128 CH TDC-SCALER For 10x10 GEM

SEA Technical Note

Issue: Draft
Revision: 0.9

Reference: INFN no reference
Created: 25th Nov 2010
Last modified: 1st Feb 2013

Prepared By: A. Balla, P. Ciambrone, F.Murtas

INFN - LNF

Document Status Sheet

1. Document Title: 128 CH TDC-SCALER For 10x10 GEM **2. Document Reference Number:**			
3. Issue	**4. Revision**	**5. Date**	**6. Reason for change**
Draft	0.1	25th Nov 2010	First version
Draft	0.2	17th Jan 2011	Added connector pin out
Draft	0.3	14th Jun 2011	Added Windows system setup
Draft	0.4	4th Aug 2011	Added some TDC features
Draft	0.5	14th May 2012	Added gate delay register
Draft	0.6	29th May 2012	Added gate enable bit and CRLF to the Block read command
Draft	0.7	10th Jan 2013	Added immediate data sending to scaler with channel value Added character for scaler configuration Added character for tdc configuration
Draft	0.8	31st Jan 2013	Added some figures Modified character order
Draft	0.9	1st Feb 2013	Added some minor correction

Table 1 Document status sheet

Info

The major features of the TDC GEM board are explained here, for new version of manual, test software, programming file, check at www.infn.lnf.it/~balla in the “Applicazioni” section, TDC GEM.

Table of Contents

SEA Technical Note i
Document Status Sheet ii
Info ii
Table of Contents 3
List of figures 5
1. GEM Board 6
1.1. Board view 6
1.2. Overview 6
1.3. Board connectors 7
1.4. Communication interface 8
1.4.1. Windows system setup 8
1.4.2. Optical interface (not yet implemented) 9
1.4.3. USB interface 10
1.4.4. Serial Interface 10
1.4.5. Ethernet interface 10
1.5. Board programming 11
2. TDC GEM BOARD accessible base address 13
2.1. TDC GEM register [0xCEA00000] 13
2.1.1. STROBE register [0xCEA00000 + 0x14] 14
2.1.2. STATUS register [0xCEA00000 + 0x18] 15
2.1.3. TDC CTRL register [0xCEA00000 + 0x00] 16
2.1.4. TDC GATE WIDTH register [0xCEA00000 + 0x0C] 17
2.1.1. SCALER CTRL register [0xCEA00000 + 0x08] 18
2.1.2. SCALER GATE WIDTH register [0xCEA00000 + 0x04] 19
2.1.3. SCALER GATE NUMBER register [0xCEA00000 + 0x10] 20
2.1.4. GATE ARRIVAL register [0xCEA00000 + 0x1C] 21
2.2. SCALER Memory Region [0xC5E00000] 22
2.2.1. Example of a block read to the scaler memory region: 22
2.3. TDC Memory Region [0xC5E20000] 23
2.3.1. Example of a block read to the TDC memory region: 23
2.4. SPI SLOWCONTROL base address [0x83400000] 24
2.4.1. SPI 8 channel DAC 0 [0x83400000] 25
2.4.2. SPI 8 channel DAC 1 [0x83400100] 25
3. Help on board configuration 26
3.1. Scaler configuration 26

3.1.1. Mode 0, single shot, external gate, external width 26
3.1.2. Mode 1, single shot, external gate, software width and delay 27
3.1.3. Mode 2, single shot, software gate, software width and delay 27
3.1.4. Mode 3, multi shot, external gate, external width 28
3.1.5. Mode 4, multi shot, external gate, software width and delay 28
3.1.6. Mode 5, multi shot, software gate, software width and delay 29
3.1.7. Mode 6, multi shot, external gate, external width, auto send 29
3.1.8. Mode 7, multi shot, external gate, software width, auto send 30
3.1.9. Mode 8, multi shot, software gate, software width, auto send 30
3.2. Tdc configuration 31
3.2.1. Mode common start, positive or negative edge 31
3.2.2. Mode common stop, positive or negative edge 32
3.2.3. Mode common start, positive and negative edge 32
3.2.4. Mode common stop, positive and negative edge 33

List of figures

Table 1 Document status sheet ii
Figure 1 TDC GEM BOARD Version 1 6
Figure 2 TDC GEM BOARD CONNECTION 7
Table 2 TDC GEM Board Connector 7
Table 4 Usb report descriptor 10
Figure 3 Impact software 11
Figure 4 Programming options 11
Figure 5 Program the prom 12
Figure 6 RS232 console 12
Table 5 TDC GEM Base Address 13
Table 6 TDC GEM Register base address 13
Table 7 Bit assignment of STROBE Register 14
Table 8 Bit assignment of STATUS register 15
Table 9 Bit assignment of TDC CTRL register 16
Table 10 Bit assignment of TDC GATE WIDTH register 17
Table 12 Bit assignment of SCALER GATE WIDTH register 19
Table 13 Bit assignment of SCALER GATE NUMBER register 20
Table 14Bit assignment of SCALER GATE ARRIVAL register 21
Table 15 Scaler Memory Region 22
Table 16 TDC Memory Region 23
Table 17 SPI SLOWCONTROL chip base address 24
Table 18 SPI 8 channel DAC 0 chip base address 25
Table 19 SPI 8 channel DAC 1 chip base address 25
Figure 7 Mode 0 configuration 26
Figure 8 Mode 1 configuration 27
Figure 9 Mode 2 configuration 27
Figure 10 Mode 3 configuration 28
Figure 11 Mode 4 configuration 28
Figure 12 Mode 5 configuration 29
Figure 13 Mode 6 configuration 29
Figure 14 Mode 7 configuration 30
Figure 15 Mode 8 configuration 30
Figure 16 Mode common start positive edge 31
Figure 17 Mode common stop negative edge 32
Figure 18 Mode common start positive and negative edge 32
Figure 19 Mode common stop positive and negative edge 33

1. GEM Board

1.1. Board view

Figure 1 TDC GEM BOARD Version 1

1.2. Overview

In this manual, an FPGA based acquisition board of 128 LVDS channels is described.

The firmware loaded on the board is able perform counts (**SCALER MODE**) or time measurements (**TDC MODE**) of discriminated signal coming on the 128 input channels.

For the time measurements it is possible to select the positive or negative or both edges; in common start or common stop mode. The minimum input channel width is **5 ns**. The maximum input channel frequency is **100 MHz**, if a negative or positive edge TDC mode is used, **50 MHz** if both edge TDC mode has been chosen (minimum distance between measured edge must be **10 ns**).

The time measurements are stored in **16 bits** register with **1.25** ns per bit. It is also possible to define the maximum number of hits acquired for each channel and the acquisition time window.

In the SCALRE MODE the 128 counts are stored in a **24 bits** registers.

1.3. **Board connectors**

The Board connector and switch are show in Figure 2, see Table 2 for connector description.

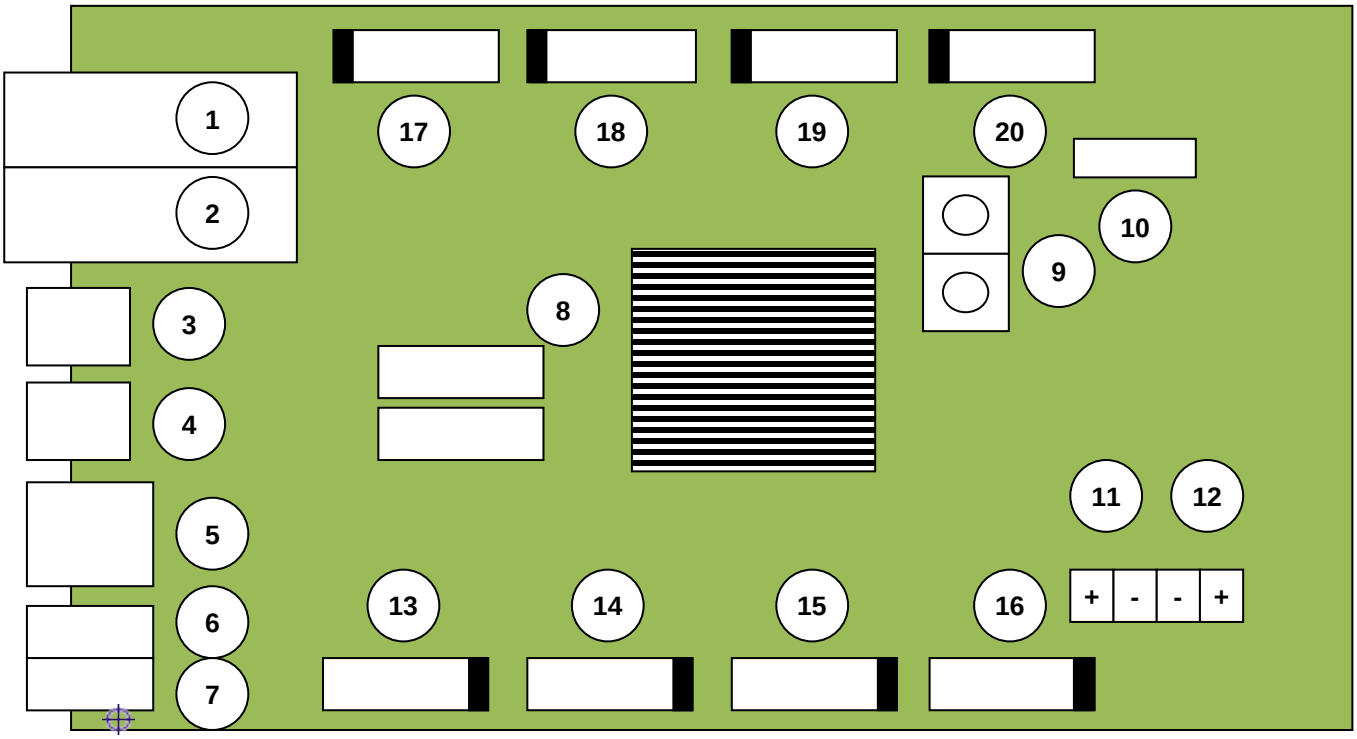

Figure 2 TDC GEM BOARD CONNECTION

Connector#	Connector description
1, 2	Optical link connectors (SEA LINK)
3, 4	USB connectors
5	10, 100, 1000 Ethernet connector
6	TDC common start or TDC common stop (input NIM Level)
7	GATE (input NIM Level)
8	RS232 connector (console) baud 9600,8,n,1
9	Address switch for Board **MAC ADDRESS** is : **000a35000100** + (address switch)
10	Programming header, use Xilinx platform cable to reprogram FPGA
11, 12	Power Connector (11 Analog, 12 Digital)
13	From Carioca Gem Board input channel **80-95** (black strip is the less significant channel)
14	From Carioca Gem Board input channel **64-79** (black strip is the less significant channel)
15	From Carioca Gem Board input channel **112-127** (black strip is the less significant channel)
16	From Carioca Gem Board input channel **96-111** (black strip is the less significant channel)
17	From Carioca Gem Board input channel **16-31** (black strip is the less significant channel)
18	From Carioca Gem Board input channel **0-15** (black strip is the less significant channel)
19	From Carioca Gem Board input channel **48-63** (black strip is the less significant channel)
20	From Carioca Gem Board input channel **32-47** (black strip is the less significant channel)

Table 2 TDC GEM Board Connector

1.4. Communication interface

The communication interfaces (Optical, USB, Ethernet and serial) use a custom protocol to read or write register on the TDC GEM.

It is based on 8 bit character and it has only 3 command, one for write register, one for read single and one for read block.

To perform a write command at address 0x81400000 with a value 0x8,the string to be sent is :

w8140000008 <enter>

or

w814000000008 <enter>

or

w8140000000000008 <enter>

be careful at the value record length must always be 2, 4 or 8 characters long.

The **<enter>** character is a carriage return line feed (/n/r).

The interface return the string **w8140000000000008** to acknowledge the operation.

To perform a read command at address 0x81400000 the string to be sent is:

r81400000 <enter>

The interface return:

r8140000000000008

to acknowledge the operation. The data is always the last 8 characters.

Ethernet also implement a block transfer command to download large data buffer of data, to perform a block transfer command at address 0xC5E20000 the string to be sent is :

BC5E20000<enter>

The interface return:

BC5E2000000000000<enter>

00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 <enter>

00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 <enter>

00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 <enter>

00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 <enter>

ecc.

;<enter>

The characters ";<enter>" ends the data block trasmitted.

1.4.1. Windows system setup

Windows by default sends the TCP ACK packets received with 200 ms delay. To eliminate this delay is necessary to add some keys in the registry using regedit.

The following changes apply to Windows Vista (Google search on those relating to other operating systems).

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters\Interfaces\{Adapter-id}]

{Adapter-id} = Ethernet board (check the assigned ip number)

"TcpAckFrequency"= dword:00000001

from here you can also paste the following text into a . reg file and run it to make the changes automatically.
Reboot the PC after making the changes.

Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\MSMQ]

[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\MSMQ\Parameters]

"TCPNoDelay"=dword:00000001

[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\MSMQ\Parameters\OCMsetup]

[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\MSMQ\Parameters\Security]

"SecureDSCommunication"=dword:00000000

[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\MSMQ\Parameters\setup]

[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\MSMQ\Setup]

1.4.2. Optical interface (not yet implemented)

For this type of communication a dedicated optical interface is necessary to perform slow control operation and download data.

1.4.3. USB interface

USB interface support High speed USB. The communication through USB interface use a HID driver. The commands are sent through numbers and not character strings

This is the report descriptor used.

Report descriptor	Hex	Hex	Hex
Usage Page (Vendor Defined)	0x06	0x00	0xFF
Usage (IO Device)	0x09	0x01	
Collection (Application)	0xA1	0x01	
Logical Minimum (0)	0x15	0x00	
Logical Maximum (255)	0x25	0xFF	
Usage Minimum (0)	0x19	0x01	
Usage Maximum (255)	0x29	0xFF	
Report Size (64)	0x75	0x08	
Report Count (128)	0x95	0x40	
Input (Data,Var,Abs) = data to host	0x81	0x02	
Usage Minimum (0)	0x19	0x01	
Usage Maximum (255)	0x29	0xFF	
Report Size (8)	0x75	0x08	
Report Count (16)	0x95	0x10	
Output (Data,Var,Abs) = data from host	0x91	0x02	
End Collection	0xC0		

Table 3 Usb report descriptor

1.4.4. Serial Interface

Serial interface is a RS232 interface set at 9600 baud 8 bit 1 bit stop. Commands are sent in characters

1.4.5. Ethernet interface

The Ethernet interface responds on port 23 using telnet protocol. The last byte of the MAC ADDRESS of the board is manually configurable by the address switch.

Board MAC ADDRESS is : 000a35000100 + (address switch)

Commands are sent in characters. Board respond to ping command and telnet operation can be performed according to custom protocol command.

1.5. Board programming

Board can be reprogrammed by the user if a Xilinx platform usb cable are available. On the board there are two Xilinx prom (xcv32p).In one prom it is stored the hardware configuration of the FPGA and in the second prom there are the software code of the inside FPGA PowerPC.

To reprogram the board a software can be downloaded from Xilinx site at:

http://www.xilinx.com/support/download/index.htm.

select version 12.1 and install the Lab Tools version that contain the software to reprogram the board.

This are the instruction to reprogram the board:

- Launch Impact software
- select the boundary scan window
- select and run the scan chain command

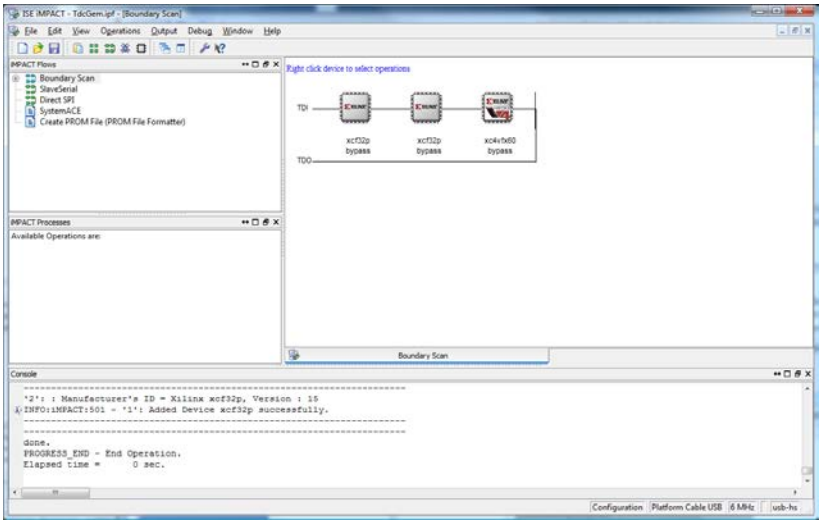

Figure 3 Impact software

- 3 device will appear, the first is the software prom where to put the file .mcs with the PPC software, the second is the prom where to put the file .mcs with the FPGA hardware, the third device is the FPGA and must be put in bypass mode.

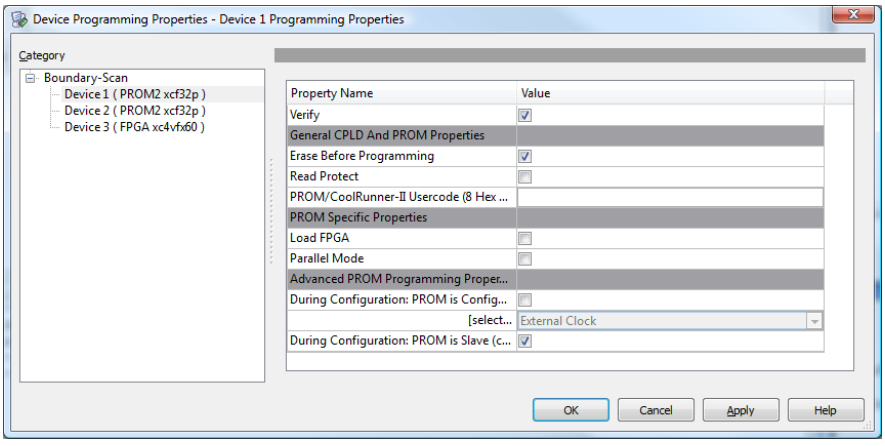

Figure 4 Programming options

- Select the right file for all the device, set the programming option with “erase before programming” and “verify” for each prom.
- Right click on each prom, a menu will appear, then select program, the prom will be erased, programmed and verified.
- Repeat the operation on the second prom.
- Close the program.

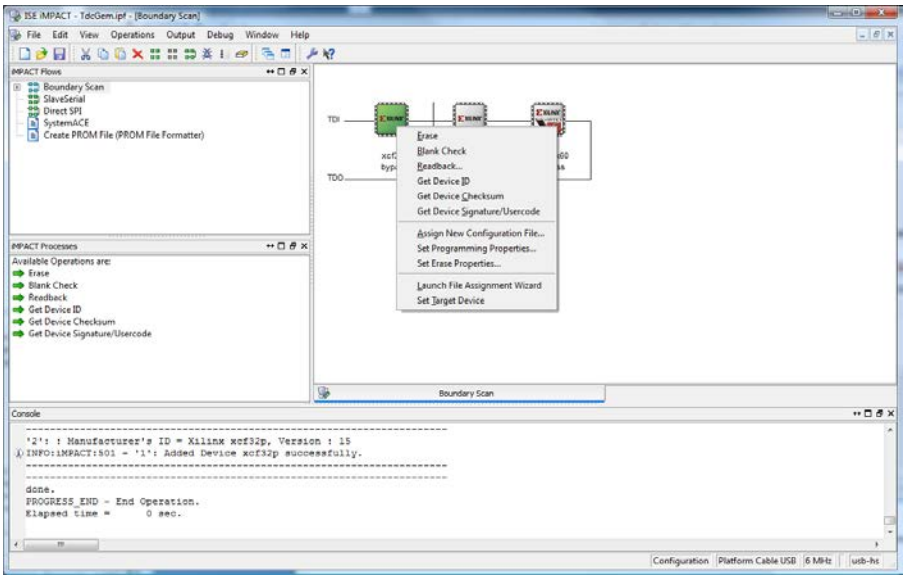

Figure 5 Program the prom

Switch off and switch on the board, if everything is ok the led near the program header will be on.

A console could be connected to rs232 connector (number 8 in Figure 2), if everything is ok the new software and hardware version will appear.

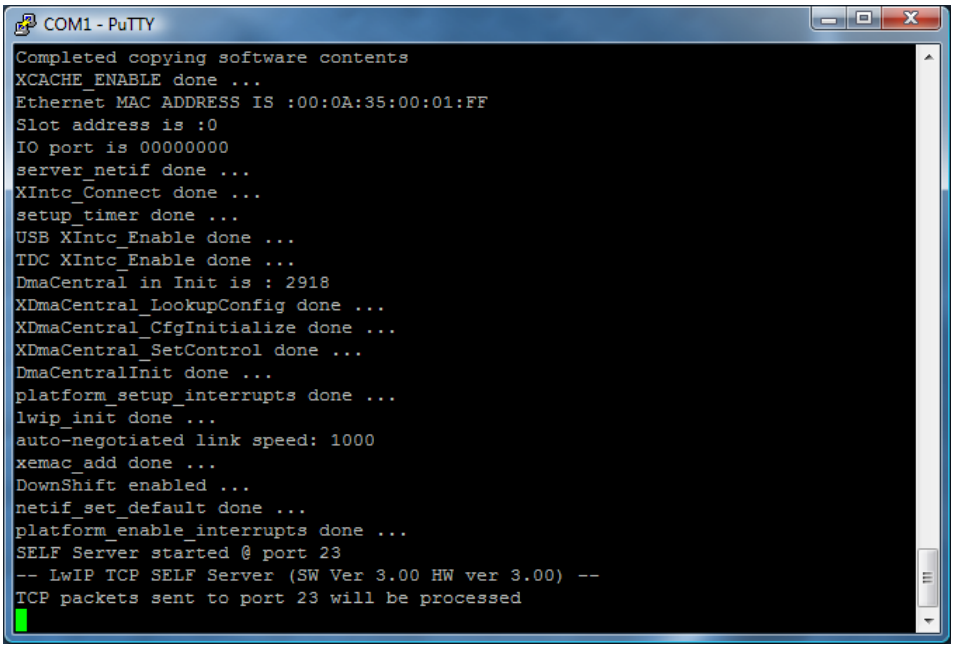

Figure 6 RS232 console

2. TDC GEM BOARD accessible base address

In this chapter, the TDC GEM internal base address are shown.

Board base address	Base Address	High Address	Size	mode access
TDC GEM Register	0xCEA00000	0xCEA0FFFF	64K	**SINGLE**
TDC GEM Scaler memory region	0xC5E00000	0xC5E0FFFF	64K	**SINGLE/BLOCK**
TDC GEM tdc memory region	0xC5E20000	0xC5E2FFFF	64K	**SINGLE/BLOCK**
SPI_SLOWCONTROL	0x83400000	0x8340FFFF	4K	**SINGLE**

Table 4 TDC GEM Base Address

2.1. TDC GEM register [0xCEA00000]

In this chapter, the TDC GEM register are described in detail.

RW = read and write register

RO = read only register

WO = write only register

Addr. (hex)	TDC GEM Register	Use for setup	Reference name	Default value	Mode
14	STROBE register	COMMON	SLV_REG5	0x00000000	WO
18	STATUS register	COMMON	SLV_REG6	0x00000000	RO
00	TDC CTRL register	TDC	SLV_REG0	0x000000F0	RW
0C	TDC GATE WIDTH register	TDC	SLV_REG3	0x00000064	RW
08	SCALER CTRL register	SCALER	SLV_REG2	0x00000001	RW
04	SCALER GATE WIDTH register	SCALER	SLV_REG1	0x05F5E100	RW
10	SCALER GATE NUMBER register	SCALER	SLV_REG4	0x00000000	RW
1C	GATE ARRIVAL register	SCALER	SLV_REG7	0x00000000	RW
20	Not used		SLV_REG8	0x00000000	RW
24	Not used		SLV_REG9	0x00000000	RW
28	Not used		SLV_REG10	0x00000000	RW
2C	Not used		SLV_REG11	0x00000000	RW
30	Not used		SLV_REG12	0x00000000	RW
34	Not used		SLV_REG13	0x00000000	RW
38	Not used		SLV_REG14	0x00000000	RW
3C	Not used		SLV_REG15	0x00000000	RW

Table 5 TDC GEM Register base address

2.1.1. STROBE register [0xCEA00000 + 0x14]

This register is write only.

Write bit(0) to logic one send a software strobe to the scaler.

Write bit(1) to logic one send a software start/stop to the tdc.

The default value is 0 (0x00000000).

STROBE register			
Bit		**Bit**	
0	SCALER STROBE	16	
1	TDC START/STOP	17	
2		18	
3		19	
4		20	
5		21	
6		22	
7		23	
8		24	
9		25	
10		25	
11		26	
12		28	
13		29	
14		30	
15		31	

Table 6 Bit assignment of STROBE Register

2.1.2. STATUS register [0xCEA00000 + 0x18]

This register is read only.

The status register report the number of TDC channel, the maximum number of hit per channel, the number of bit per TDC channel and if a scaler is present.

Bit(0 to7) are the number of board channels.

Bit(8 to15) are the max number of hit per tdc channels.

Bit(16 to23) are the max number of bit per tdc channels.

Bit(24 to 31) are the hardware version of the FPGA, read this number and divived by ten to have the version number (example : read decimal value of 25, the hardware version is 2.50).

STATUS register			
Bit		**Bit**	
0	TDC_NUM_OF_CH	16	TDC_NUM_OF_BIT
1	TDC_NUM_OF_CH	17	TDC_NUM_OF_BIT
2	TDC_NUM_OF_CH	18	TDC_NUM_OF_BIT
3	TDC_NUM_OF_CH	19	TDC_NUM_OF_BIT
4	TDC_NUM_OF_CH	20	TDC_NUM_OF_BIT
5	TDC_NUM_OF_CH	21	TDC_NUM_OF_BIT
6	TDC_NUM_OF_CH	22	TDC_NUM_OF_BIT
7	TDC_NUM_OF_CH	23	TDC_NUM_OF_BIT
8	TDC_NUM_OF_HIT	24	TDC_NUM_OF_BIT
9	TDC_NUM_OF_HIT	25	TDC_VER_NUM
10	TDC_NUM_OF_HIT	25	TDC_VER_NUM
11	TDC_NUM_OF_HIT	26	TDC_VER_NUM
12	TDC_NUM_OF_HIT	28	TDC_VER_NUM
13	TDC_NUM_OF_HIT	29	TDC_VER_NUM
14	TDC_NUM_OF_HIT	30	TDC_VER_NUM
15	TDC_NUM_OF_HIT	31	TDC_VER_NUM

Table 7 Bit assignment of STATUS register

2.1.3. TDC CTRL register [0xCEA00000 + 0x00]

This register is read and write.

Bit (0) of this register set the TDC in COMMON STOP if set to logic zero or in COMMON START if set to logic one.

Bit (4 to 7) set the number of hit per channel the TDC have to collect, default value is 15 according with the maximum number of hit written in the status register.

Bit(8) set if the TDC have to write the arrived of the positive edge.

Bit(9) set if the TDC have to write the arrived of the negative edge.

At list one of this two bit must be set at logic one.

The minimum distance between two acquiring edge can't be less than 10 ns.

Bit(16) set to logic one indicate data ready to read.

A block read command reset the bit(16) to zero.

If a start/stop arrived, the TDC DATAREADY are set to one.

When TDC DATAREADY are different from zero the data are ready to be read. If a Block read are performed TDC DATAREADY return to zero.

Read character "TDC configuration" to understand the use of this register.

TDC CTRL register			
Bit		**Bit**	
0	Mode (0)	16	TDC DATAREADY
1		17	
2		18	
3		19	
4	Hit Count(0)	20	
5	Hit Count(1)	21	
6	Hit Count(2)	22	
7	Hit Count(3)	23	
8	Select POSITIVE edge	24	
9	Select NEGATIVE edge	25	
10		25	
11		26	
12		28	
13		29	
14		30	
15		31	

Table 8 Bit assignment of TDC CTRL register

2.1.4. TDC GATE WIDTH register [0xCEA00000 + 0x0C]

This register is read and write.

The gate width value is calculated to be 1 bit = 10 ns.

In COMMON START MODE or in COMMON STOP MODE the TDC acquire all input pulse for the time defined in this register.

The default value is 100 (0x00000064) is equal at 1 micro second of gate width.

TDC GATE WIDTH register			
Bit		**Bit**	
0	TDC GATE WIDTH	16	TDC GATE WIDTH
1	TDC GATE WIDTH	17	TDC GATE WIDTH
2	TDC GATE WIDTH	18	TDC GATE WIDTH
3	TDC GATE WIDTH	19	TDC GATE WIDTH
4	TDC GATE WIDTH	20	TDC GATE WIDTH
5	TDC GATE WIDTH	21	
6	TDC GATE WIDTH	22	
7	TDC GATE WIDTH	23	
8	TDC GATE WIDTH	24	
9	TDC GATE WIDTH	25	
10	TDC GATE WIDTH	25	
11	TDC GATE WIDTH	26	
12	TDC GATE WIDTH	28	
13	TDC GATE WIDTH	29	
14	TDC GATE WIDTH	30	
15	TDC GATE WIDTH	31	

Table 9 Bit assignment of TDC GATE WIDTH register

2.1.1. SCALER CTRL register [0xCEA00000 + 0x08]

This register is read and write.

Bit(0) of this register set the use of software or external width. (0=software, 1=external)

Bit(1) enable the gate hardware input :
when set to 1 it enables the external lemo connector input
when set to 0 it disable the external lemo connector input
after a gate arrival this bit is automatically reset to zero.

Bit(2) set to one, enable the immediate data send on the Ethernet interface only.

Bit (4 to 31) set the delay between the external gate arrival and the internal gate generation, the time step is **10 ns per bit**.

The default value is : 0x00000001 (gate width from external connector and delay = 0).

Refer to the **Scaler Configuration** chapter for more details on the use of this register.

SCALER CTRL register			
Bit		**Bit**	
0	Gate use external width (1=external)	16	Gate Delay Time (12)
1	Gate enable (1=enabled)	17	Gate Delay Time (13)
2	Immediate data send (1=enabled)	18	Gate Delay Time (14)
3		19	Gate Delay Time (15)
4	Gate Delay Time (0)	20	Gate Delay Time (16)
5	Gate Delay Time (1)	21	Gate Delay Time (17)
6	Gate Delay Time (2)	22	Gate Delay Time (18)
7	Gate Delay Time (3)	23	Gate Delay Time (19)
8	Gate Delay Time (4)	24	Gate Delay Time (20)
9	Gate Delay Time (5)	25	Gate Delay Time (21)
10	Gate Delay Time (6)	25	Gate Delay Time (22)
11	Gate Delay Time (7)	26	Gate Delay Time (23)
12	Gate Delay Time (8)	28	Gate Delay Time (24)
13	Gate Delay Time (9)	29	Gate Delay Time (25)
14	Gate Delay Time (10)	30	Gate Delay Time (26)
15	Gate Delay Time (11)	31	Gate Delay Time (27)

Table 10 Bit assignment of SCALER CTRL register

2.1.2. SCALER GATE WIDTH register [0xCEA00000 + 0x04]

This register is read and write.

Through this register it is possible to define by software the time width of the acquisition. If the software width is enabled (see register 0xCEA00008) the acquisition time is independent from the external signal width, and the starting time coincide with the arrival of the external signal (the first falling edge of LEMO "GATE" input)

The time step is **10ns per bit**.

The default value is 100 000 00 (0x05F5E100) is equal at 1 second of gate width.

Width smaller than 1 millisecond are not allowed.

SCALER GATE WIDTH register			
Bit		**Bit**	
0	SCALER GATE WIDTH	16	SCALER GATE WIDTH
1	SCALER GATE WIDTH	17	SCALER GATE WIDTH
2	SCALER GATE WIDTH	18	SCALER GATE WIDTH
3	SCALER GATE WIDTH	19	SCALER GATE WIDTH
4	SCALER GATE WIDTH	20	SCALER GATE WIDTH
5	SCALER GATE WIDTH	21	SCALER GATE WIDTH
6	SCALER GATE WIDTH	22	SCALER GATE WIDTH
7	SCALER GATE WIDTH	23	SCALER GATE WIDTH
8	SCALER GATE WIDTH	24	SCALER GATE WIDTH
9	SCALER GATE WIDTH	25	SCALER GATE WIDTH
10	SCALER GATE WIDTH	25	SCALER GATE WIDTH
11	SCALER GATE WIDTH	26	SCALER GATE WIDTH
12	SCALER GATE WIDTH	28	SCALER GATE WIDTH
13	SCALER GATE WIDTH	29	SCALER GATE WIDTH
14	SCALER GATE WIDTH	30	SCALER GATE WIDTH
15	SCALER GATE WIDTH	31	SCALER GATE WIDTH

Table 11 Bit assignment of SCALER GATE WIDTH register

2.1.3. SCALER GATE NUMBER register [0xCEA00000 + 0x10]

This register is read and write, but only the first 16 bit can be written.

This register set the number of consecutive gate acquisition.

After all gate as been arrived the number of generated gate are written in the bit(16 to 23) of the register.

The default value is 0 (0x00000000) at reset condition.

If the value of this register is set to zero and a gate arrived, the SCALER GATE NUMBER(0) and the SCALER DATAREADY(0) are set to one.

When SCALER DATAREADY are different from zero and the first 16 bit are equal to the second 16 bit, the data are ready to be read.

Read character “Scaler configuration” to understand the use of this register.

.

SCALER GATE NUMBER register			
Bit		**Bit**	
0	SCALER GATE NUMBER(0)	16	SCALER DATAREADY(0)
1	SCALER GATE NUMBER(1)	17	SCALER DATAREADY(1)
2	SCALER GATE NUMBER(2)	18	SCALER DATAREADY(2)
3	SCALER GATE NUMBER(3)	19	SCALER DATAREADY(3)
4	SCALER GATE NUMBER(4)	20	SCALER DATAREADY(4)
5	SCALER GATE NUMBER(5)	21	SCALER DATAREADY(5)
6	SCALER GATE NUMBER(6)	22	SCALER DATAREADY(6)
7	SCALER GATE NUMBER(7)	23	SCALER DATAREADY(7)
8	SCALER GATE NUMBER(8)	24	SCALER DATAREADY(8)
9	SCALER GATE NUMBER(9)	25	SCALER DATAREADY(9)
10	SCALER GATE NUMBER(10)	25	SCALER DATAREADY(10)
11	SCALER GATE NUMBER(11)	26	SCALER DATAREADY(11)
12	SCALER GATE NUMBER(12)	28	SCALER DATAREADY(12)
13	SCALER GATE NUMBER(13)	29	SCALER DATAREADY(13)
14	SCALER GATE NUMBER(14)	30	SCALER DATAREADY(14)
15	SCALER GATE NUMBER(15)	31	SCALER DATAREADY(15)

Table 12 Bit assignment of SCALER GATE NUMBER register

2.1.4. GATE ARRIVAL register [0xCEA00000 + 0x1C]

This register is store at the time of the last scaler gate arrived .

The gate time value is calculated to be 1 bit = 1 micro second.

SCALER GATE ARRIVAL register			
Bit		**Bit**	
0	GATE ARRIVAL(0)	16	GATE ARRIVAL(16)
1	GATE ARRIVAL(1)	17	GATE ARRIVAL(17)
2	GATE ARRIVAL(2)	18	GATE ARRIVAL(18)
3	GATE ARRIVAL(3)	19	GATE ARRIVAL(19)
4	GATE ARRIVAL(4)	20	GATE ARRIVAL(20)
5	GATE ARRIVAL(5)	21	GATE ARRIVAL(21)
6	GATE ARRIVAL(6)	22	GATE ARRIVAL(22)
7	GATE ARRIVAL(7)	23	GATE ARRIVAL(23)
8	GATE ARRIVAL(8)	24	GATE ARRIVAL(24)
9	GATE ARRIVAL(9)	25	GATE ARRIVAL(25)
10	GATE ARRIVAL(10)	25	GATE ARRIVAL(26)
11	GATE ARRIVAL(11)	26	GATE ARRIVAL(27)
12	GATE ARRIVAL(12)	28	GATE ARRIVAL(28)
13	GATE ARRIVAL(13)	29	GATE ARRIVAL(29)
14	GATE ARRIVAL(14)	30	GATE ARRIVAL(30)
15	GATE ARRIVAL(15)	31	GATE ARRIVAL(31)

Table 13Bit assignment of SCALER GATE ARRIVAL register

2.2. SCALER Memory Region [0xC5E00000]

In this chapter, the Scaler Memory Region register are described in detail.

At the end of every gate arrived the value of each channel is written in this memory region.

Channel are arranged in 128 x 4 memory location.

A single read can be performed to read the value of specific channel instead to perform a block read.

Addr. (hex)	SCALER Memory Region	Default value	Mode
000	CH 0	0x00000000	RO
004	CH 1	0x00000000	RO
008	CH 2	0x00000000	RO
01C	CH 3	0x00000000	RO
TO	…		
1FC	CH 127	0x00000000	RO

Table 14 Scaler Memory Region

2.2.1. Example of a block read to the scaler memory region:

BC5E00000<enter>

The interface return:

BC5E000000F0acd19<enter>

00000001 00000001 00000007 00000003 00000000 00000000 00000000 00000000 <enter>

00000000 00000000 00000000 00000000 00000000 000000A0 000000F0 00000000 <enter>

00000000 00000000 00000000 00000000 00000000 00000000 00000001 00000000 <enter>

00000000 00000005 00000000 00000000 00000001 00000000 00000000 00000000 <enter>

00000001 00000001 00000007 00000003 00000000 00000000 00000000 00000000 <enter>

00000000 00000000 00000000 00000000 00000000 000000A0 000000F0 00000000 <enter>

00000000 00000000 00000000 00000000 00000000 00000000 00000001 00000000 <enter>

ecc.

The first 8 character after the echo of the Block command is the gate time arrival of the gate in microseconds, the next sixteen line are the channel value.

Frame size is fixed.

In this example channel 0,1,2,3,13,14,22,25,28,32,33 ecc. contain data.

2.3. TDC Memory Region [0xC5E20000]

In this chapter, the TDC Memory Region register are described in detail.

At the end of every start/stop arrived the value of each channel is written in this memory region, at the end of the channel scanning the number of valid data are written in :

location 0x000 for the first 32 channel, until 0x3FC for the channel value,

location 0x400 for the second 32 channel, until 0x7FC for the channel value,

location 0x800 for the third 32 channel, until 0xBFC for the channel value,

location 0xC00 for the last 32 channel, until 0xFFC for the channel value,

of that memory region.

Because the TDC are multi-hit, in every location are written the channel number of the arrival data, the bit (23 to 31) of every location is the channel number.

A single read can be performed to read the value of specific channel instead to perform a block read.

Addr. (hex)	TDC Memory Region		Default value	Mode
000	NUMBER OF VALID DATA TO READ		0x00000000	RO
004	CH NUMBER	CH VALUE	0x00000000	RO
008	CH NUMBER	CH VALUE	0x00000000	RO
01C	CH NUMBER	CH VALUE	0x00000000	RO
TO	…			
XXX	CH LAST	CH VALUE	0x00000000	RO

Table 15 TDC Memory Region

2.3.1. Example of a block read to the TDC memory region:

BC5E20000<enter>

The interface return:

BC5E000000000000A<enter>

00000001 00000010 01000007 01000013 01000C01 65000A20 65000B00 65000CD0 <enter>

6F000003 7C000100 <enter>

The first 8 character after the echo of the Block command is the number of active channel to read, the next line are the channel value, the most significant byte of the four byte word is the channel number, the last tree byte are the channel value.

Frame size is variable.

In this example there are 2 hit on channel “00”, 3 hit on channel “01”, 3 hit on channel “65” (101), 1 hit on channel “6F” (111) and 1 hit on channel “7C” (124).

2.4. SPI SLOWCONTROL base address [0x83400000]

In this chapter, the SPI SLOWCONTROL chip base address are described in detail.

The TDC GEM Board contain two 8 channel 12 bit DAC to set the threshold A end threshold B of the front end boards discriminators.

The DAC are write only.

The onboard DAC should be 12, 14 or 16 bit device, if using 12 bit device the written value must be multiplied by 16, if 14 bit DAC is used written value must be multiplied by 4, if 16 bit DAC is used written value is the right value.

On TDCGEM the DAC is a 12 bit device.

Write example of 0xFFF value to channel 3 (THRB CH 80-95) on DAC 1 (12 bit DAC):

W8340010C0000FFF0<enter>

From Addr. (hex)	SPI SLOWCONTROL chip & board	Reference name	To Addr. (hex)	Mode
000	8 channel DAC 0	R_DAC0	02B	WO
100	8 channel DAC 1	R_DAC1	12B	WO

Table 16 SPI SLOWCONTROL chip base address

2.4.1. SPI 8 channel DAC 0 [0x83400000]

The SPI 8 channel DAC 0 register is a write only register.

The write value is a 16 bit data.

Addr. (hex)	Register Description	Reference name	Mode
0x00	Write value on channel 0 (THRA CH 0-15)	SPI_D0C0	WO
0x04	Write value on channel 1 (THRB CH 0-15)	SPI_D0C1	WO
0x08	Write value on channel 2 (THRA CH 16-31)	SPI_D0C2	WO
0x0C	Write value on channel 3 (THRB CH 16-31)	SPI_D0C3	WO
0x10	Write value on channel 4 (THRA CH 32-47)	SPI_D0C4	WO
0x14	Write value on channel 5 (THRB CH 32-47)	SPI_D0C5	WO
0x18	Write value on channel 6 (THRA CH 48-63)	SPI_D0C6	WO
0x1C	Write value on channel 7 (THRB CH 48-63)	SPI_D0C7	WO

Table 17 SPI 8 channel DAC 0 chip base address

2.4.2. SPI 8 channel DAC 1 [0x83400100]

The SPI 8 channel DAC 1 register is a write only register.

The write value is a 16 bit data.

Addr. (hex)	Register Description	Reference name	Mode
0x00	Write value on channel 0 (THRA CH 64-79)	SPI_D1C0	WO
0x04	Write value on channel 1 (THRB CH 64-79)	SPI_D1C1	WO
0x08	Write value on channel 2 (THRA CH 80-95)	SPI_D1C2	WO
0x0C	Write value on channel 3 (THRB CH 80-95)	SPI_D1C3	WO
0x10	Write value on channel 4 (THRA CH 96-111)	SPI_D1C4	WO
0x14	Write value on channel 5 (THRB CH 96-111)	SPI_D1C5	WO
0x18	Write value on channel 6 (THRA CH 112-127)	SPI_D1C6	WO
0x1C	Write value on channel 7 (THRB CH 112-127)	SPI_D1C7	WO

Table 18 SPI 8 channel DAC 1 chip base address

3. Help on board configuration

Some help to configure the board.

First step is to set the threshold A and threshold B of the front end boards discriminators.

Write the SPI 8 channel DAC 0 and SPI 8 channel DAC 1 at a suitable value.

3.1. Scaler configuration

The Board is able to counts 128 rates in several time window configurations.
The Board has an input lemo connector, called GATE, through which an **external** NIM signal can be received to define the acquisition time window.

The start time of **gate** can be started with a **software** command or with the **external** signal.

The time **width** can be defined by **software** or can be used the width of the **external** signal.

It is possible to define also a **delay** that will be applied between the external signal arrival and the real acquisition start.

The gate of the scaler channels could be made using multiple options. When data are ready to download (bit<0 to 15> of **SCALER GATE NUMBER Configuration register [0xCEA00010]** are equal to bit<16 to 31>), use the procedure in character "TDC GEM SCALER Memory Region", for data download and data frame detail.

3.1.1. Mode 0, single shot, external gate, external width

This is the common use of a scaler; the acquisition time gate is defined totally by the external signal and the 128 LVDS signals are counted and stored in memory at the end of the gate.

Figure 7 Mode 0 configuration

To set this option use :

SCALER Configuration register [0xCEA00008]

- the time width is defined by the external signal, set the bit[0] to **1**
- enable external gate, set the bit[1] to **1**
- this bit is automatically reset to **0** when a gate occur
- set the bit[1] to **1** every time you want enable for an another external gate
- set the **Immediate data send** bit[2] to **0**

Minimum gate width is 10 ns. Maximum gate width undefined.

3.1.2. Mode 1, single shot, external gate, software width and delay

In this mode the external gate input act as a trigger. When the external signal arrive, an internal gate is generated with a programmable time width, after a programmable delay

Figure 8 Mode 1 configuration

SCALER GATE WIDTH register [0xCEA00004].

- define the time width, bit [0..31] with time step of 10 ns per bit.

SCALER Configuration register [0xCEA00008])

- define the **delay**, bit [4..31] with time step of 10 ns per bit.
- the time width is defined by software, set the bit[0] to **0**
- enable external gate, set the bit[1] to **1**
- this bit is automatically reset to **0** when a gate occur
- set the bit[1] to **1** every time you want enable for an another external gate
- set the **Immediate data send** bit[2] to **0**

Range of gate width : . Minimum 10 ns - Maximum 42.0 s.

Range of delay : . . . . . Minimum 0 ns - Maximum 2.7 s.

3.1.3. Mode 2, single shot, software gate, software width and delay

In this mode a software strobe is used to generate an internal gate.

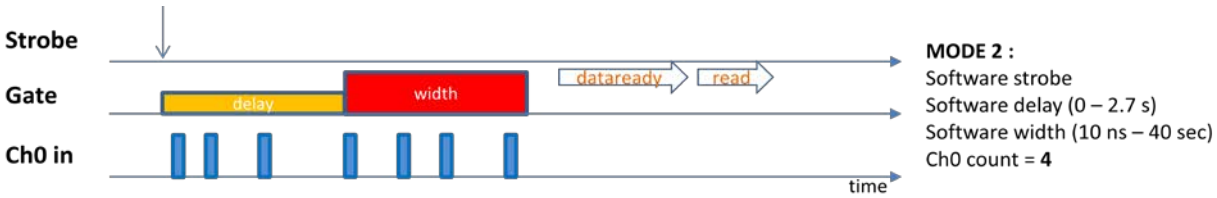

Figure 9 Mode 2 configuration

SCALER GATE WIDTH register [0xCEA00004].

- define the time width, bit [0..31] with time step of 10 ns per bit.

SCALER Configuration register [0xCEA00008])

- define the **delay**, bit [4..31] with time step of 10 ns per bit.
- the time width is defined by software, set the bit[0] to **0**
- set the **Immediate data send** bit[2] to **0**

STROBE Register [0xCEA00014]

- Set bit[0] to **1** send the software trigger, this bit is reset at **0** automatically.

Software gate is always allowed.

Range of gate width : . Minimum 10 ns - Maximum 42.0 s.

Range of delay : . . . . . Minimum 0 ns - Maximum 2.7 s.

The data is readable when the **SCALER GATE NUMBER Configuration register [0xCEA00010]** are equal to 0x00010001.

3.1.4. Mode 3, multi shot, external gate, external width

This mode act exactly like the single shot mode but can collect multiple gate into memory.
Up to **255 gates** could be collected into the memory before download the data. To set this option use the same configuration of **Mode 0.**

Figure 10 Mode 3 configuration

SCALER GATE NUMBER Configuration register [0xCEA00010].

- Define the number of gates to collect

Minimum gate width is 300 us.

3.1.5. Mode 4, multi shot, external gate, software width and delay

This mode acts exactly like the single shot mode but can collect multiple gates into memory. Up to 255 gates, **without dead time between each other**, could be collected into memory before download the data. The delay is always between the external trigger and the first internal gate.

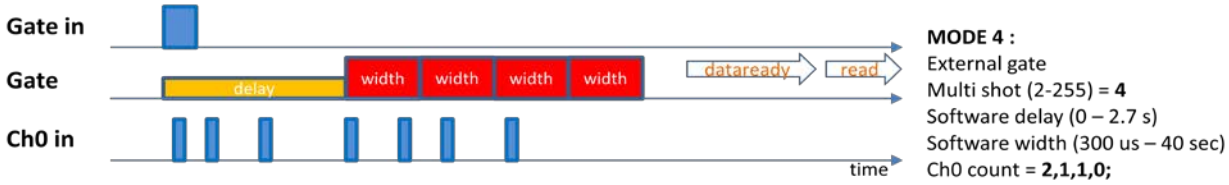

Figure 11 Mode 4 configuration

To set this option use the same configuration of **mode 1.**

SCALER GATE NUMBER Configuration register [0xCEA00010].

- Define the number of gates to collect

Range of gate width: . Minimum 300 us - Maximum 42.0 s.

Range of delay: . . . . . Minimum 0 ns - Maximum 2.7 s.

3.1.6. Mode 5, multi shot, software gate, software width and delay

This mode act exactly like the **mode 4** but the trigger is software. Up to 255 gates could be collected into memory before download the data.

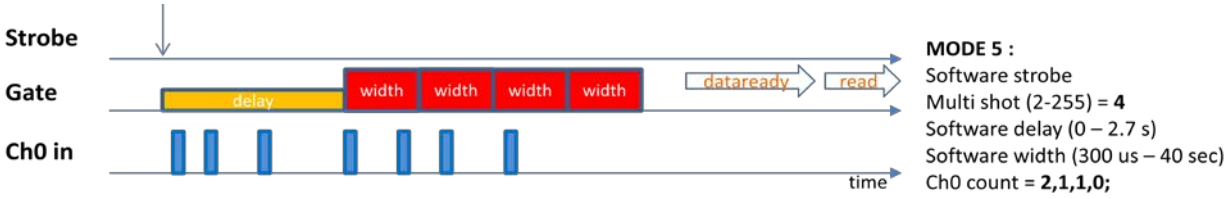

Figure 12 Mode 5 configuration

To set this option use the same configuration of the **mode 2** and

SCALER GATE NUMBER Configuration register [0xCEA00010].

- Define the number of gates to collect

Range of gate width: . Minimum 300 us - Maximum 42.0 s.

Range of delay: . . . . . Minimum 0 ns - Maximum 2.7 s.

3.1.7. Mode 6, multi shot, external gate, external width, auto send

This mode act exactly like the multi shot mode **(3)** but the maximum number of gates could reach **65535** and data are not collected into memory but data packet are sent immediately at the end of each gate.

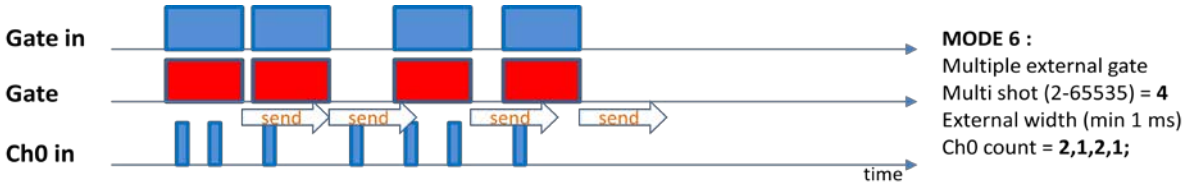

Figure 13 Mode 6 configuration

To set this option use the same configuration of the **mode 3** but the **Immediate data send** must be set.

SCALER Configuration register [0xCEA00008]:

- set the **Immediate data send** bit[2] to **1**

SCALER GATE NUMBER Configuration register [0xCEA00010].

- Define the number of gates to collect

The data are sent immediately after the end of each gate.
Sometimes the immediate data send could omit to send a data packet if the send buffer is full.
In this case the number of data packet received could be less than the number of gate set.

This inefficiency depends from the gate width and the number of gates chosen. In each packet is written the time stamp in microseconds and can be easily found the missing packet.

The minimum external gate width is 1 ms.

3.1.8. Mode 7, multi shot, external gate, software width, auto send

This mode act exactly like the multi shot mode (**mode 4**) but the number of gate are up to **65535** and data are not collected into memory but are sent immediately at the end of each gate .

To set this option use the same configuration of the **mode 3** but the **Immediate data send** must be set.

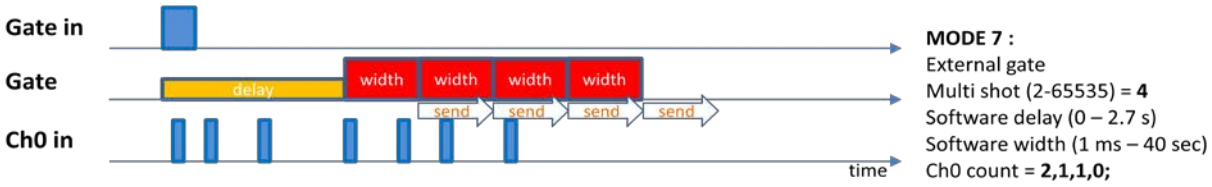

Figure 14 Mode 7 configuration

SCALER Configuration register [0xCEA00008]:

- set the **Immediate data send** bit[2] to **1**

SCALER GATE NUMBER Configuration register [0xCEA00010].

- Define the number of gates to collect

The data are sent immediately after the end of every gate.

Sometimes the immediate data send could omit to send a data packet if the send buffer is full.
In this case the number of data packet received could be less than the number of gate set.
This inefficiency depends from the gate width and the number of gates chosen. In each packet is written the time stamp in microseconds and can be easily found the missing packet.

Range of gate width: . Minimum 1 ms - Maximum 42.0 s.

3.1.9. Mode 8, multi shot, software gate, software width, auto send

This mode act exactly like the multi shot mode (**mode 7**) but the acquisition start with a software strobe.

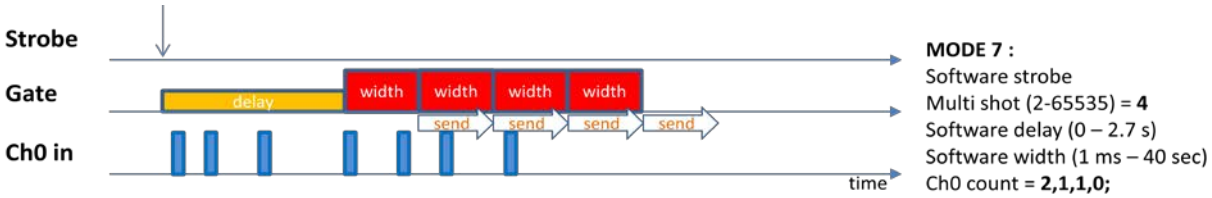

Figure 15 Mode 8 configuration

3.2. Tdc configuration

TDC can be configured using the **TDC Configuration register [0xCEA00000]**.

It could be configured to measure the arrival time on positive edge, negative edge or positive and negative edge to read the input signal width (set bit<8 to 9> of this register).

Every channel is a multi hit channel and the maximum number of hit could be set from 1 to 15 hit (set bit<4 to 7> of this register).

The tdc can work in common stop or in common start setting bit<0> of this register.

Also the acquisition window should be set using the **TDC GATE WIDTH register** at 10 ns per bit:

When data are ready to download (bit<16> of **TDC Configuration register [0xCEA00000]** are equal to '1'), read the procedure in character "TDC GEM TDC Memory Region", for data download and data frame detail.

For test purpose also a software strobe could be send.

3.2.1. Mode common start, positive or negative edge

In this mode the data are collected in common start. Select positive or negative edge on the input signal.

First data are always the nearest to start/stop signal. In this example data are collected as :

$Data = T^{pos}(0), T^{pos}(1), T^{pos}(2), T^{pos}(3)$

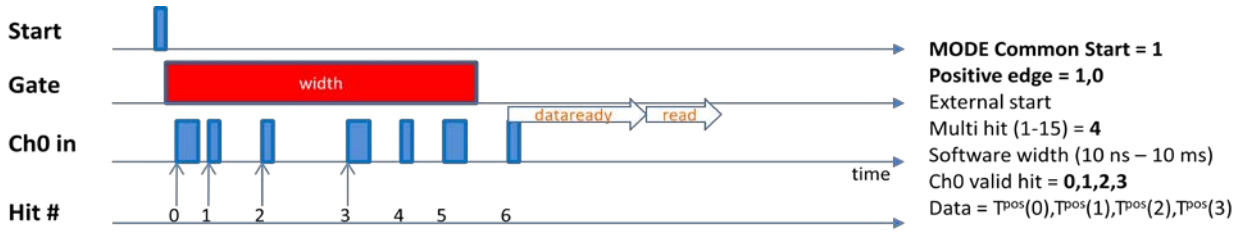

Figure 16 Mode common start positive edge

TDC CTRL register [0xCEA00000]

- set the **mode** bit[0] to **1**
- set the **number of hit** bit[4-7] to **4**
- set the **positive edge** bit[8] to **1**
- set the **negative edge** bit[9] to **0**

TDC GATE WIDTH register [0xCEA0000C]

- set the **width** bit[0-20] to **a value**, default is 1 us (0x64).

3.2.2. Mode common stop, positive or negative edge

In this mode the data are collected in common stop. Select positive or negative edge on the input signal.

First data are always the nearest to start/stop signal. In this example data are collected as :

$Data = T^{neg}(5), T^{neg}(4), T^{neg}(3), T^{neg}(2)$

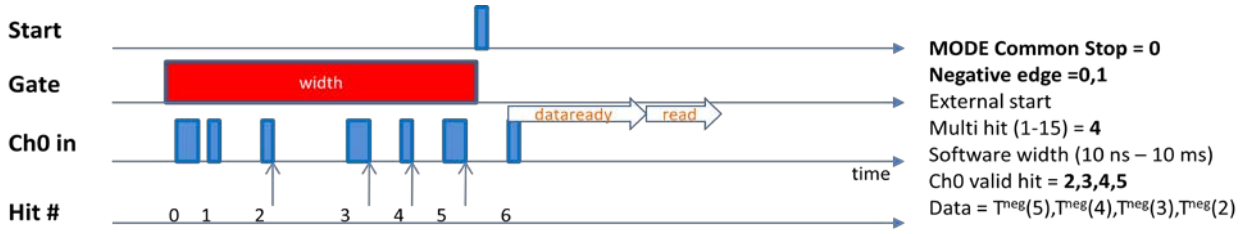

Figure 17 Mode common stop negative edge

TDC CTRL register [0xCEA00000]

- set the **mode** bit[0] to **0**
- set the **number of hit** bit[4-7] to **4**
- set the **positive edge** bit[8] to **0**
- set the **negative edge** bit[9] to **1**

TDC GATE WIDTH register [0xCEA0000C]

- set the **width** bit[0-20] to **a value**, default is 1 us (0x64).

3.2.3. Mode common start, positive and negative edge

In this mode the data are collected in common start. Select positive and negative edge on the input signal.

First data are always the nearest to start/stop signal. In this example data are collected as :

$Data = T^{pos}(0), T^{neg}(0), T^{pos}(1), T^{neg}(1)$

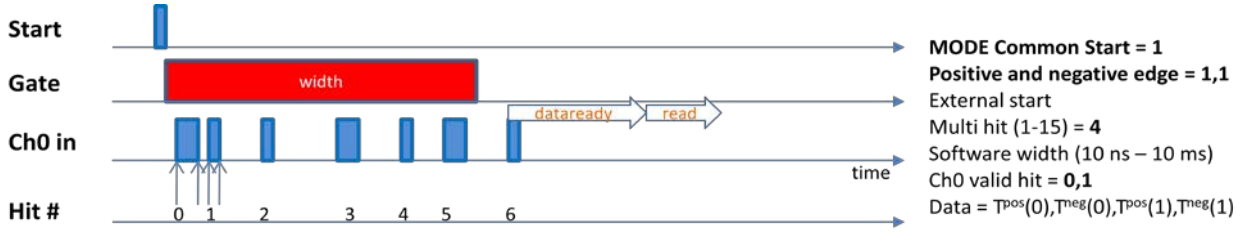

Figure 18 Mode common start positive and negative edge

TDC CTRL register [0xCEA00000]

- set the **mode** bit[0] to **1**

- set the **number of hit** bit[4-7] to **4**
- set the **positive edge** bit[8] to **1**
- set the **negative edge** bit[9] to **1**

TDC GATE WIDTH register [0xCEA0000C]

- set the **width** bit[0-20] to **a value**, default is 1 us (0x64).

3.2.4. Mode common stop, positive and negative edge

In this mode the data are collected in common stop. Select positive and negative edge on the input signal.

First data are always the nearest to start/stop signal. In this example data are collected as :

$$Data = T^{neg}(5), T^{pos}(5), T^{neg}(4), T^{pos}(4)$$

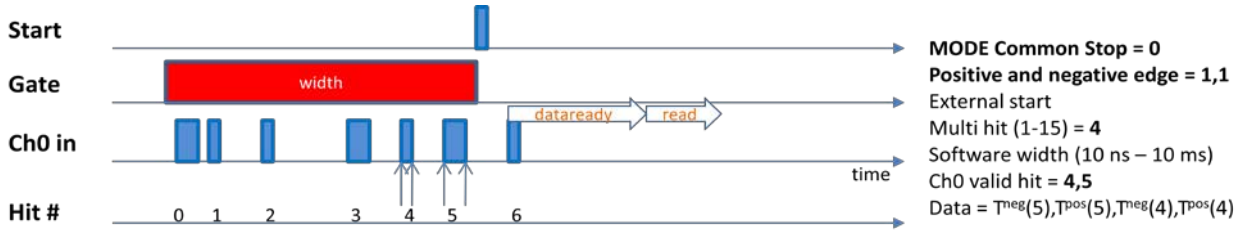

Figure 19 Mode common stop positive and negative edge

TDC CTRL register [0xCEA00000]

- set the **mode** bit[0] to **0**
- set the **number of hit** bit[4-7] to **4**
- set the **positive edge** bit[8] to **1**
- set the **negative edge** bit[9] to **1**

TDC GATE WIDTH register [0xCEA0000C]

- set the **width** bit[0-20] to **a value**, default is 1 us (0x64).