

THE WESTERN DESIGN CENTER, INC.

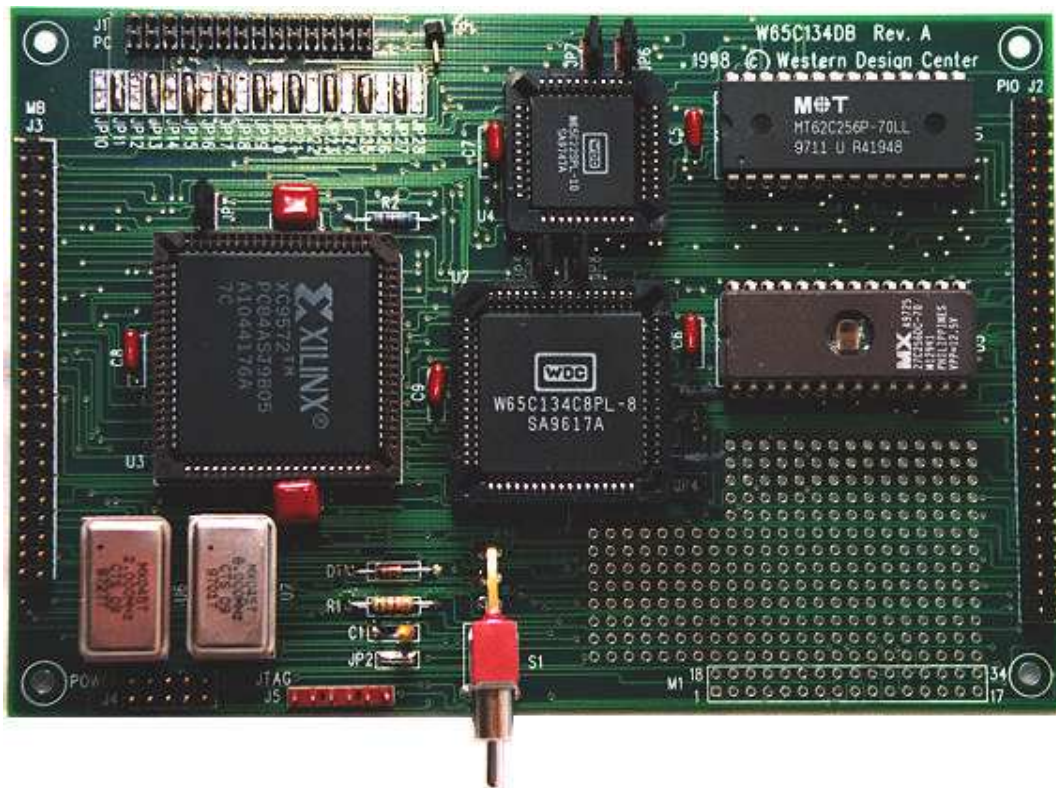
2166 E. Brown Rd. Mesa, AZ 85213

Ph 480-962-4545 Fx 480-835-6442

www.westerndesigncenter.com

Data Sheet

W65C134DB Developer Board



Copyright © 2001 by The Western Design Center, Inc.
All Rights Reserved Worldwide

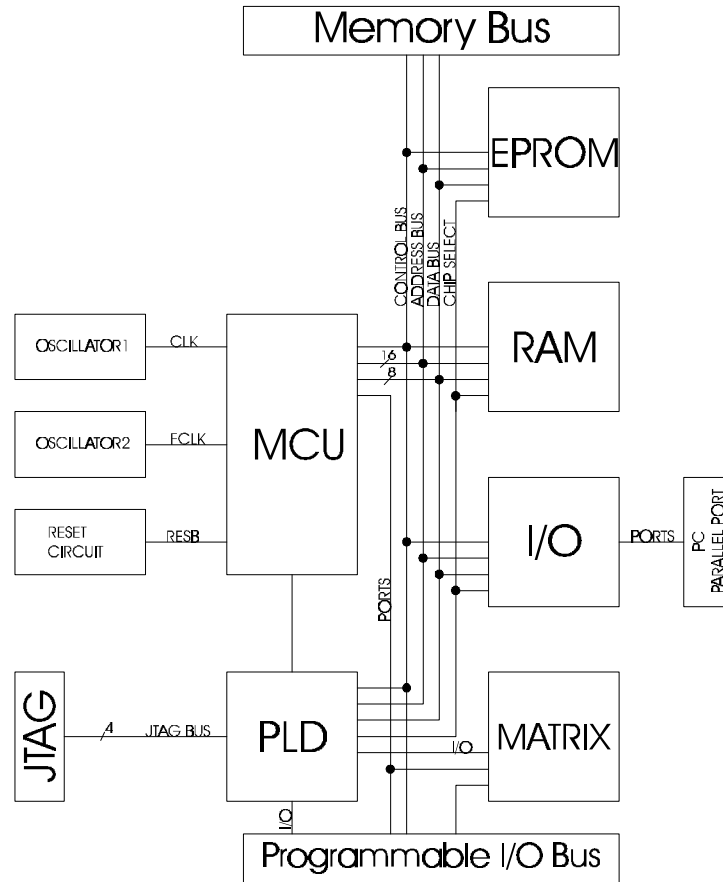
Table Of Contents

1. Overview	2
1.1 Block Diagram	2
1.2 Features	2
2. Pin Assignments	3
2.1 PC Interface J1:	4
2.2 Programmable I/O Bus J2:	5
2.3 Memory Bus J3:	6
2.4 Power Board Connector J4:	7
2.5 JTAG J5:	7
3. RTL-Code of the PLD	8
3.1 General Information	8
3.1.1 Introduction	8
3.1.2 PLD Information	8
3.2 PLD interfacing	8
3.2.1 Connections	9
3.2.2 Pin Assignments	9
3.3 Module Assignments	10
3.3.1 Continuous Assignments	11
3.3.2 RDY Assignment	11

1. Overview

The W65C134DB is used for W65C134 core microcontroller (IP) System-Chip Development, W65C134S (chip) System Development, or Embedded W65C134DB (board) Development.

1.1 Block Diagram



1.2 Features

W65C134S 8-bit MCU, total access to all control lines, Memory Bus, Programmable I/O Bus, PC Interface, 20 I/O lines, two on-board oscillators, 32K SRAM, 32K EPROM, W65C22S Versatile Interface Adapter VIA peripheral chip, on-board matrix, PLD for Memory map decoding and ASIC design.

The on-board W65C134S and the W65C22S devices have measurement points for power consumption. Power input is provided by an optional power board which plugs into the 10 pin power header.

An EPROM programmer or an EPROM emulator is required to use the board. WDC's Software Development System includes a W65C02S Assembler and Linker, W65C02S C-Compiler and Optimizer, and W65C02S Simulator/Debugger. WDC's PC IO daughter board can be used to connect the Developer Board to the parallel port of a PC.

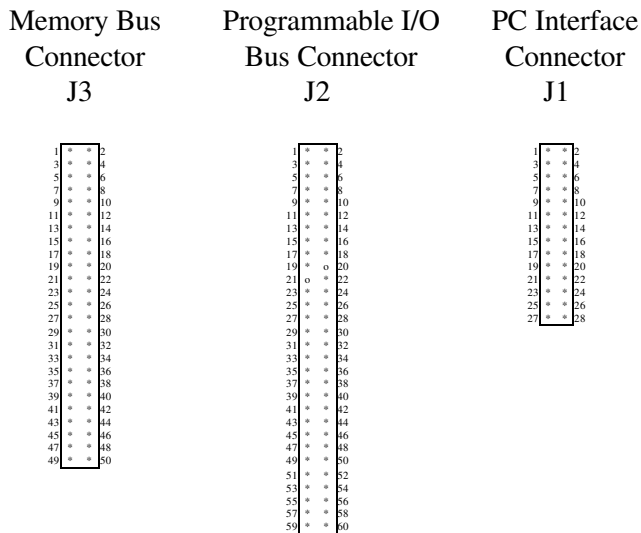
For more information about WDC's Software Development System, please refer to the Software Development System Manual available upon request.

Memory map:

CS1B: 8000-FFFF ⇒ EPROM (27C256)
 CS3B: 0100-7FFF ⇒ SRAM (62C256)
 CS2B: 0030-003F ⇒ VIA (W65C22S)

2. Pin Assignments

Total power pins: 24
 Total Matrix pins: 29



o = not connected
 * = connected

2.1 PC Interface J1:

PC Interface	Label	W65C22
8 pins	Port A:	
7	PA0	2
8	PA1	3
9	PA2	4
10	PA3	5
11	PA4	6
12	PA5	7
13	PA6	8
14	PA7	9
8 pins	Port B:	
17	PB0	10
18	PB1	12
19	PB2	13
20	PB3	14
21	PB4	15
22	PB5	16
23	PB6	17
24	PB7	18
4 pins	Handshake control:	
5	CA1	44
6	CA2	43
15	CB1	19
16	CB2	20
8 pins	Power:	
1	VCC	
2	VCC	
27	VCC	
28	VCC	
3	GND	
4	GND	
25	GND	
26	GND	

The PC Interface connector is a 28 pin connector. It is used to connect a small daughter board (WDC ZIO-1). With the WDC ZIO-1 board the developer board can be hooked up to a bi-directional printer port of a PC through a parallel printer cable. Refer to the Software Development System Manual for more information.

2.2 Programmable I/O Bus J2:

Programmable I/O Bus	Label	CPU	PLD	Matrix
15 pins	PLD:			
5	PLD36	-	36	-
6	PLD37	-	37	-
7	PLD45	-	45	-
8	PLD39	-	39	-
9	PLD46	-	46	-
10	PLD44	-	44	-
11	PLD51	-	51	-
12	PLD52	-	52	-
13	PLD47	-	47	-
14	PLD54	-	54	-
15	PLD55	-	55	-
16	PLD48	-	48	-
17	PLD50	-	50	-
18	PLD57	-	57	-
19	PLD53	-	53	-
2 pins	No Connect:			
20	N.C.	-	-	-
21	N.C.	-	-	-
30 pins	Ports:			
22	P30	37	-	-
23	P31	38	-	-
24	P32	39	-	-
25	P33	40	-	-
26	P34	41	-	-
27	P35	42	-	-
28	P36	43	-	-
29	P37	44	-	-
30	P42	56	-	-
31	P43	57	-	-
32	P44	58	-	-
33	P45	59	-	-
34	P46	60	-	-
35	P47	61	-	-
36	P50	62	-	-
37	P51	63	-	-
38	P52	64	-	-
39	P53	65	-	-
40	P54	66	-	-
41	P55	67	-	-
42	P56	68	-	-
43	P57	1	-	-
44	P60	2	-	11
45	P61	3	-	12
46	P62	4	-	13
47	P63	5	-	14
48	P64	6	-	15
49	P65	7	-	16
50	P66	8	-	17
51	P67	9	-	18
5 pins	Matrix:			
52	MATRIX1	-	-	19
53	MATRIX2	-	-	20
54	MATRIX3	-	-	21
55	MATRIX4	-	-	22
56	MATRIX5	-	-	23
8 pins	Other:			
-	CS3B	-	56	24
-	CS2B	-	65	25
-	CS1B	-	62	26
-	IRQB (VIA)	-	66	27
-	NMIPN	-	61	-
-	RDYOUT	-	58	-
-	OSCE0	-	-	28
-	OSCE1	-	-	29
8 pins	Power:			
3	VCC			
4	VCC			
57	VCC			
58	VCC			
1	GND			
2	GND			
59	GND			
60	GND			

This 60 pin connector can be used to customize the developer board and to hook up additional peripheral/daughter boards.

2.3 Memory Bus J3:

Memory Bus	Label	CPU	PLD	MATRIX
24 pins	Address Bus:			
5	A0	19	4	-
6	A1	20	1	-
7	A2	21	6	-
8	A3	22	7	-
9	A4	23	2	-
10	A5	24	3	-
11	A6	25	11	-
12	A7	26	5	-
13	A8	28	9	-
14	A9	29	13	-
15	A10	30	10	-
16	A11	31	18	-
17	A12	32	20	-
18	A13	33	12	-
19	A14	34	14	-
20	A15	35	23	-
21	A16	-	15	-
22	A17	-	24	-
23	A18	-	63	-
24	A19	-	69	-
25	A20	-	67	-
26	A21	-	68	-
27	A22	-	70	-
28	A23	-	71	-
8 pins	Data Bus			
29	D0	45	76	-
30	D1	46	72	-
31	D2	47	74	-
32	D3	48	75	-
33	D4	49	77	-
34	D5	50	79	-
35	D6	51	80	-
36	D7	52	81	-
10 pins	Control Signals:			
37	OEB	-	83	-
38	WEB	-	82	-
39	RESB	10	84	-
40	BE	-	25	-
41	RDY	-	17	-
42	PLD31	-	31	-
43	PLD32	-	32	-
44	PLD19	-	19	-
45	SYNC	-	34	-
46	PHI2	18	35	-
10 pins	CPU to PLD:			
-	BERDY	15	21	1
-	RUNSYNC	12	26	2
-	CLKOB	17	-	3
-	CLK	16	-	4
-	FCLK	14	-	5
-	FCLKOB	13	-	6
-	RWB	11	40	7
-	P42/IRQ2B	56	33	8
-	P41/IRQ1B	55	41	9
-	P40/NMIB	54	43	10
8 pins	Power:			
3	VCC			
4	VCC			
47	VCC			
48	VCC			
1	GND			
2	GND			
49	GND			
50	GND			

The Memory Bus connector is a 50 pin connector which has the most important signals to drive memory daughter boards or other peripherals.

2.4 Power Board Connector J4:

Power Board Connector	Label
8 pins	Power:
1	GND
2	GND
9	GND
10	GND
3	VCC
4	VCC
7	VCC
8	VCC
2 pins	Reset:
5	RESB
6	RESB

The Reset pins are connected direct to the W65C134 chip on the board. If solder bridge JP2 is closed, switch S1 can be used to pull RESB low (position 'up' and position 'down'). When switch S1 is not activated ('middle' position) the Reset signal RESB is pulled high.

As VCC a +5V source should be used.

2.5 JTAG J5:

JTAG	Label
4 pins	JTAG:
3	TMS
4	TDO
5	TCK
6	TDI
2 pins	Power:
1	VCC
2	GND

To reconfigure the on-board Xilinx CPLD, the Xilinx design manager and a cable is needed. However there is a introductory kit for \$99 available on the Internet. The link to this web page is as follows:

http://www.insight-electronics.com/linecard/xilinx/cpld_starter_kit

Please be aware that the Xilinx CPLD is already configured with WDCs default settings. Refer to 3.3 Module Assignments for more information.

3. RTL-Code of the PLD

3.1 General Information

The PLD chip is a XILINX XC9572 for changing the chip select and I/O functions if required. To change the PLD chip to suit your own setup, you need XILINX Data Manager for the XC9572 CPLD chip. The W65C134DB includes an on-board programming header for JTAG configuration. For more details refer to the circuit diagram.

3.1.1 Introduction

For chip select, interrupt logic and other control signal management a programmable logic device (PLD) is used. On all WDCs Developer Boards a XILINX CPLD is programmed for this purpose. The Developer Boards can be used as programming adapter board for these devices. WDCs default logic in this device occupies only about 30% of the available macrocells (see 3.1.2 PLD Information). Therefore there is plenty of room for customizing the Developer Board with the programmable logic device. For more detailed information about available I/Os and their connections, please refer to 2 Pin Assignments.

3.1.2 PLD Information

The XILINX part number is XC9572PC84-7. It has an endurance of 10,000 program/erase cycles. This CPLD has 4 Function Blocks with 18 macrocells each. Every Function Block has a programmable AND array with 36 inputs, therefore 72 true and complement signals. In the AND array these signals can be formed into Product Terms. There are 5 available Product Terms to each macrocell, giving 90 Product Terms for each Function Block. The CPLD has also 69 user I/Os and JTAG feature.

The logic for the W65C134 Developer Board requires 22 (out of 72) macrocells (30%) and 16 (out of 360) Product Terms (4%).

3.2 PLD interfacing

The pinout of the PLD device is predefined through the layout of the Developer Board. The pin assignment list is required so customized logic can be implemented easily. The pin assignment list can be seen in section 3.2.2 Pin Assignments.

3.2.1 Connections

As mentioned above, the CPLD has 69 available user I/Os. All of those user I/Os are accessible on the Developer Boards. Some are wired to the Memory Bus or Programmable I/O Bus connectors, some are wired to the on-board matrix and some are used as on-board control signals to eliminate additional logic chips.

3.2.2 Pin Assignments

PLD	INPUT/OUTPUT	USED/NOT USED	Label	Memory Bus	Programmable I/O Bus	CPU	MATRIX
Address Bus:							
4	INPUT	NOT USED	A0	5	-	19	-
1	INPUT	NOT USED	A1	6	-	20	-
6	INPUT	NOT USED	A2	7	-	21	-
7	INPUT	NOT USED	A3	8	-	22	-
2	INPUT	USED	A4	9	-	23	-
3	INPUT	USED	A5	10	-	24	-
11	INPUT	USED	A6	11	-	25	-
5	INPUT	USED	A7	12	-	26	-
9	INPUT	USED	A8	13	-	28	-
13	INPUT	USED	A9	14	-	29	-
10	INPUT	USED	A10	15	-	30	-
18	INPUT	USED	A11	16	-	31	-
20	INPUT	USED	A12	17	-	32	-
12	INPUT	USED	A13	18	-	33	-
14	INPUT	USED	A14	19	-	34	-
23	INPUT	USED	A15	20	-	35	-
15	OUTPUT	USED	A16	21	-	-	-
24	OUTPUT	USED	A17	22	-	-	-
63	OUTPUT	USED	A18	23	-	-	-
69	OUTPUT	USED	A19	24	-	-	-
67	OUTPUT	USED	A20	25	-	-	-
68	OUTPUT	USED	A21	26	-	-	-
70	OUTPUT	USED	A22	27	-	-	-
71	OUTPUT	USED	A23	28	-	-	-
Data Bus:							
76	INPUT	NOT USED	D0	29	-	45	-
72	INPUT	NOT USED	D1	30	-	46	-
74	INPUT	NOT USED	D2	31	-	47	-
75	INPUT	NOT USED	D3	32	-	48	-
77	INPUT	NOT USED	D4	33	-	49	-
79	INPUT	NOT USED	D5	34	-	50	-
80	INPUT	NOT USED	D6	35	-	51	-
81	INPUT	NOT USED	D7	36	-	52	-
Control Signals:							
83	OUTPUT	USED	OEB	37	-	-	-
82	OUTPUT	USED	WEB	38	-	-	-
84	INPUT	USED	RESB	39	-	10	-
25	OUTPUT	USED	BE	40	-	-	-
17	OUTPUT	USED	RDY	41	-	-	-
31	INPUT	NOT USED	PLD31	42	-	-	-
32	INPUT	NOT USED	PLD32	43	-	-	-
19	INPUT	NOT USED	PLD19	44	-	-	-
34	INPUT	NOT USED	SYNC	45	-	-	-
35	INPUT	USED	PHI2	46	-	18	-
PLD Bus:							
46	INPUT	NOT USED	PLD46	-	9	-	-
44	INPUT	NOT USED	PLD44	-	10	-	-
51	INPUT	NOT USED	PLD51	-	11	-	-
52	INPUT	NOT USED	PLD52	-	12	-	-
47	INPUT	NOT USED	PLD47	-	13	-	-
54	INPUT	NOT USED	PLD54	-	14	-	-
Other Signals:							
21	INPUT	USED	BE/RDY	-	-	15	1
26	INPUT	USED	RUN/SYNC	-	-	12	2
40	INPUT	USED	RWB	-	-	11	7
33	OUTPUT	USED	P42/IRQ2B	-	-	56	8
41	OUTPUT	NOT USED	P41/IRQ1B	-	-	55	9
43	OUTPUT	USED	P40/NMIB	-	-	54	10
36	INPUT	NOT USED	PLD36	-	5	-	-
37	INPUT	NOT USED	PLD37	-	6	-	-
45	INPUT	NOT USED	PLD45	-	7	-	-
39	INPUT	NOT USED	PLD39	-	8	-	-
55	INPUT	NOT USED	PLD55	-	15	-	-
48	INPUT	NOT USED	PLD48	-	16	-	-
50	INPUT	NOT USED	PLD50	-	17	-	-
57	INPUT	NOT USED	PLD57	-	18	-	-
53	INPUT	NOT USED	PLD53	-	19	-	-
56	OUTPUT	USED	CS3B	-	-	-	24
65	OUTPUT	USED	CS2B	-	-	-	25
62	OUTPUT	USED	CS1B	-	-	-	26
66	INPUT	USED	IRQB (VIA)	-	-	-	27
61	INPUT	USED	NMIPIN	-	-	-	-
58	OUTPUT	USED	RDYOUT	-	-	-	-

3.3 Module Assignments

The implemented logic of the PLD is pretty simple and straight forward. To keep it simple, the RTL-Code is not divided into submodules. The Code includes two major assignment types. These are the 'Continuous Assignments' and the 'Event-triggered Assignments'.

```
.....\
**
**          RTL-Code in Verilog of the W65C134DB PLD
**          Version 08.27.98
**
**          1998 COPYRIGHT THE WESTERN DESIGN CENTER, INC.
**          ALL RIGHTS RESERVED
**
**.....\
module PLD (A0,A1,A2,A3,A4,A5,A6,A7,A8,A9,A10,A11,A12,A13,A14,A15,A16,A17,A18,
A19,A20,A21,A22,A23,D0,D1,D2,D3,D4,D5,D6,D7,OEB,WEB,RESB,BE,RDY,
PLD31,PLD32,PLD19,SYNC,PHI2,PLD46,PLD44,PLD51,PLD52,PLD47,PLD54,
BERDY,RUNSYNC,RWB,P42IRQ2B,P41IRQ1B,P40NMIB,PLD36,PLD37,PLD45,PLD39,
PLD55,PLD48,PLD50,PLD57,PLD53,CS3B,CS2B,CS1B,IROBZIA,NMIPIN,RDYOUT);

//INPUT/OUTPUT DECLARATION
input  A0,A1,A2,A3,A4,A5,A6,A7,A8,A9,A10,A11,A12,A13,A14,A15,D0,D1,D2,D3,D4,D5,
D6,D7,RESB,RDY,BE,PHI2,RUNSYNC,RWB,IROBZIA,NMIPIN,PLD31,PLD32,PLD19,
PLD46,PLD44,PLD51,PLD52,PLD47,PLD54,PLD36,PLD37,PLD45,PLD39,PLD55,PLD48,
PLD50,PLD57,PLD53;
output A16,A17,A18,A19,A20,A21,A22,A23,OEB,WEB,BERDY,SYNC,P42IRQ2B,P41IRQ1B,
P40NMIB,CS3B,CS2B,CS1B,RDYOUT;

//TYPE OF SIGNAL DEFINITION
wire  A0,A1,A2,A3,A4,A5,A6,A7,A8,A9,A10,A11,A12,A13,A14,A15,A16,A17,A18,A19,
A20,A21,A22,A23,D0,D1,D2,D3,D4,D5,D6,D7,OEB,WEB,RESB,PHI2,BE,RDY,RUNSYNC,
RWB,IROBZIA,NMIPIN,P42IRQ2B,P41IRQ1B,P40NMIB,CS3B,CS2B,CS1B,PLD31,PLD32,
PLD19,PLD46,PLD44,PLD51,PLD52,PLD47,PLD54,PLD36,PLD37,PLD45,PLD39,PLD55,
PLD53,RDYCTRL,BERDY;
reg   RUN,SYNC,RDYOUT;

//INTERNAL SIGNALS, REGISTERS
reg   NMIB_latch; //Latch NMIB on negative edge of PHI2

//CONTINUES ASSIGNMENTS
assign {A16,A17,A18,A19,A20,A21,A22,A23} = 8'h00; //Bank Address Bus pulled low
assign OEB = !(RWB & PHI2); //Output Enable Bar
assign WEB = !(RWB & PHI2); //Write Enable Bar
assign P42IRQ2B = IROBZIA; //Interrupt Request Bar
assign P41IRQ1B = 'b1;
assign P40NMIB = NMIPIN; //Non Maskable Interrupt Bar
assign CS3B = (A15|!(A8|A9|A10|A11|A12|A13|A14)); //RAM Chip Select Bar 0100-7FFF
assign CS2B = ((A6|A7|A8|A9|A10|A11|A12|A13|A14|A15) //VIA Chip Select Bar
!(A5&A4)); //0030-003F
assign CS1B = !A15; //EPROM Chip Select Bar 8000-FFFF
assign RDYCTRL = 'b1; //RDY control, for debug assign available PLD I/O

assign BERDY = 'b1; //pulled high for debug
//use line below for system design
//always @(BE or RDY or PHI2) BERDY = (!PHI2) ? BE : RDY; //Multiplex signals BE and RDY

//change might be necessary, depending on application specific use
always @(posedge PHI2) SYNC = RUNSYNC; //SYNC is high during OpCode fetch
always @(negedge PHI2) RUN = RUNSYNC; //RUN output,if needed assign to available PLD I/O

//RDY ASSIGNMENT
always @(negedge PHI2) NMIB_latch = P40NMIB; //Indicates a 'fresh' NMIB
always @(P40NMIB or NMIB_latch or P42IRQ2B or RESB or RDY or RDYCTRL) //Force RDY after interrupt
if ( (!P40NMIB & NMIB_latch) | !P42IRQ2B | !RESB | RDY) RDYOUT = RDYCTRL;
else RDYOUT = 'b0;

endmodule
```

3.3.1 Continuous Assignments

The continuous assignment is used for most of the required signals. The Bank Address is pulled low and the chip select signals are generated through those assignments. Signals like output enable (OEB), write enable (WEB), bus enable (BE) and the interrupt signals (IRQB and NMIB) are generated by continuous assignments as well.

A signal called RDYCTRL (ready control) is continuously assigned high. This signal can be used for debugging or other features to pull the RDY signal of the microprocessor unit low.

3.3.2 RDY Assignment

The RDY assignment is an event-triggered assignment. It is based on two 'always' blocks. One 'always' block is used to latch the NMIB signal and monitors if an edge occurred on this signal.

The second 'always' block includes all interrupt signals, including reset and RDY of the microprocessor unit. The above mentioned RDYCTRL signal works like an enable signal of the RDYOUT output, but since RDYCTRL is continuously assigned high, it has no effect on this logic. The RDYCTRL becomes important when debug features are implemented.

The following schematic shows the circuit implementation of the event-triggered assignment statements:

