miniLA
Communication Protocol & Specification

Stateanalysis

Firmware version 2.2

Communication protocol

miniLA Interface

The miniLA communicates via parallel port in standard EPP mode. EPP mode is not described here. More information can be found for example on page http://www.beyondlogic.org/epp/epp.htm.

EPP Memory map

There are several registers for control of the miniLA mapped in the EPP address range. Mapping is shown in the table below:

Address	*Write*	*Read*
0	Control Register	Data Register
1	Trigger Ev. Counter Register	Status & Version Reg.
2	Trigger Length Counter Reg.	unused
3	unused	Status Register 2
4	Pre/Postrigger Register	unused
5	Trigger Value Reg. (X7:X0)	unused
6	Trigger Value Reg. (X15:X8)	unused
7	Trigger Edge Reg. (E7:E0)	unused
8	Trigger Edge Reg. (E15:E8)	unused
9	Trigger Mask Reg. (M7:M0)	unused
10	Trigger Mask Reg. (M15:M8)	unused
11	unused	unused
12	unused	unused
13	Trigger Control Register	unused

*) when reading from unused entry, the outputs are in high impedance

Description of EPP registers

- **Control Register**

7	6	5	4	3	2	1	0
RUN	CLR	PCLK	AINC	STOP	-	BS1	BS0

BS1:BS0 Byte selector, defines byte order when reading data from data register. After every read from data register the byte selector is automatically incremented.

Bytesel BS1:BS0	*Data at data register*
00	data (7:0)
01	data (15:8)
10	time (7:0)
11	time (15:8)

data(15:0) is the channel sample value, time(15:0) is number of clocks of sample duration decreased by 1.

STOP Writing of logic 1 interrupts the sampling.

AINC Writing of logic 1 enables the auto increment of memory address. The memory address is incremented when bit is 1, data from data register are read and current bytesel has value “11”.

PCLK Writing of logic 1 increments the memory address. Bit is cleared automatically. Active only when bit DONE active or timebase set to "11111".

CLR Writing of logic 1 resets the miniLA. It doesn't affect the EPP registers.

RUN Writing of logic 1 starts the sampling.

- **Trigger Events Counter Register**

7	6	5	4	3	2	1	0
-	-	-	-	J3	J2	J1	J0

J3:J0 Bits J3:J0 define number of trigger hits before the postrigger data are stored decreased by 1.

- **Trigger Length Counter Register**

7	6	5	4	3	2	1	0
-	-	-	-	K3	K2	K1	K0

K3:K0 Bits define the minimal number of clocks for which the input signal must be stable to be considered as trigger hit. Value K3:K0 = "0000" is not valid.

- **Pre/Post trigger Register**

7	6	5	4	3	2	1	0
-	-	-	PRD	P3	P1	P1	P0

P3:P0 Bits P3:P0 define length of the pretrigger/postrigger in 8K steps.

PRD Pretrigger Disable - Writing of logic 1 disables the pretrigger.

PRD	***P3:P0***	***Pretrigger length***	***Postrigger length***	***Trigger position***
0	0000	8K	120K	8K
0	0001	16K	112K	16K
0	0010	24K	116K	24K
.	.	.	.	.
0	1110	120K	8K	120K
0	1111*	--	--	--
1	0000	0	8K	1
1	0001	0	16K	1
1	0010	0	24K	1
.	.	.	.	.
1	1111	0	128K	1

*) not supported setting

- **Trigger Value Registers**

adr 5:

7	6	5	4	3	2	1	0
X7	X6	X5	X4	X3	X2	X1	X0

adr 6:

7	6	5	4	3	2	1	0
X15	X14	X13	X12	X11	X10	X9	X8

X15:X0 Bits X15:X0 are compared with input data D15:D0. The trigger hit occurs if the result of comparison is true.

- **Trigger Edge Registers**

adr 7:

7	6	5	4	3	2	1	0
E7	E6	E5	E4	E3	E2	E1	E0

adr 8:

7	6	5	4	3	2	1	0
E15	E14	E13	E12	E11	E10	E9	E8

E15:E0 Bits E15:E0 defines whether input bits D15:D0 are checked for edge (1) or for value (0). In cooperation with trigger value register it defines rising edge (bit in value reg 1) or falling edge (bit in value reg 0) sensitivity.

Note1: Only unmasked bits can be set to 1. In control software this can be accommodated by command like edge_reg=edge_reg&mask_reg.

Note2: When edge detection is used, trigger length register should be set to 1.

- **Trigger Mask Registers**

adr 9:

7	6	5	4	3	2	1	0
M7	M6	M5	M4	M3	M2	M1	M0

adr 10:

7	6	5	4	3	2	1	0
M15	M14	M13	M12	M11	M10	M9	M8

M15:M0 Bits M15:M0 defines which bits in Trigger Value Registers (X15:X0) are tested or ignored. When at logic 1, the bit is tested, otherwise ignored.

- **Trigger Control Register**

7	6	5	4	3	2	1	0
-	-	-	-	-	-	ETS	ETV

ETV Value for comparison with external trigger input.

ETS Writing of logic 1 enables external trigger input and disables internal trigger. Conditions defined by Trigger Events Counter and Trigger Length Counter applies also for external trigger.

IIT Writing of logic 1 inverts the result of internal trigger (trigger event occurs when result of comparation between data bus and Trigger Value Reg (masked) is FALSE).

- **Data Register**

Depending on the current state of the miniLA, this register provides access either to input data or to data stored in memory during sampling. Selection of byte order is defined by bits BS1 and BS0 in control register.

RUN	*Data available at Data Register*
0	Input data (delayed by 3 clocks)
1	Data from memory

- **Status and Version Register**

7	6	5	4	3	2	1	0
DONE	HW2	HW1	HW0	FW3	FW2	FW1	FW0

FW3:FW0 Bits FW3:FW0 defines the CPLD firmware version.

HW2:HW0 Bits HW2:HW0 defines the hardware version.

DONE Status bit, logic 1 when sampling completed or interrupted.

- **Status Register 2**

7	6	5	4	3	2	1	0
DONE	RUN	CLR	TRIG	SCT	-	-	-

SCT Sample Counter Watermark – Used for data retrieval, see section Recommended Command Sequence.

TRIG Logic 1 indicates successful detection of trigger condition during sampling.

CLR Copy of bit CLR in control register. Logic 1 indicates the miniLA is in reset state.

RUN Copy of bit RUN in control register. Logic 1 indicates the start of sampling was enabled. Bit is not cleared when sampling is completed or interrupted.

DONE Logic 1 indicates that the sampling was completed or interrupted and the data can be read from the memory.

Recommended command sequence

- **Initialization**
  1. Write a value 0x40 into Control Register. This puts the miniLA in reset state.
  2. Write appropriate values in all remaining registers (registers are not affected by miniLA reset, all registers are set to 0x00 at power-up)
  3. At this stage the input data of the miniLA can be accessed by reading the data register.
  4. Sampling starts by writing a value 0x80 into Control Register.

- **Data sampling**
  1. During the sampling period, the input data can be accessed in the same way as in the initialization phase.
  2. Periodically read Status Register (or Status Register 2) and wait until the bit DONE is at logic 1.
  3. If it is necessary to interrupt the sampling, write value 0x08 into Control Register.

- **Data retrieving**
  1. Data can be retrieved from the memory only when the status bit DONE is at logic 1.
  2. Write value 0x1F to Pre/Post trigger register.
  3. Read Status Register 2 and check the value of bit SCT.

4. Data are ready for read if the value of bit SCT is at logic 1. If it is not, increment the memory address (by writing value 0x20 into control register) so many times (*n* times) until bit SCT is at logic 1.
5. Generate 1 additional clock pulse (write value 0x20 into control register), enable auto increment (write 0x10 into control register) and then read 128K-(*n*+1) longwords of data by successive reading of the Data Register.

Specification

Number of channels	16
Sample rate	100MHz (fixed)
Sample memory	128K * 16 for channel samples and 128K * 16 for hardware compression data
Input current	±10μA
Input capacitance	10pF
Min/max input voltage	0 to 5V
Input threshold log0 / log1	<0.8V / >2.4V
Setup/Hold time	4ns / 0ns for internal clock (clk_o)
Trigger channels	16 channels (0,1,X, rising/falling edge)
Number of trigger events	selectable 1-16
Min. length of trigger event	selectable 1-16
Pretrigger/posttrigger size	selectable in 8K steps
Communication interface	LPT (in EPP mode)