User manual

8-Channel Delay Timer

Rev. 2.0

F.Tonisch

09 November 2006

Specifications:

- IP Interface Logic
- Manchester Decoder for event decoding
- 8 Outputs with programmable delay and daisy chain option
- 4 Gate Outputs with programmable delay and width
- 2 Clock outputs 9MHz and 1MHZ fixed
- 1 variable Clock output (9MHZ, 4.5MHz, 2.25MHz, 1MHz)
- Interrupt logic

1. Description

The design of the 8-channel delay timer is based on the IP-UNI-XILINX board, a general purpose digital I/O module containing a XILINX XCS40XL FPGA.

The Delay Timer provides eight channels of digital pulse outputs with programmable delay, four gate outputs and various clock outputs off different frequencies.

The outputs are triggered from an incoming Manchester encoded 8-bit event on the 9MHz input clock line. Two adjacent channels are controlled by one **E**vent **C**ontrol **R**egister (ECR). After decoding the event it is compared with the contents of the upper byte of the ECR (D15-D08). In case the event matches the data the corresponding channel is triggered. The two output channels can be separately enabled/disabled (D00-D01: 1/0). The length of the output pulse is chosen to be either 1µs (D03-D02: 0) or 10µs (D03-D02: 1). The time base for the delay can be either 110ns (D5=0) or 1µs (D5=1). For very long delays two channels can be daisy chained. The pairs are Ch0-Ch2, Ch1-Ch3, Ch4-Ch6 and Ch5-Ch7. If daisy chaining is enabled (D07-D06 set) the second channel is no longer triggered by the event but by the output of the first channel in the chain.

The gate outputs are formed from two output channels. The pairing is as follows: Gate0; Ch0-Ch2, Gate1; Ch1-Ch3, Gate2; Ch4-Ch6 and Gate3; Ch5-Ch7. The gate output starts with the first output and stops with the second one. **Important: If the delay of the starting channel is bigger then that one of the stopping channel then the polarity and the timing of the gate pulse are wrong! It results in very long pulses depending on the repetition rate of the incoming events.** The delay for each channel is written to a 16-bit **D**elay **R**egister.

In addition the module serves three different clock outputs. Two of them have fixed frequencies (1MHz, 9MHz). One the third output the frequency can be selected (D07-D06 of the fourth ECR). The available frequencies are:

Var Clock:	Frequency:
0x0:	9MHz
0x1:	4.5MHz
0x2:	2.25MHz
0x3:	1MHz

This clock output is always synchronized by event 0xA7.

The module is able to generate interrupts. The source of the interrupt can be either an event itself or a triggered output with delay. The event selection for interrupts is done in four registers. Each interrupt source can be enabled separately by a mask register. Pending interrupts can be cleared also separately.

A special set of event values has been reserved for transmission of 32-bit data over the clock line. It ranges from $0x to $2x. To store data received from the incoming clock a RAM of 16-bit width and a depth of 32 words (starting location *base + 0x40*) have been implemented in the I/O space of the unit. Because two words are required to store each data sixteen incoming data words can be stored. The data source of the RAM may be chosen by means of D0 in register at location *base + 0x18*. If bit is zero the data come from input clock line otherwise it writable by the computer. The content of the RAM is readable all the time

independent from the value of this bit. The option to write data by the computer is mainly implemented for test purpose. A second bit D4 in the above mentioned register indicates if data are valid. If the value is zero it means that a data transmission is ongoing and data are still changing. This bit is automatically cleared at the beginning of a transmission and set again if data where validated.

A data transmission starts on receiving $1x, where the value x is interpreted as the address location where to store the data following within the RAM. The 32-bit data are transmitted in portions of eight 4-bit nibbles and received as $\$0y_0$ - $\$0y_7$ events. The data then are validated by receiving $2x.

Event Value:	Function:	where:
$1x	Start Transmission	x = Address in Destination RAM
$\$0y_0$ - $\$0y_7$	8 x 4bit Data	y_n = Data Nibble 0-7
$2x	End of Transmission	x = Address in Destination RAM (for validation)

The transmission of new data could be indicated by using the interrupt capabilities of the module. If desired some other mechanism to block writing of data to the RAM could be implemented in future design. The status bit could be used for instance to block writing of data. In this case it should be reset by the computer allowing new data to receive.

At the moment only one type of data event is defined: For x=0 a running 32-bit event number is send by the clock module. This event number will be used to synchronize data collected from different servers. The definition of more types may follow.

2. I/O Addresses (Register)

Address	Mode	Bit #															
		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Delay Register (DR)

0x00	R/W	Delay Channel 0
0x02	R/W	Delay Channel 1
0x04	R/W	Delay Channel 2
0x06	R/W	Delay Channel 3
0x08	R/W	Delay Channel 4
0x0A	R/W	Delay Channel 5
0x0C	R/W	Delay Channel 6
0x0E	R/W	Delay Channel 7

Event & Control Register (ECR)

0x10	R/W	Event # Ch 0-1								Daisy Chain Ch1-3	Ch0-2	Time base CH0-1	not used	Long Pulse Ch1	Ch0	Enable Ch1	Ch0
0x12	R/W	Event # Ch 2-3								not used	Input Clock Status	Time base CH2-3	not used	Long Pulse Ch3	Ch2	Enable Ch3	Ch2
0x14	R/W	Event # Ch 4-5								Daisy Chain Ch5-7	Ch4-6	Time base CH4-5	not used	Long Pulse Ch5	Ch4	Enable Ch5	Ch4
0x16	R/W	Event # Ch 6-7								Var Clock		Time base Ch6-7	not used	Long Pulse Ch7	Ch6	Enable Ch7	Ch6
0x18	R/W	not used											Rec. Stat.	not			RAM Mode
0x1A	-	not used															
0x1C	-	not used															
0x1E	-	not used															

Interrupt Control Register (IRC)

0x20	R/W	Interrupt Source Event # 1								Interrupt Source Event # 0							
0x22	R/W	Interrupt Source Event # 3								Interrupt Source Event # 2							
0x24	R/W	Interrupt Source Event # 5								Interrupt Source Event # 4							
0x26	R/W	Interrupt Source Event # 7								Interrupt Source Event # 6							
0x28	R/W	Mask Interrupt								Mask Interrupt							
		EV7	EV6	EV5	EV4	EV3	EV2	EV1	EV0	TRG7	TRG6	TRG5	TRG4	TRG3	TRG2	TRG1	TRG0
0x2A	R	Request Interrupt								Request Interrupt							
		EV7	EV6	EV5	EV4	EV3	EV2	EV1	EV0	TRG7	TRG6	TRG5	TRG4	TRG3	TRG2	TRG1	TRG0
	W	Clear Interrupt								Clear Interrupt							
		EV7	EV6	EV5	EV4	EV3	EV2	EV1	EV0	TRG7	TRG6	TRG5	TRG4	TRG3	TRG2	TRG1	TRG0
0x2C	-	not used															
0x2E	R/W	Interrupt Vector															

Table 1: Registers

3. I/O Mapping (RAM)

Address	Data Type (x)	Bit #															
		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x40	0	Event Number Low Data															
0x42		Event Number High Data															
0x44	1	reserved for later use															
0x46		reserved for later use															
0x48	2	reserved for later use															
0x4A		reserved for later use															
0x4C	3	reserved for later use															
0x4E		reserved for later use															
0x50	4	reserved for later use															
0x52		reserved for later use															
0x54	5	reserved for later use															
0x56		reserved for later use															
0x58	6	reserved for later use															
0x5A		reserved for later use															
0x5C	7	reserved for later use															
0x5E		reserved for later use															
0x60	8	reserved for later use															
0x62		reserved for later use															
0x64	9	reserved for later use															
0x66		reserved for later use															
0x68	10	reserved for later use															
0x6A		reserved for later use															
0x6C	11	reserved for later use															
0x6E		reserved for later use															
0x70	12	reserved for later use															
0x72		reserved for later use															
0x74	13	reserved for later use															
0x76		reserved for later use															
0x78	14	reserved for later use															
0x7A		reserved for later use															
0x7C	15	reserved for later use															
0x7E		reserved for later use															

A. IP Interface connections (P1)

Signal	Pin #		Signal
GND	**1**	**2**	IPCLK
RESET*	**3**	**4**	D0
D1	**5**	**6**	D2
D3	**7**	**8**	D4
D5	**9**	**10**	D6
D7	**11**	**12**	D8
D9	**13**	**14**	D10
D11	**15**	**16**	D12
D13	**17**	**18**	D14
D15	**19**	**20**	
	21	**22**	
	23	**24**	+5V
GND	**25**	**26**	GND
+5V	**27**	**28**	RW
IDSEL*	**29**	**30**	
	31	**32**	
INTSEL*	**33**	**34**	
IOSEL*	**35**	**36**	
A1	**37**	**38**	
A2	**39**	**40**	
A3	**41**	**42**	INTREQ0*
A4	**43**	**44**	
A5	**45**	**46**	
A6	**47**	**48**	ACK*
	49	**50**	GND

Table 2: IP-Interface

B. I/O connections (P2)

Signal	Pin #		Signal
Gate0	**1**	**2**	GND
Gate1	**3**	**4**	GND
Gate2	**5**	**6**	GND
Gate3	**7**	**8**	GND
1MHZOUT	**9**	**10**	GND
9MHZOUT	**11**	**12**	GND
EventCLKOUT	**13**	**14**	GND
VarCLKOUT	**15**	**16**	GND
Output0	**17**	**18**	GND
Output1	**19**	**20**	GND
Output2	**21**	**22**	GND
Output3	**23**	**24**	GND
Output4	**25**	**26**	GND
Output5	**27**	**28**	GND
Output6	**29**	**30**	GND
Output7	**31**	**32**	GND
Ext9MHZIn	**33**	**34**	GND
	35	**36**	GND
EventCLKIN	**37**	**38**	GND
GND	**39**	**40**	GND
	41	**42**	
	43	**44**	
	45	**46**	
	47	**48**	
	49	**50**	IRQ_LED

Table 3: I/OConnector

C. XILINX Pinout

	Logic Block		IP-UNI-XILINX			Delay Timer	
			Name	Type	Pin #	Name	Type
1	IP-Interface & Configuration	Confiuration PROM	CDATA	INPUT	P153	CDATA	INPUT
2			CCLK	INPUT	P155	CCLK	INPUT
3			LDS	OUTPUT	P60	PROM_OE	OUTPUT
4			DONE	OUTPUT	P104	DONE	OUTPUT
5		Control Lines	IDSEL	INPUT	P202	IDSEL-	INPUT
6			IOSEL	INPUT	P203	IOSEL-	INPUT
7			INTSEL	INPUT	P201	INTSEL-	INPUT
8			ACK	OUTPUT	P5	ACK-	OUTPUT
9			BS0	INPUT	P190	BS0-	INPUT
10			BS1	INPUT	P189	BS1-	INPUT
11			RW	INPUT	P206	RW	INPUT
12			RESET	INPUT	P106	RESET	INPIT
13		Interrupt Lines	INTREQ0	OUTPUT	P27	INTREQ0-	OUTPUT
14			INTREQ1	OUTPUT	P20		
15		Address Lines	A1	INPUT	P199	IPA1	INPUT
16			A2	INPUT	P178	IPA2	INPUT
17			A3	INPUT	P172	IPA3	INPUT
18			A4	INPUT	P194	IPA4	INPUT
19			A5	INPUT	P193	IPA5	INPUT
20			A6	INPUT	P196	IPA6	INPUT
21		Data Lines	D0	BIDIR	P36	IPDATA0	BIDIR
22			D1	BIDIR	P37	IPDATA1	BIDIR
23			D2	BIDIR	P34	IPDATA2	BIDIR
24			D3	BIDIR	P35	IPDATA3	BIDIR
25			D4	BIDIR	P31	IPDATA4	BIDIR
26			D5	BIDIR	P32	IPDATA5	BIDIR
27			D6	BIDIR	P30	IPDATA6	BIDIR
28			D7	BIDIR	P29	IPDATA7	BIDIR
29			D8	BIDIR	P23	IPDATA8	BIDIR
30			D9	BIDIR	P24	IPDATA9	BIDIR
31			D10	BIDIR	P17	IPDATA10	BIDIR
32			D11	BIDIR	P16	IPDATA11	BIDIR
33			D12	BIDIR	P8	IPDATA12	BIDIR
34			D13	BIDIR	P9	IPDATA13	BIDIR
35			D14	BIDIR	P15	IPDATA14	BIDIR
36			D15	BIDIR	P14	IPDATA15	BIDIR
37		Test Point	IPSEL	OUTPUT	P200	IPSEL	OUTPUT
38			READTP	OUTPUT	P3	READTP	OUTPUT
39			WRITETP	OUTPUT	P204	WRITETP	OUTPUT
40		Clock	IPCLK	INPUT	P49	8MHZIN	INPUT

Table 4: FPGA Pinout

	Logic Block		IP-UNI-XILINX			Delay Timer	
			Name	Type	Pin #	Name	Type
41	User Space	IO-Buffer0	BUFFDIR0	OUTPUT	P107	Logic Low	OUTPUT
42			BUFF_OE0	OUTPUT	P109	Logic Low	OUTPUT
43			IO0	BIDIR	P68	GATE0	OUTPUT
44			IO1	BIDIR	P75	GATE1	OUTPUT
45			IO2	BIDIR	P88	GATE2	OUTPUT
46			IO3	BIDIR	P96	GATE3	OUTPUT
47			IO4	BIDIR	P82	1MHZ_OUT	OUTPUT
48			IO5	BIDIR	P133	9MHZTP	OUTPUT
49			IO6	BIDIR	P144	TCLKOUT	OUTPUT
50			IO7	BIDIR	P80	VAR_CLK	OUTPUT
51		IO-Buffer1	BUFFDIR1	OUTPUT	P129	Logic Low	OUTPUT
52			BUFF_OE1	OUTPUT	P128	Logic Low	OUTPUT
53			IO8	BIDIR	P67	OUTPUT0	OUTPUT
54			IO9	BIDIR	P74	OUTPUT1	OUTPUT
55			IO10	BIDIR	P69	OUTPUT2	OUTPUT
56			IO11	BIDIR	P76	OUTPUT3	OUTPUT
57			IO12	BIDIR	P87	OUTPUT4	OUTPUT
58			IO13	BIDIR	P95	OUTPUT5	OUTPUT
59			IO14	BIDIR	P89	OUTPUT6	OUTPUT
60			IO15	BIDIR	P97	OUTPUT7	OUTPUT
61		IO-Buffer2	BUFFDIR2	OUTPUT	P152		
62			BUFF_OE2	OUTPUT	P151		
63			IO16	BIDIR	P159		
64			IO17	BIDIR	P161		
65			IO18	BIDIR	P162		
66			IO19	BIDIR	P163		
67			IO20	BIDIR	P164		
68			IO21	BIDIR	P165		
69			IO22	BIDIR	P166		
70			IO23	BIDIR	P167		

Table 5: FPGA Pinout (cont.)

	Logic Block		IP-UNI-XILINX			Delay Timer	
			Name	Type	Pin #	Name	Type
71	User Space	Inputs	SPEC_IN	INPUT	P147	TCLKIN	INPUT
72			IN1	INPUT	P123		
73			IN2	INPUT	P125		
74		Output	AUX_OUT	OUTPUT	P126	LED_INTERR	OUTPUT
75		Delay1	1DLYDRV	OUTPUT	P142	1DLYDRV	OUTPUT
76			1DEL20	INPUT	P141	1DEL20	INPUT
77			1DEL40	INPUT	P138	1DEL40	INPUT
78			1DEL60	INPUT	P139	1DEL60	INPUT
79			1DEL80	INPUT	P136	1DEL80	INPUT
80			1DEL100	INPUT	P137	1DEL100	INPUT
81		Delay2	2DLYDRV	OUTPUT	P119	2DLYDRV	OUTPUT
82			2DEL20	INPUT	P117	2DEL20	INPUT
83			2DEL40	INPUT	P115	2DEL40	INPUT
84			2DEL60	INPUT	P116	2DEL60	INPUT
85			2DEL80	INPUT	P113	2DEL80	INPUT
86			2DEL100	INPUT	P114	2DEL100	INPUT
87		Test Point	TP1	OUTPUT	P185	Q1	OUTPUT
88			TP2	OUTPUT	P146	Q2	OUTPUT
89			TP3	OUTPUT	P145	SDATAOUT	OUTPUT
90			TP4	OUTPUT	P134	DVLDOUT	OUTPUT
91		Header	HIO0	BIDIR	P58		
92			HIO1	BIDIR	P59		
93			HIO2	BIDIR	P61		
94			HIO3	BIDIR	P62		
95			HIO4	BIDIR	P70		
96			HIO5	BIDIR	P77		
97			HIO6	BIDIR	P90		
98			HIO7	BIDIR	P98		
99		Clock	Ext_CLK	INPUT	P2	9MHZ_CLK	INPUT

Table 6: FPGA Pinout (cont.)