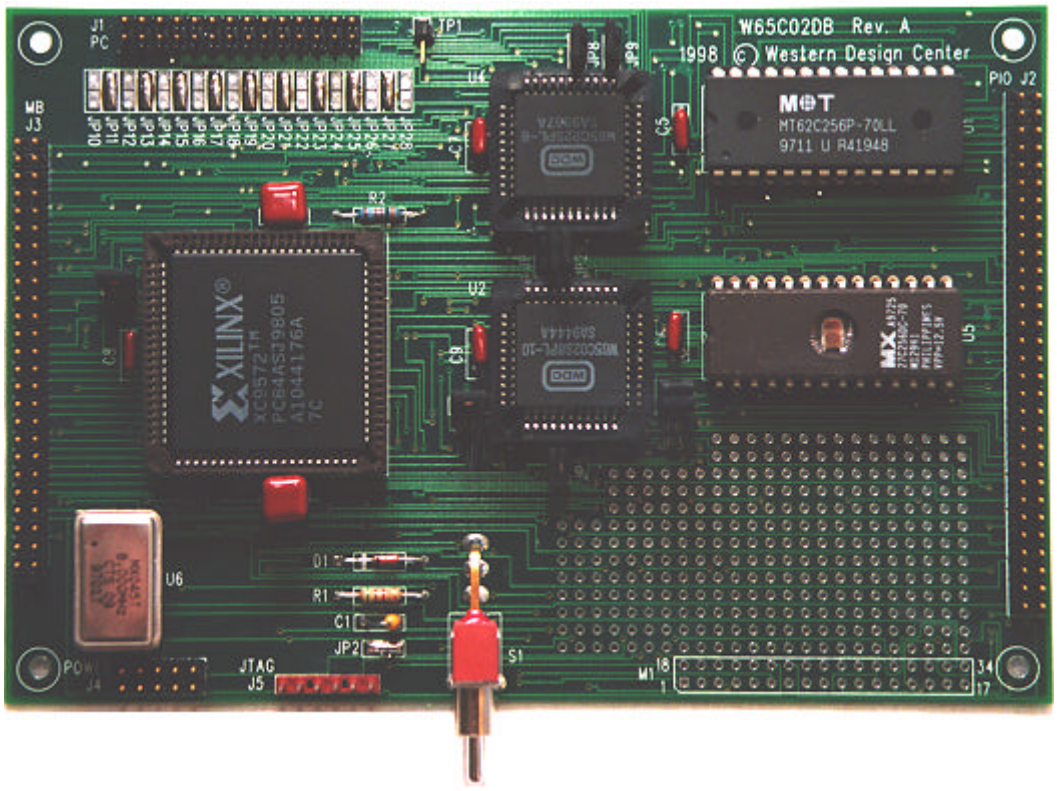
THE WESTERN DESIGN CENTER, INC.
2166 E. Brown Rd. Mesa, AZ 85213
Ph 480-962-4545 Fx 480-835-6442
www.westerndesigncenter.com

Data Sheet

W65C02DB 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**

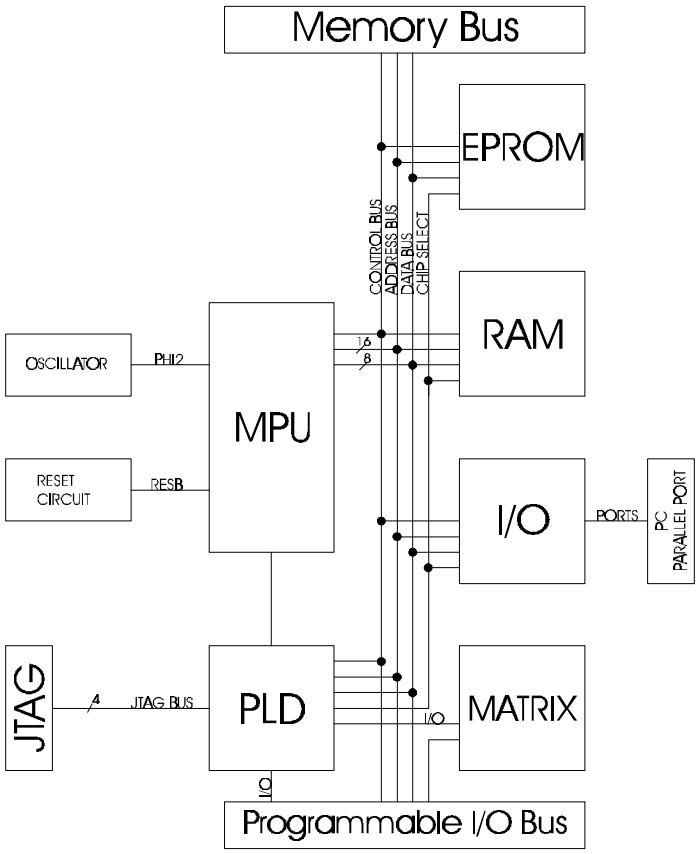
3.3.1 Continuous Assignments ____ 11

3.3.2 RDY Assignment ____ 11

1. Overview

The W65C02DB is used for W65C02 core microprocessor (IP) System-Chip Development, W65C02S (chip) System Development, or Embedded W65C02DB (board) Development.

1.1 Block Diagram



1.2 Features

W65C02S 8-bit MPU, total access to all control lines, Memory Bus, Programmable I/O Bus, PC Interface, 20 I/O lines, easy oscillator change, 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 W65C02S and the W65C22S devices have measurement points for core power consumption. Power input is provided by an optional power board which plugs into the 10 pin power header.

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.

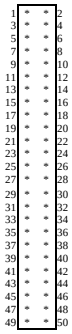
Memory map:

CS1B:	8000-FFFF	⇒	EPROM (27C256)
CS3B:	0000-00EF & 0100-7FFF	⇒	SRAM (62C256)
CS2B:	00F0-00FF	⇒	VIA (W65C22S)

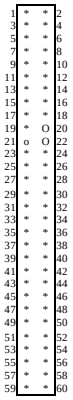
2. Pin Assignments

Total power pins: 24
Total Matrix pins: 33

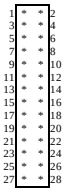
Memory Bus Connector J3



Programmable I/O Bus Connector J2



PC Interface Connector J1



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 I/O:			
5	PLD21	-	21	-
6	PLD26	-	26	-
7	PLD40	-	40	-
8	PLD33	-	33	-
9	PLD41	-	41	-
10	PLD43	-	43	-
11	PLD36	-	36	-
12	PLD37	-	37	-
13	PLD45	-	45	-
14	PLD39	-	39	-
15	PLD55	-	55	-
16	PLD48	-	48	-
17	PLD50	-	50	-
18	PLD57	-	57	-
19	PLD53	-	53	-
3 pins:	No Connect:			
20	N.C.	-	-	-
21	N.C.	-	-	-
22	N.C.	-	-	-
29 pins:				
23	OEB	-	83	9
24	WEB	-	82	10
25	D0	36	76	11
26	D1	35	72	12
27	D2	34	74	13
28	D3	33	75	14
29	D4	32	77	15
30	D5	31	79	16
31	D6	30	80	17
32	D7	29	81	18
33	A0	10	4	19
34	A1	11	1	20
35	A2	13	6	21
36	A3	14	7	22
37	A4	15	2	-
38	A5	16	3	-
39	A6	17	11	-
40	A7	18	5	-
41	A8	19	9	-
42	A9	20	13	-
43	A10	21	10	-
44	A11	22	18	-
45	A12	25	20	-
46	A13	26	12	-
47	A14	27	14	-
48	A15	28	23	-
49	RESB	44	84	-
50	RWB	38	52	6
51	PHI2	41	35	23
5 pins	Matrix:			
52	MATRIX1	-	-	24
53	MATRIX2	-	-	25
54	MATRIX3	-	-	26
55	MATRIX4	-	-	27
56	MATRIX5	-	-	28
7 pins:	Other:			
-	CS3B	-	56	29
-	CS2B	-	65	30
-	CS1B	-	62	31
-	IRQB (VIA)	-	66	32
-	NMIPIN	-	61	-
-	RDYOUT	-	58	-
-	OSCE	-	-	33
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	10	4	19
6	A1	11	1	20
7	A2	13	6	21
8	A3	14	7	22
9	A4	15	2	-
10	A5	16	3	-
11	A6	17	11	-
12	A7	18	5	-
13	A8	19	9	-
14	A9	20	13	-
15	A10	21	10	-
16	A11	22	18	-
17	A12	25	20	-
18	A13	26	12	-
19	A14	27	14	-
20	A15	28	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	36	76	11
30	D1	35	72	12
31	D2	34	74	13
32	D3	33	75	14
33	D4	32	77	15
34	D5	31	79	16
35	D6	30	80	17
36	D7	29	81	18
10 pins	Control Signals:			
37	OEB	-	83	9
38	WEB	-	82	10
39	RESB	44	84	-
40	BE	40	25	-
41	RDY	3	17	-
42	PLD31	-	31	-
43	PLD32	-	32	-
44	PLD19	-	19	-
45	SYNC	8	34	-
46	PHI2	41	35	23
8 pins	CPU to PLD:			
-	PHI2O	43	-	1
-	PHI1O	4	-	2
-	SOB	42	46	3
-	VPB	2	44	4
-	MLB	6	51	5
-	RWB	38	52	6
-	IRQB	5	47	7
-	NMIB	7	54	8
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 W65C02 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 W65C02DB 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 25% 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 W65C02 Developer Board requires 18 (out of 72) macrocells (25%) and 12 (out of 360) Product Terms (3%).

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	33	10	19
1	INPUT	NOT USED	A1	6	34	11	20
6	INPUT	NOT USED	A2	7	35	13	21
7	INPUT	NOT USED	A3	8	36	14	22
2	INPUT	USED	A4	9	37	15	-
3	INPUT	USED	A5	10	38	16	-
11	INPUT	USED	A6	11	39	17	-
5	INPUT	USED	A7	12	40	18	-
9	INPUT	USED	A8	13	41	19	-
13	INPUT	USED	A9	14	42	20	-
10	INPUT	USED	A10	15	43	21	-
18	INPUT	USED	A11	16	44	22	-
20	INPUT	USED	A12	17	45	25	-
12	INPUT	USED	A13	18	46	26	-
14	INPUT	USED	A14	19	47	27	-
23	INPUT	USED	A15	20	48	28	-
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	25	36	11
72	INPUT	NOT USED	D1	30	26	35	12
74	INPUT	NOT USED	D2	31	27	34	13
75	INPUT	NOT USED	D3	32	28	33	14
77	INPUT	NOT USED	D4	33	29	32	15
79	INPUT	NOT USED	D5	34	30	31	16
80	INPUT	NOT USED	D6	35	31	30	17
81	INPUT	NOT USED	D7	36	32	29	18
			Control Signals:				
83	OUTPUT	USED	OEB	37	23	-	9
82	OUTPUT	USED	WEB	38	24	-	10
84	INPUT	USED	RESB	39	49	44	-
25	OUTPUT	USED	BE	40	-	40	-
17	INPUT	USED	RDY	41	-	3	-
31	INPUT	NOT USED	PLD31	42	-	-	-
32	INPUT	NOT USED	PLD32	43	-	-	-
19	INPUT	NOT USED	PLD19	44	-	-	-
34	INPUT	NOT USED	SYNC	45	-	8	-
35	INPUT	USED	PHI2	46	51	41	23
46	INPUT	NOT USED	SOB	-	-	42	3
44	INPUT	NOT USED	VPB	-	-	2	4
51	INPUT	NOT USED	MLB	-	-	6	5
52	INPUT	USED	RWB	-	50	38	6
47	OUTPUT	USED	IRQB	-	-	5	7
54	OUTPUT	USED	NMIB	-	-	7	8
21	INPUT	NOT USED	PLD21	-	5	-	-
26	INPUT	NOT USED	PLD26	-	6	-	-
40	INPUT	NOT USED	PLD40	-	7	-	-
33	INPUT	NOT USED	PLD33	-	8	-	-
41	INPUT	NOT USED	PLD41	-	9	-	-
43	INPUT	NOT USED	PLD43	-	10	-	-
36	INPUT	NOT USED	PLD36	-	11	-	-
37	INPUT	NOT USED	PLD37	-	12	-	-
45	INPUT	NOT USED	PLD45	-	13	-	-
39	INPUT	NOT USED	PLD39	-	14	-	-
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	-	-	-	29
65	OUTPUT	USED	CS2B	-	-	-	30
62	OUTPUT	USED	CS1B	-	-	-	31
66	INPUT	USED	IRQB (VIA)	-	-	-	32
61	INPUT	USED	NMIPIN	-	-	-	-
58	OUTPUT	USED	RDYOUT	-	-	-	-

3.3 Module Assignments

The implemented logic of the CPLD 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 W65C02DB PLD                **
**                          Version 07.07.98                               **
**                                                                         **
**                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,SOB,VPB,MLB,RWB,IRQB,NMIB,PLD21,PLD26,
            PLD40,PLD33,PLD41,PLD43,PLD36,PLD37,PLD45,PLD39,PLD55,PLD48,PLD50,
            PLD57,PLD53,CS3B,CS2B,CS1B,IRQBVIA,NMIPIN,RDYOUT);

//INPUT/OUTPUT DECLARATION
input  A0,A1,A2,A3,A4,A5,A6,A7,A8,A9,A10,A11,A12,A13,A14,A15,RESB,SYNC,PHI2,SOB,
       VPB,MLB,RWB,IRQBVIA,NMIPIN,RDY,D0,D1,D2,D3,D4,D5,D6,D7,PLD31,PLD32,PLD19,
       PLD21,PLD26,PLD40,PLD33,PLD41,PLD43,PLD36,PLD37,PLD45,PLD39,PLD55,PLD48,
       PLD50,PLD57,PLD53;
output OEB,WEB,BE,IRQB,NMIB,CS3B,CS2B,CS1B,RDYOUT,A16,A17,A18,A19,A20,A21,A22,
       A23;

//TYPE OF SIGNAL DEFINITION
wire   A0,A1,A2,A3,A4,A5,A6,A7,A8,A9,A10,A11,A12,A13,A14,A15,RESB,SYNC,PHI2,SOB,
       VPB,MLB,RWB,IRQBVIA,NMIPIN,OEB,WEB,BE,IRQB,NMIB,CS3B,CS2B,CS1B,RDYCTRL,
       A16,A17,A18,A19,A20,A21,A22,A23,D0,D1,D2,D3,D4,D5,D6,D7,RDY,PLD31,PLD32,
       PLD19,PLD21,PLD26,PLD40,PLD33,PLD41,PLD43,PLD36,PLD37,PLD45,PLD39,PLD55,
       PLD48,PLD50,PLD57,PLD53;
reg    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 BE   = 1'b1;                                     //Bus enable pulled high
assign IRQB = IRQBVIA;                                  //Interrupt Request Bar
assign NMIB = NMIPIN;                                   //Non Maskable Interrupt Bar
assign CS3B = (A15|(!(A8|A9|A10|A11|A12|A13|A14)&       //RAM Chip Select Bar
                    A7&A6&A5&A4));                      //0000-00EF & 0100-07FFF
assign CS2B = ((A8|A9|A10|A11|A12|A13|A14|A15)|         //VIA Chip Select Bar
              !(A7&A6&A5&A4));                         //00F0-00FF
assign CS1B = !A15;                                     //EPROM Chip Select Bar 8000-FFFF
assign RDYCTRL = 1'b1;                                  //RDY control, for debug assign available PLD I/O

//RDY ASSIGNMENT
always @(negedge PHI2) NMIB_latch = NMIB;                    //Indicates a 'fresh' NMIB
always @(NMIB or NMIB_latch or IRQB or RESB or RDY or RDYCTRL) //Force RDY after interrupt
  if ((!NMIB & NMIB_latch) | !IRQB | !RESB | RDY) RDYOUT = RDYCTRL;
  else RDYOUT = 1'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:

