_[2:0]	6,8,10	eZ80® Development Platform ID	OUT	
CON_DIS	12	Console Disable	IN	If a shunt is installed between pins 12 and 14, the Console function on the eZ80® Development Platform is disabled.
Reserved	16,18			
PD[7:0]	22,24,26, 28,30,32, 34,36	Port D, Bit[7:0]	IN/OUT	
PB[7:0]	40,42,44, 46,48,50, 52,54	Port B, Bit[7:0]	IN/OUT	

Note: *All of the signals are driven directly by the CPU.

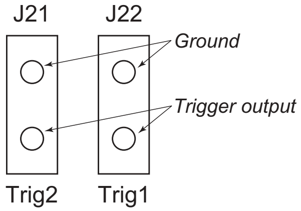


Figure 8. Trigger Pins J21 and J22

Bits 6 and 7 in Table 12 are the control bits for the user triggers. If either bit is a 1, the corresponding Trig1 and Trig2 signals are driven High. If either bit is 0, the corresponding Trig1 and Trig2 signals are driven Low.

Embedded Modem Socket Interface

The eZ80® Development Platform features a socket for an optional 56K modem (a modem is not included in the kit).

Connectors J1, J5, and J9 provide connection capability. The modem socket interface provided by these three connectors is shown in Figure 9. Tables 9 through 11 identify the pins for each connector. The embedded modem utilizes UART1, which is available via the Port C pins.

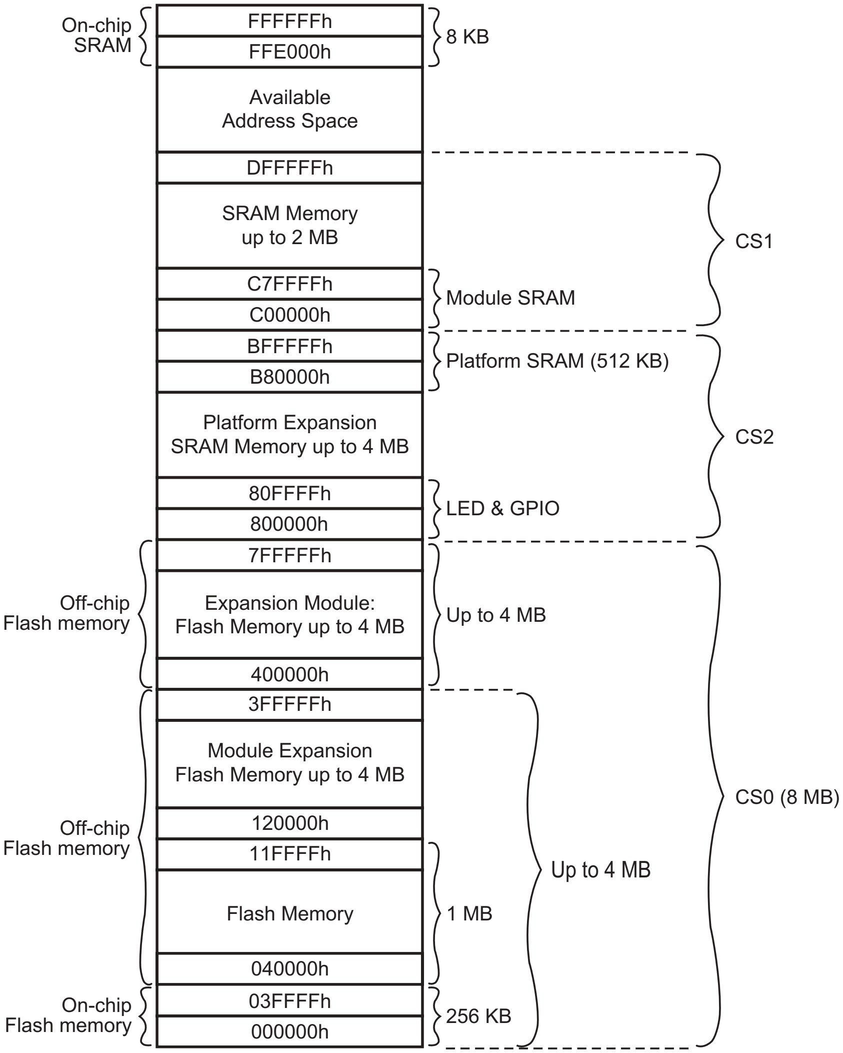


Figure 10. Memory Map of the eZ80® Development Platform and eZ80F91 Module

LEDs

As stated on page 28, LEDs D1, D2, D3, and D4 function as status indicators for an optional modem. This section describes each LED and the LED matrix device.

LED Matrix

The 7x5 LED matrix device on the eZ80® Development Platform is a memory-mapped device that can be used to display information, such as programmed alphanumeric characters. For example, the LED display sample program that is shipped with this kit displays the alphanumeric message:

```
eZ80
```

To illuminate any LED in the matrix, its respective anode bit must be set to 1 and its corresponding cathode bit must be set to 0.

Bits 0–6 in Table 7 are LED anode bits. They must be set High (1) and their corresponding cathode bits, bits 0–4 in Table 12, must be set Low (0) to illuminate each of the LED's, respectively.

If bit 7 in the GPIO Output Control Register is 1, all of the GPIO lines are configured as inputs. If this bit is 0, all of the GPIO lines are configured as outputs.

Table 12 indicates the multiple register functions of the LED cathode, modem, and triggers. This table shows the bit configuration for each cathode bit. Bits 5, 6, and 7 do not carry any significance within the LED matrix. These three bits are control bits for the modem reset, Trig1, and Trig2 functions, respectively.

Jumper J12

The J12 jumper connection controls the selection of a 5V or 3VDC power supply to the embedded modem, if an embedded modem is used. See Table 17.

Table 17. J12—5VDC/3.3VDC for an Embedded Modem

Shunt Status	Function	Affected Device
1–2	5VDC is provided to power the embedded modem.	Embedded modem.
2–3	3.3VDC is provided to power the embedded modem.	Embedded modem.

Jumper J14

The J14 jumper connection controls the polarity of the Ring Indicator. See Table 18.

Table 18. J14—RI

Shunt Status	Function	Affected Device
1–2	The Ring Indicator for UART1 is inverted.	UART1.
2–3	The Ring Indicator for UART1 is not inverted.	UART1.

Connectors

A number of connectors are available for connecting external devices such as the ZPAKII Debug Tool, PC serial ports, external modems, the console, and LAN/telephone lines.

J6 and J8 are the headers, or connectors, that provide pin-outs to connect any external application module, such as ZiLOG's Thermostat Application Module.

Connector J6

The J6 connector provides pin-outs to make use of GPIO functionality.

Connector J8

The J8 connector provides pin-outs to access memory and other control signals.

Console

Connector P2 is the RS232 terminal, which can be used for observing the console output. P2 can be connected to the PC running HyperTerminal if required.

Modem

Connector P3 provides a terminal for connecting an external modem, if used with the eZ80F91 Development Kit.

I^{2}C Devices

The two I^{2}C devices on the eZ80® Development Platform are the U2 EEPROM and the U13 Configuration register. The EEPROM provides 16KB of memory. The Configuration register provides access to control the configuration of an application-specific function at the Application Module Interface. Neither device is utilized by the eZ80F91 Development

Timing diagram in the eZ80F91 Product Specification (PS0192) for assistance). The Flash turn-off time (T_{OD}) is 25 ns—the duration from $\overline{OE}$ or $\overline{CE}$ going High to Flash output drivers in a high-impedance state. For further information, see the MT28F008 data sheet on www.micron.com.

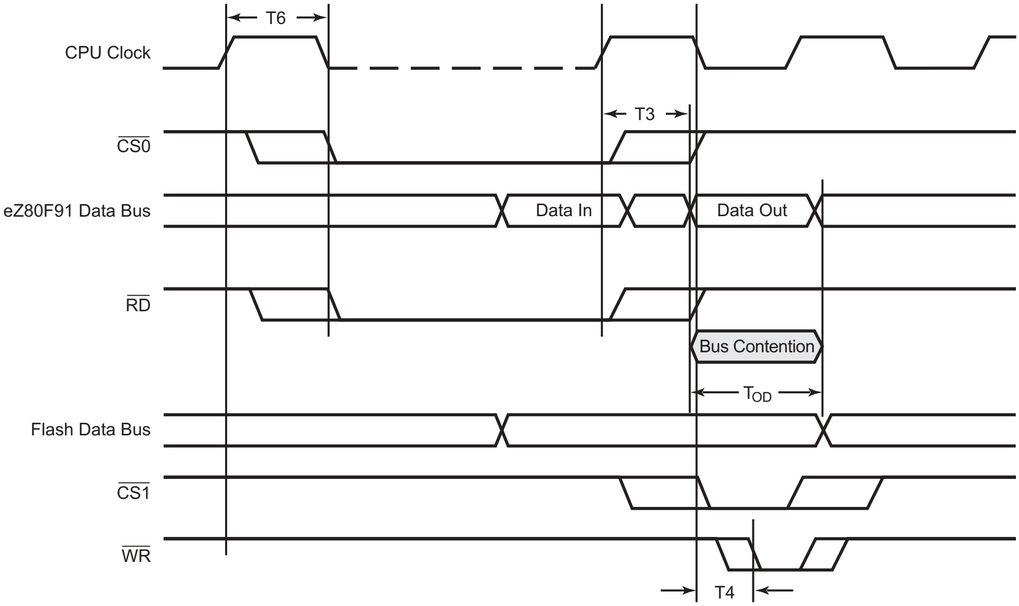


Figure 11. Possible Bus Contention without Fast Buffer

Essentially, after the eZ80F91 device accesses Flash memory, a time duration of 8.8 ns + 25 ns = 33.8 ns can transpire before Flash memory stops driving the data bus. At that time, the eZ80F91 device is well into the next bus cycle. Assuming this next cycle is the Memory Write cycle, then the data output of the eZ80F91 device is valid not later than T3 = 7.5 ns, and the write pulse is asserted not later than 4.5 ns after the falling edge of the CPU Clock (14.5 ns from the rising edge if the CPU Clock is 50 MHz). The duration of bus contention, T_{CON}, is 33.8 ns –

Figure 14 illustrates the bottom layer silkscreen of the eZ80F91 Module.

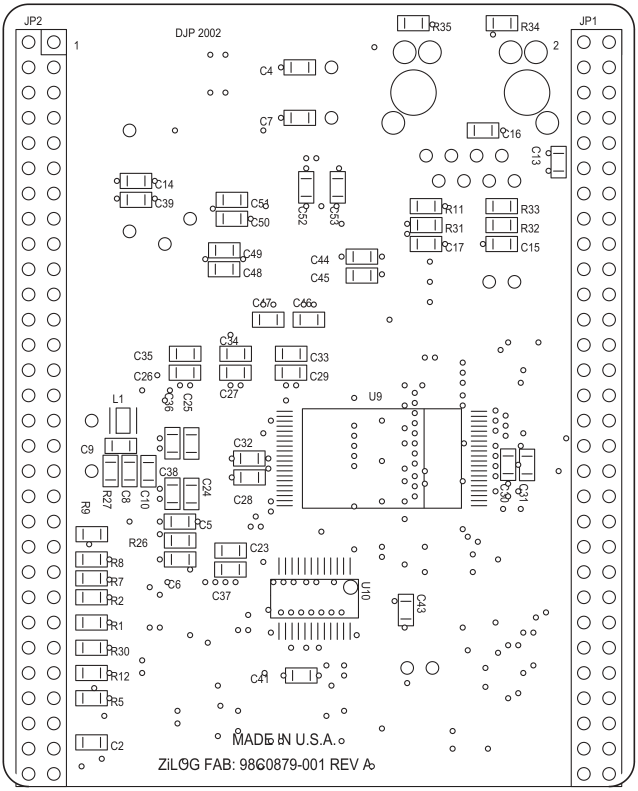


Figure 14. eZ80F91 Module—Bottom Layer

The code that follows provides an example of how this function is enabled on the eZ80F91 Module.

```
//Init_IRDA
// Make sure to first set PD2 as a port bit, an output and set it Low.

PD_ALT1 &= 0xFC;           // PD0 = uart0tx, PD1 = uart0_rx
PD_ALT2 |= 0x03;           // Enable alternate function
UART_LCTL0= 0x80;          // Select dlab to access baud rate generator
BRG_DLRL0=0x2F;            // Baud rate Masterclock/(16*baudrate)
BRG_DLRH0=0x00;            // High byte of baud rate
UART_LCTL0=0x00;           // Disable dlab
UART_FCTL0=0xC7;           // Clear tx fifo, enable fifo
UART_LCTL0=0x03;           // 8bit, N, 1 stop
IR_CTL = 0x03;             // enable IRDA Encode/decode and Receive
                           // enable bit.
//IRDA_Xmit

IR_CTL = 0x01;             //Disable receive
Putchar(0xb0);             //Output a byte to the uart0 port.
```

Flash Loader Utility

The Flash Loader utility integrated within ZDS II allows the user a convenient way to program on-chip Flash memory. Please refer to the ZiLOG Developer Studio—eZ80Acclaim! User Manual (UM0144) for more details.

Mounting the Module

The eZ80F91 Module features 2 60-pin connectors. However, the eZ80® Development Platform contains 50-pin sockets for this module. When mounting the eZ80F91 Module onto the eZ80® Development Platform, check its orientation to the platform to ensure a correct fit. Observe the underside of the module to note that pin 60 of the JP2 connector is removed and that its corresponding socket on the eZ80® Development Platform is plugged.

ZPAKII

ZPAKII is a debug tool used to develop and debug hardware and software. It is a networked device featuring an Ethernet interface and an RS232 console port. ZPAKII is shipped with a preconfigured IP address that can be changed to suit the user on a local network. For more information about using and configuring ZPAKII, please refer to the eZ80Acclaim! Development Kits Quick Start Guide (QS0020) and the ZPAKII Product User Guide (PUG0015).

ZDI Target Interface Module

The ZDI Target Interface Module provides a physical interface between ZPAKII and the eZ80® Development Platform. The TIM module supports ZDI functions. For more information on using the TIM module or ZDI please refer to the eZ80Acclaim! Development Kits Quick Start Guide (QS0020) and the eZ80F91 Module Product Specification (PS0193).

JTAG

Connector P1 is the JTAG connector on the eZ80® Development Platform. JTAG will be supported in the next offering of eZ80® products.

Application Modules

ZiLOG offers the Thermostat Application module, which can be used for evaluating and developing process control and simple I/O applications. The Thermostat Application module is equipped with an LCD display that can be used to display process control and other physical parameters. For additional reading about the Thermostat application, please see the Java Thermostat Demo Application Note (AN0104) on zilog.com.

Appendix A

General Array Logic Equations

This appendix shows the equations for disabling the Ethernet signals provided by the U10 and U15 General Array Logic (GAL) devices.

U10 Address Decoder

```
//`defineidle2'b00
//`definestate12'b01
//`definestate22'b11
//`definestate32'b10
// FOR eZ80 Development Platform Rev B
// This PAL generates 4 memory chip selects

module f92_decod(
  nCS_EX, //Enables Extension Module's Memory when Low
  nFL_DIS,//When Low, Module Flash is disabled (nDIS_FL=0),
          //When High, nDIS_FL depends upon state of
          //nmemenX
  nCS0,
  A7,     //A23
  A6,     //A22
  A5,     //A21
  A4,     //A20
  A3,     //A19
  A2,     //A18
  A1,     //A17
  A0,     //A16
```

```
        nCS2,
        nEX_FL_DIS,     //disables Flash on the expansion
                        //module, when Low
        nEM_EN,         //enables Development Platform LED
                        //and Port A emulation circuit
        nDIS_FL,        //disables Module Flash when Low
        nL_RD,          //enables local data bus to be read by CPU
        nmemen1,
        nmemen2,
        nmemen3,
        nmemen4
        );

input

        nFL_DIS         /* synthesis loc="P4"*/,
        nCS0            /* synthesis loc="P5"*/,
        nCS2            /* synthesis loc="P3"*/, //was 23
        A7              /* synthesis loc="P6"*/,
        A6              /* synthesis loc="P7"*/,
        A5              /* synthesis loc="P9"*/,
        A4              /* synthesis loc="P10"*/,
        A3              /* synthesis loc="P11"*/,
        A2              /* synthesis loc="P12"*/,
        A1              /* synthesis loc="P13"*/,
        A0              /* synthesis loc="P16"*/,
        nEX_FL_DIS      /* synthesis loc="P2"*/;
                        //input[7:0]A;upper part of Address Bus of F92
                        //A23=A7,A22=A6,A21=A5,A20=A4,A19=A3
                        //A18=A2,A17=A1,A16=A0
```

```
//wire nDIS_FL = (nFL_DIS) ? ~nEXP_EN : ~(nFL_DIS);

wire nDIS_FL = nFL_DIS & nEXP_EN; //if either of them
            //is 0 Flash is
            //disabled

assign nCS_EX = (nEX_FL_DIS) ? nEXP_EN : ~(nEX_FL_DIS);
assign nL_RD =
~((nmemen1==0)|(nmemen2==0)|(nmemen3==0)|(nmemen4==0)|(nEM_EN==0)|(
nCS_EX==0));
assign nmemen4 = ~((nCS2==0)&({A7,A6,A5,A4,A3}==5'h17));
assign nmemen3 = ~((nCS2==0)&({A7,A6,A5,A4,A3}==5'h16));
assign nmemen2 = ~((nCS2==0)&({A7,A6,A5,A4,A3}==5'h15));
assign nmemen1 = ~((nCS2==0)&({A7,A6,A5,A4,A3}==5'h14));
assign nEM_EN = ~((nCS2==0)&({A7,A6,A5,A4,A3,A2,A1,A0}==8'h80));
endmodule
```

U15 Address Decoder

```
`define     anode     8'h00
`define     cathode   8'h01
`define     latch     8'h02
// FOR eZ80 Development Platform Rev B
// This PAL generates signals that control Expansion
// Module access, LED and Port A emulation
// This device is a GAL22LV10-5JC (5ns tpd) or
// equivalent with Package = 28 pin PLCC
//
//
```

```
      A5             /* synthesis loc="P13"*/,
      A6             /* synthesis loc="P27"*/,
      A7             /* synthesis loc="P26"*/,
      nIORQ          /* synthesis loc="P2"*/,
      nRD            /* synthesis loc="P7"*/,
      nCS            /* synthesis loc="P25"*/, //CS3 for CS9800
      nWR            /* synthesis loc="P9"*/,
      nMREQ          /* synthesis loc="P16"*/;

output
      nEM_RD         /* synthesis loc="P17"*/,
      nEM_WR         /* synthesis loc="P18"*/,
      nCT_WR         /* synthesis loc="P19"*/,
      nAN_WR         /* synthesis loc="P20"*/,
      nDIS_ETH       /* synthesis loc="P21"*/;

parameter anode=8'h00;
parameter cathode=8'h01;
parameter latch=8'h02;


wire [7:0] address={A7,A6,A5,A4,A3,A2,A1,A0};

assign nEM_WR =
~((nDIS_EM==1)&(nWR==0)&(nEM_EN==0)&(address==latch));
assign nEM_RD =
~((nDIS_EM==1)&(nRD==0)&(nEM_EN==0)&(address==latch));

assign nAN_WR =
~((nDIS_EM==1)&(nWR==0)&(nEM_EN==0)&(address==anode));
```

Customer Feedback Form

If you note any inaccuracies while reading this User Manual, please copy and complete this form, then mail or fax it to ZiLOG (see *Return Information*, below). We also welcome your suggestions!

eZ80F91 Development Kit
Serial # or Board Fab #/Rev. #
Software Version
Document Number
Host Computer Description/Type

Customer Information

Name	Country
Company	Phone
Address	Fax
City/State/Zip	E-Mail

Return Information

ZiLOG
System Test/Customer Support
532 Race Street
San Jose, CA 95126
Phone: (408) 558-8500
Fax: (408) 558-8536
Email: zservice@zilog.com

Problem Description or Suggestion

Provide a complete description of the problem or your suggestion. If you are reporting a specific problem, include all steps leading up to the occurrence of the problem. Attach additional pages as necessary.