CMS

Trigger Sequencer Card

User Manual

VERSION 4.0

Institut de Physique Nucléaire de Lyon

1 Contents

2 FIGURES

3 TABLES

4 INTRODUCTION

4.1 GENERAL DESCRIPTION

The TSC (Trigger Sequencer Card) is intended to assume timing aspects in the CMS silicon tracker test stations.
It is presented in a PCI form factor for workstations (Figure 1).
Placed between trigger, clock source and FEC-CCU, it can also supply a laser pulser.
The trigger part assumes gating, counting and blocking following triggers. Internal clock and triggers are also provided.
The sequencer part matches all APV requests including trigger, calibration and reset.
All the necessary software for TSC set up is provided.

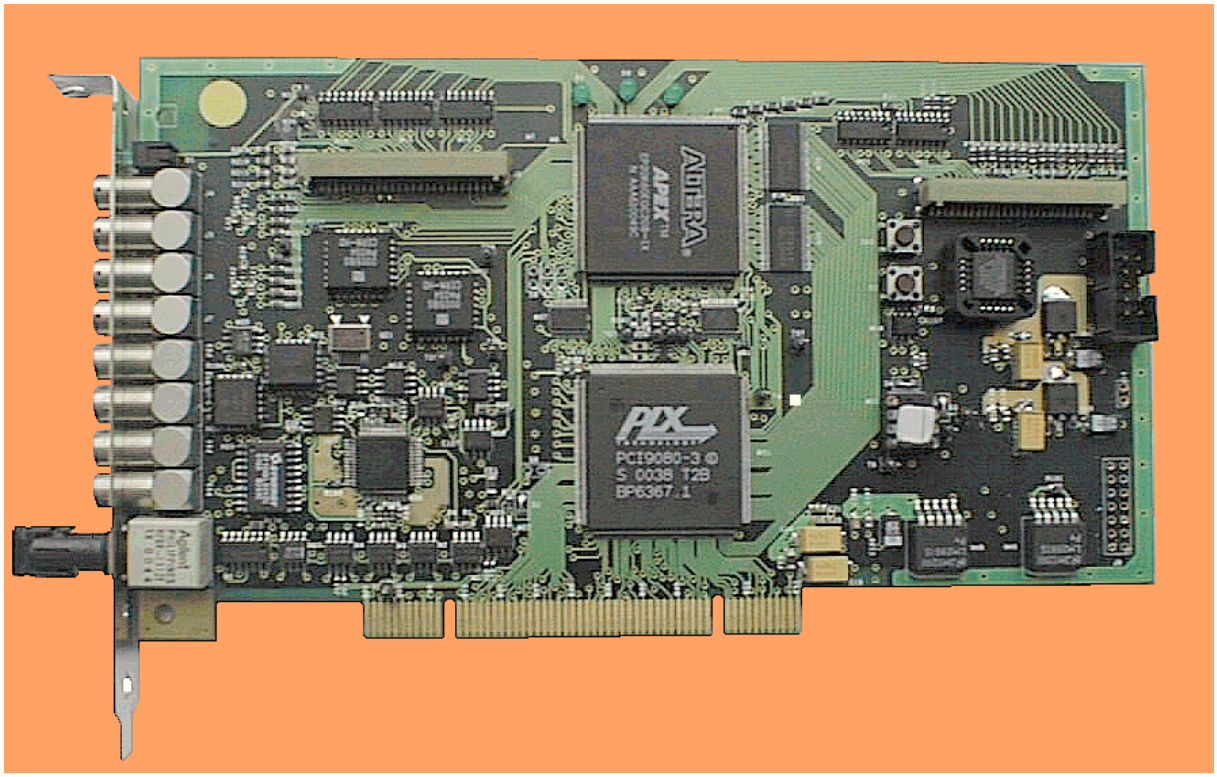

Figure 1 Photograph of TSC

4.2 ARCHITECTURE

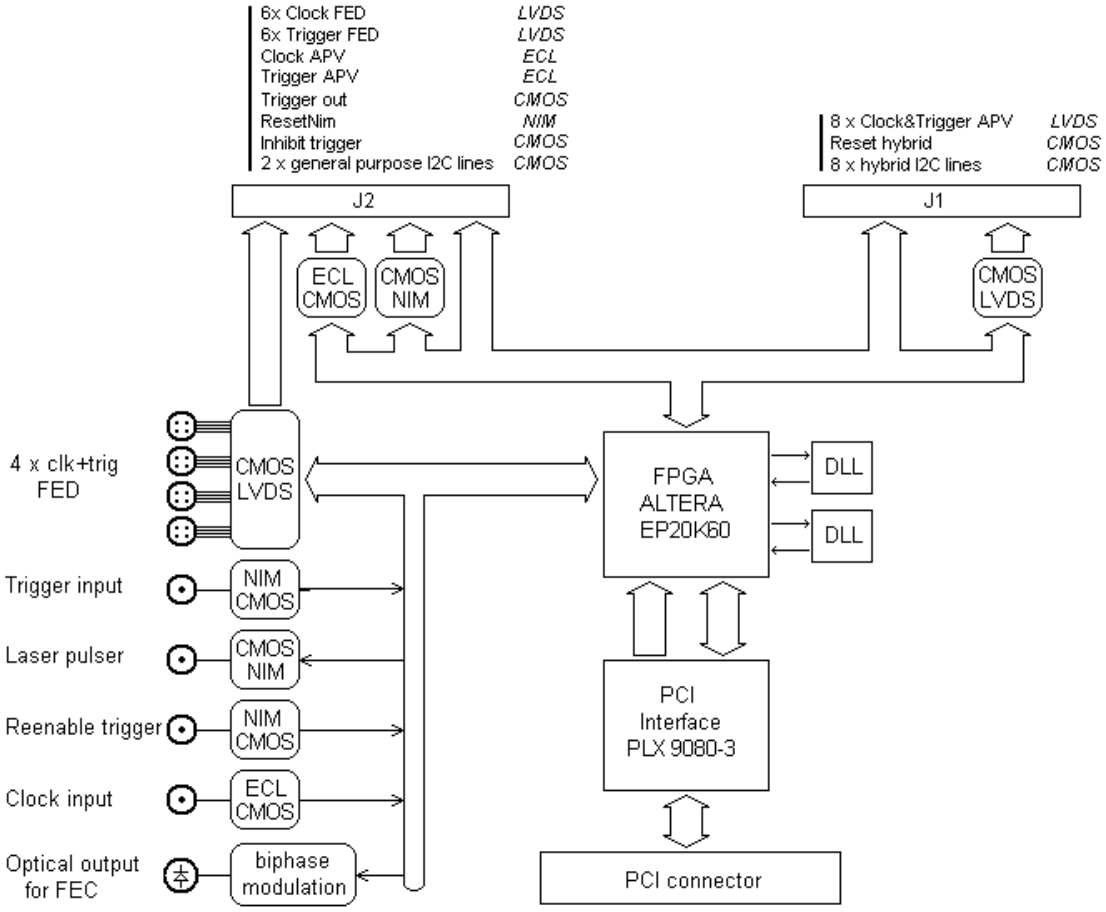

Figure 2 Block Diagram

Main parts of this card include a **PCI bridge** to interface PCI bus, a **FPGA** for trigger logic, allowing future modifications, **biphase modulator** for TTCrx, 2 **DLLs** for precise delays and several **level adaptors**.

4.3 CAUTION FOR HEATING

Because of ECL use, this card produces heat. Fanning is mandatory.

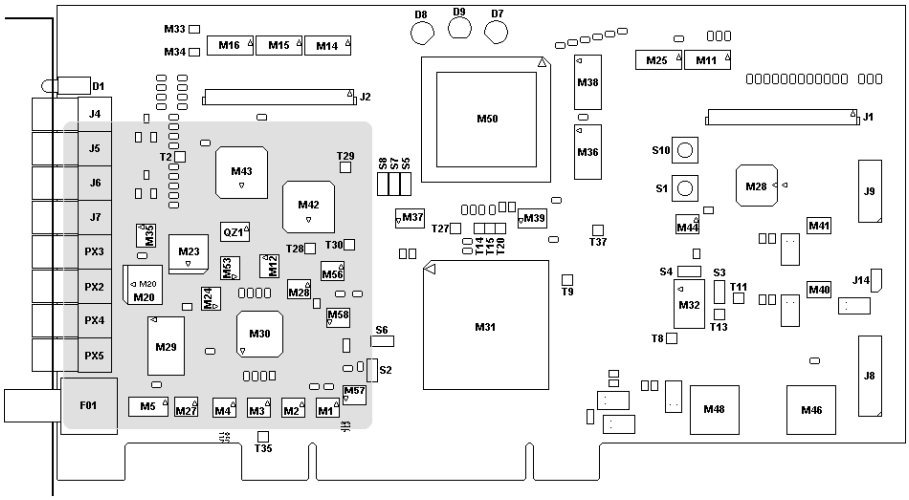

Figure 3 Area (grey) to be fanned

5 HARDWARE

5.1 FRONT PANEL

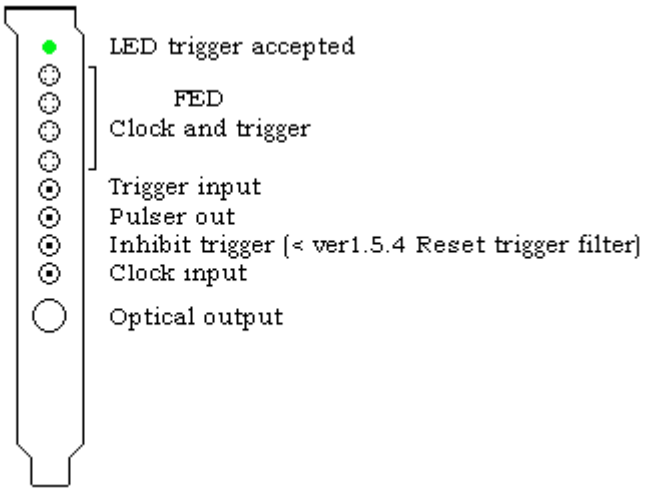

Figure 4. Front panel connectors

- **LED trigger accepted** : highlights each time a trigger is accepted and sent to APVs.
- **FED clock and trigger** (4x): LVDS clock and trigger outputs for FED(s).
- **Trigger input** : NIM input for external trigger.
- **Pulser out** : NIM output to trigger a pulser.
- **Inhibit trigger :** NIM input for hardware inhibit trigger. Previously (<1.5.4) was **Reset Trigger Filter** : NIM input for hardware Reset Trigger Filter.
- **Clock input** : ECL external 40 Mhz clock input.
- **Optical output** : Optical connection to the optical FEC input.

5.2 INSIDE CONNECTORS

Local supply J8. For debugging. HE10 14 pins connector

Pin number	Signal name	Description
1	VCC	+ 5 volts
2	VCC	+ 5 volts
3	VCC	+ 5 volts
4	GND	
5	GND	
6	GND	
7	GND	
8	GND	
9	GND	
10	VM12_PCI	-12 volts
11	VM12_PCI	-12 volts
12	VIO	Universal PCI supply
13	DVDD	+3.3 volts local from +5 volts
14	DVDD	+3.3 volts local from +5 volts

Table 1. Description of local supply connector

JTAG J9. HE10 10 pins connector.

Pin number	Signal name	Description
1	TCK	JTAG clock
2	GND	
3	TDO	JTAG data out
4	DVDD	+3.3 volts local from +5 volts
5	TMS	JTAG mode select
6	DVDD	+3.3 volts local from +5 volts
7	Nc	
8	Nc	
9	TDI	JTAG data in
10	GND	

Table 2. Description of JTAG connector

One of the PCI for workstations form factor's problem is the short front panel size. Because of this, two more SMC 50 pins connectors were added, J1 and J2.

See below tables 3 and 4 for description.

Pin number	Signal name	Description	Technology
1	HYB7_SDA	I2C data for hybrid #7	***
2	HYB7_SCL	I2C clock for hybrid #7	***
3	GND		
4	HYB6_SDA	I2C data for hybrid #6	***
5	HYB6_SCL	I2C clock for hybrid #6	***
6	GND		
7	HYB5_SDA	I2C data for hybrid #5	***
8	HYB5_SCL	I2C clock for hybrid #5	***
9	GND		
10	HYB4_SDA	I2C data for hybrid #4	***
11	HYB4_SCL	I2C clock for hybrid #4	***
12	GND		
13	HYB3_SDA	I2C data for hybrid #3	***
14	HYB3_SCL	I2C clock for hybrid #3	***
15	GND		
16	HYB2_SDA	I2C data for hybrid #2	***
17	HYB2_SCL	I2C clock for hybrid #2	***
18	GND		
19	HYB1_SDA	I2C data for hybrid #1	***
20	HYB1_SCL	I2C clock for hybrid #1	***
21	GND		
22	HYB0_SDA	I2C data for hybrid #0	***
23	HYB0_SCL	I2C clock for hybrid #0	***
24	GND		
25	RESET_HYB	General hybrid reset	***
26	GND		
27	GND		
28	DVDD	+ 3.3v digital output	
29	DVDD	+ 3.3v digital output	
30	DVDD	+ 3.3v digital output	
31	GND		
32	GND		
33			
34			
35			
36			
37			
38			
39			
40			
41			
42			
43			
44			
45			
46			
47	TRAPV0_H	Clock&Trigger	LVDS+
48	TRAPV0_L	Clock&Trigger	LVDS-
49	GND		
50	GND		

Table 3. Description of J1 connector

*** Open drain maximum 3.3 volts

Pin number	Signal name	Description	Technology
1	GND		
2	SDA0	I2C data for I2C line #0	***
3	GND		
4	SCL0	I2C clock for I2C line #0	***
5	GND		
6	SDA1	I2C data for I2C line #1	***
7	GND		
8	SCL1	I2C clock for I2C line #1	***
9			
10			
11			
12			
13	APV_trig_h	High level APV trigger	LVDS+
14	APV_trig _l	High level APV trigger	LVDS-
15	APV_clock_h	High level APV clock	LVDS+
16	APV_clock _l	High level APV clock	LVDS-
17			
18			
19			
20			
21	Fed_trig_h	High level fed trigger	LVDS+
22	Fed_trig_l	Low level fed trigger	LVDS-
23	Fed_clk_h	High level fed clock	LVDS+
24	Fed_clk_l	Low level fed clock	LVDS-
25			
26			
27			
28			
29	Fed_trig_h	High level fed trigger	LVDS+
30	Fed_trig_l	Low level fed trigger	LVDS-
31	Fed_clk_h	High level fed clock	LVDS+
32	Fed_clk_l	Low level fed clock	LVDS-
33	GND		
34	INHIBIT	Trigger inhibit input	CMOS In 3.3v
35	TRIGGER_OUT	Trigger accepted output	***
36	RES_DLLOUT3	Reserved CMOS output	***
37	SCLI2C_FPGA	Local I2C clock	***
38	SDAI2C_FPGA	Local I2C data	***
39	GND		
40	RESNIM	Reserved NIM output	NIM Out
41	GND		
42	GND		
43	CKAPV_ECLN	APV clock	ECL-
44	CKAPV_ECLP	APV clock	ECL+
45	GND		
46	GND		
47	TRAPV_ECLN	APV trigger	ECL-
48	TRAPV_ECLP	APV trigger	ECL+
49	GND		
50	GND		

Table 4. Description of J2 connector

*** Open drain maximum 3.3 volts

5.3 PCI

Bridge	PLX 9080-3
Mode	J
Connector	PCI J3
Clock Speed	<= 33MHz
Switching regime	5V
Bus width	32 bit
PCI Specification	v 2.1

5.4 FPGA

ALTERA EP20K60EQC208-1X

The FPGA is the heart of the TSC. It assumes all functions of the trigger and sequencer parts, except fine delays. It is reconfigurable in firmware and thereby maintains a flexible hardware architecture.

During normal operation the FPGA is loaded on power up from the on-board Flash memory.

5.5 EPROMs

- NM93CS46N (M32) : Serial CMOS EEPROM 64 x 16bits.
  It contains PCI configuration parameters, including the TSC serial number.

- EPC2LC20 (M28) : Flash EPROM containing the FPGA design file.

5.6 DELAY LINES

Type : CERN 00051BS PHOS4 Rev 1.2 from Microelectronics Group.

Description :
- ASIC providing 4 calibrated delay lines
- Step 1 nS
- I2C programmable

5.7 JUMPERS

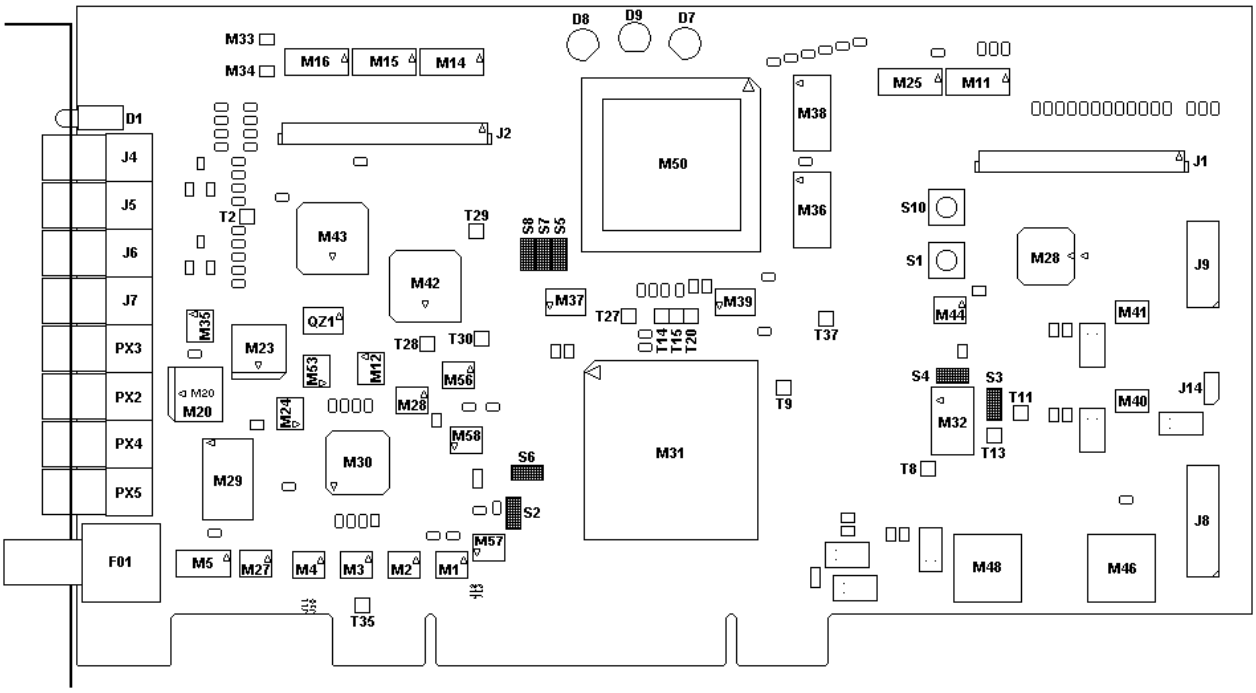

Figure 5. Jumpers

The user should not have to keep busy with jumpers.

- **PLX PCI bridge mode**

S2 : PLX normal mode : closed. PLX test mode :open. By default set to closed

- PLX EPROM

S3 : Serial EPROM protected mode : closed , non protected mode : open. By default set to closed

- **EPROMs programming**

	PLX EPROM Programmation mode	FPGA flash EPROM programmation mode
S4	Closed	Open
S5	Open	Closed
S6	Open	Closed
S7	Open	Closed
S8	Open	Closed

Table 5 : Jumper settings

5.8 BUTTONS

2 pushbuttons S1 and S10

S1 : Pushing this button
- asks for a FPGA reset.
- The FPGA returns to a cold start status.
- The contents of registers are lost.

S10 : For proper use, a PLL is used in the FPGA, pushing this button asks for this PLL reset inside the FPGA. Nothing to see with hybrid PLL.

5.9 LEDS

D1 is placed on the front panel, D7,D8 and D9 are placed close to the FPGA.

	Extingished	lighted
D1		Each time a trigger is sent to hybrid
D7	FPGA is initialized	FPGA is not initialized
D8	Not used	
D9	No clock is present	Clock OK

Table 6 : LEDs

5.10 SUPPLIES

Except VIO, only 5 volts is required for this card. 3.3 and 1.7 volts are made locally.

There was a modification on 3.3 volts. It is now made on a little mezzanine card pluged in J8. This mezzanine card has 2 functions :

- 3.3 volts regulation from 5 volts supply.
- Delayed relay to insure short rising edge of 3.3 volts for Delay Lines supply management.

6 TSC FUNCTIONALITIES

6.1 FONCTIONNAL BLOCK DIAGRAM

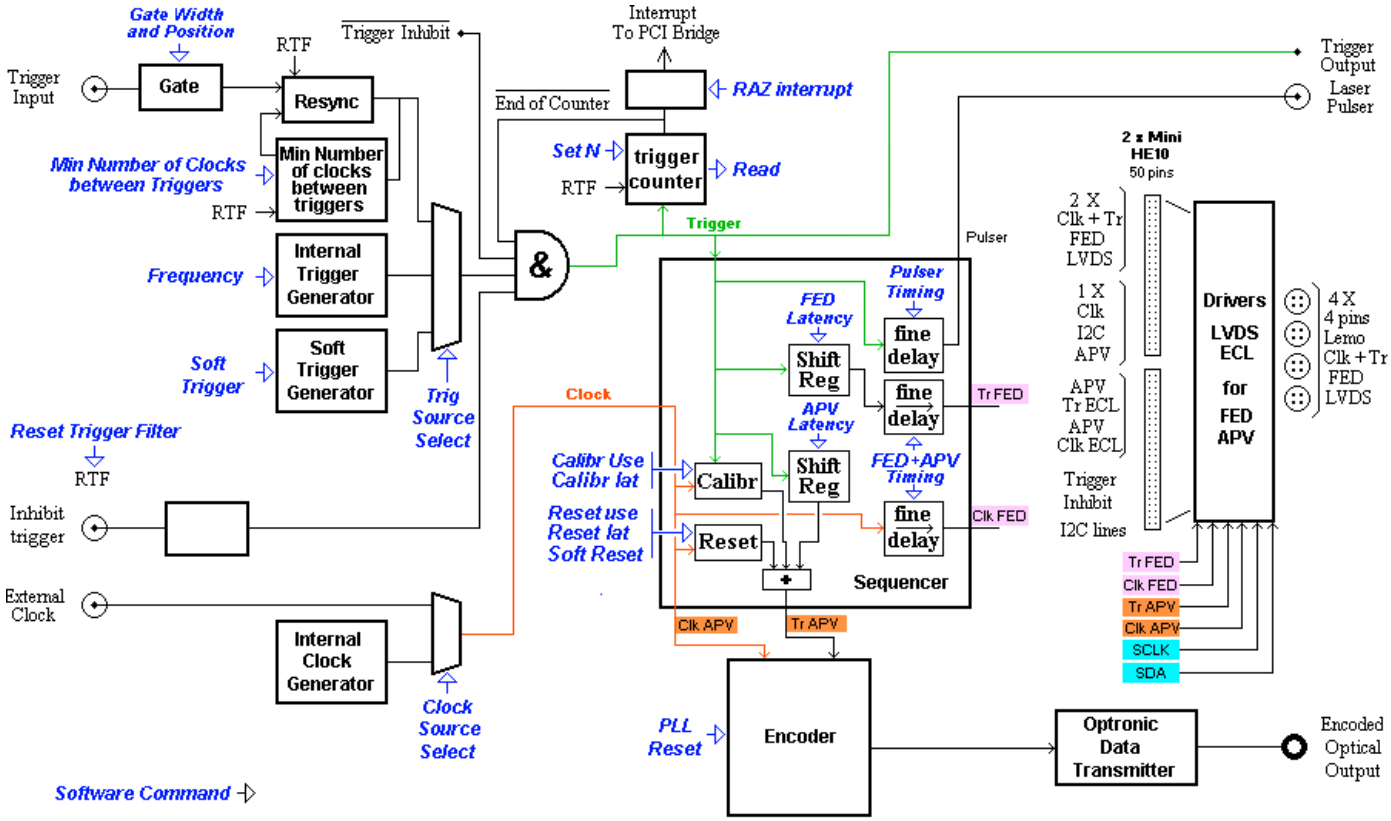

Figure 6. Fonctionnal block diagram

6.2 TRIGGER PART

3 ways of triggering exist :

- External trigger NIM input.
- Frequency programmable internal pulse generator.
- Software triggered pulse.

The trigger part is essentialy a trigger filter. Many functions are reset by Reset Trigger Filter.

6.2.1 TRIGGER SOURCE

- **Selecting internal or external source**.

  By writing bit3=0 and bit2=0 in *trigger register control* (PCIBAR2+0CH), external trigger is selected
  By writing bit3=1 and bit2=0 in *trigger register control* (PCIBAR2+0CH), internal trigger is selected

- **Selecting internal source frequency**

  The internal trigger frequency is selectable by steps of 20 microseconds.
  The value of the frequency (period) is written into *Internal trigger frequency register* (PCIBAR2+10H).

- **Selecting soft trigger**

  First, select soft trigger source by writing bit3=1 and bit2=1 in *trigger register control* (PCIBAR2+0CH).
  Each time a soft trigger is to be sent, write in *soft trigger register control* (PCIBAR2+34H).

6.2.2 TRIGGER FILTER

6.2.2.1 INHIBIT

The trigger can be inhibited by entering a logic level 0 on the inhibit pin (CMOS non 5 volts tolerant pin 34 connector J2).
A NIM input (previously Reset Trigger Filter) is now (>=1.5.4) an inhibit trigger input.
It can also be inhibited by writing bit1=1 in *trigger register control* (PCIBAR2+0CH).
At power on this function is disabled : bit1=0.

6.2.2.2 GATE

- **Theory of operation**

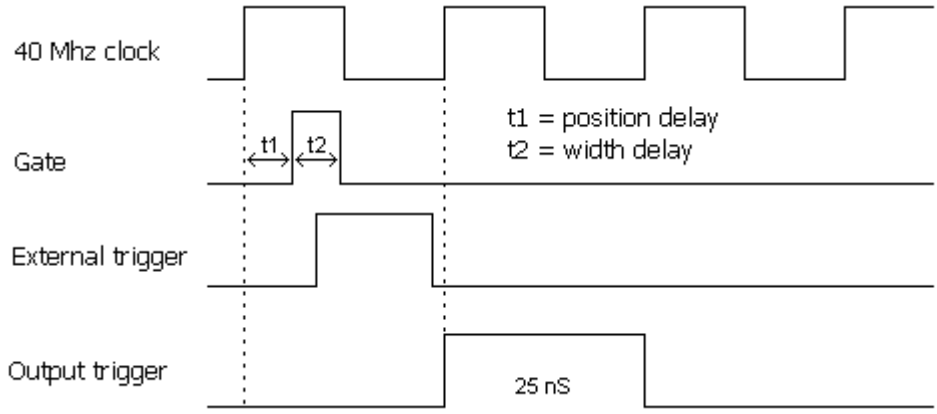

Figure 7. Trigger gate timing

A gate signal synchronized with the 40 Mhz clock is made localy, programmable in position and width by steps of 1nS. If the external trigger rising edge happens during the high level of this gate, the trigger is accepted and a signal *output trigger* (25nS duration) is generated in the following clock period.
DLLs are used to make the gate, position and width are 25 levels [0,24] programmable by steps of 1 nS.

Set position by writing bit[6:5] = 01 and position value in bit[4:0] of *Local I2C DLL1 register* (PCIBAR2+40H).
Set width by writing bit[6:5] = 11 and width value in bit[4:0] of *Local I2C DLL1 register* (PCIBAR2+40H).

This part of the filter may be inactivated by choosing position=0 and width=24.

- **Gate implementation.**

DLL1 (M42) is used : see figure 7
- Line 0 + line 1 for trigger gate position
- Line 2 + line 3 for trigger gate width

The DLLs have minimum signal delay. For trigger gate position and width, 2 lines are connected in series, one to calibrate (insure 25 nS exactly) and the other to be used as an exact delay. The first is calibrated automatically (see Software chapter), the second is set by the user.
Therefore, line 0 and 2 should be calibrated automatically while line 1 should be set for gate position and line 3 set for gate width.

All delay lines are set to 0 delay by the FPGA at power on.

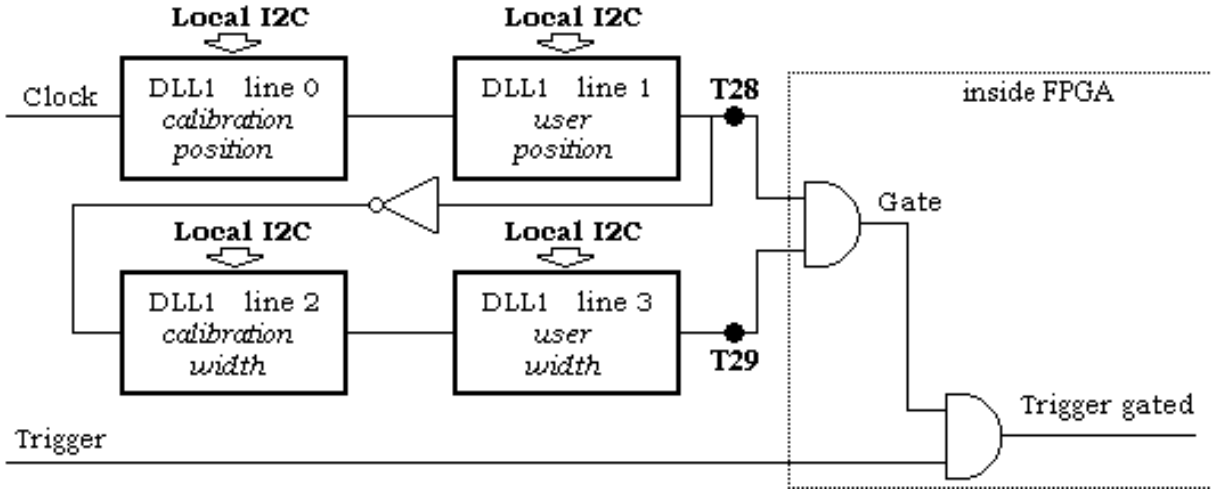

Figure 8. Trigger gate cabling

All software related to the gate management is located in :

~/Daq_Cms_Like/DeviceDriver/Tsc_Driver/gate/

The tar file can be found at the following address :

http://lyoinfo.in2p3.fr/cms/cmstraces/tscweb/software/gate.tar

Gate calibration

Tools are proposed in order to calibrate gate width and position.

For this purpose, a test setting was implemented in the FPGA (see figure 8). This mechanism switches 40 Mhz clock on 2 lines :

- One connected to a Non Gated Trigger Counter through a synchronisation circuit.
- The other is connected to a Gated Trigger Counter through the gate circuit and a synchronisation circuit.

Thereby, it becomes easy to check if the gate is transparent or not.

This mechanism is triggered by writing in *Gate Delay Calibration Request register* (PCIBAR2+50H). Scanning *Local I2C Status register* (PCIBAR2+44H) bit8=1 means calibration is done.
The counters are readable in *Local I2C Status register* (PCIBAR2+44H). The content of Non Gated Trigger Counter is in bit[15:12], the content of Gated Trigger Counter is in bit[19:16]

By reading these counters while scanning for different values of gate width and position, it is possible to calibrate the gate.
A calibrating function, using these features, is implemented in the constructor of TscDevice class.

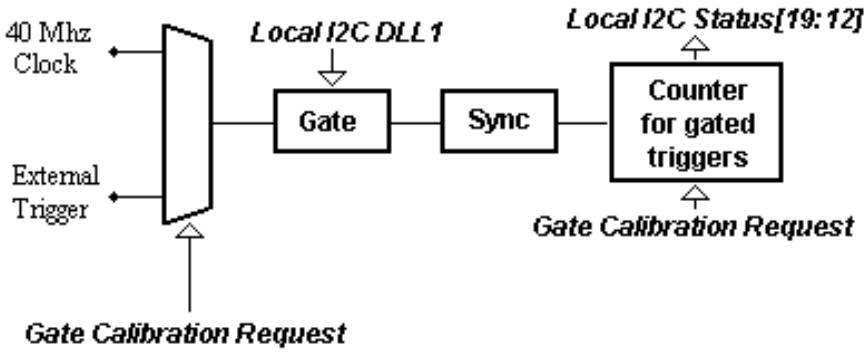

Figure 9. Trigger gate calibration

6.2.2.3 MINIMUM NUMBER OF CLOCKS BETWEEN TRIGGERS

In order not to disturb APV sequence, the number of clocks between triggers must be decided. A counter assumes this. This counter is reset by Reset Trigger Filter. It acts only for external triggers. The minimum is set to 3, it is programmable up to 65535 by writing the value in *Minimum number of clocks register* (PCIBAR2+4H).

6.2.2.4 COUNTING

The last stage of the trigger filter is a counter. It may be used to manage the spill size of the ADC. This counter is fully programmable from 0 to 65535 by writing the value in *Trigger maximum count register* (PCIBAR2+8H).
After initialization and during acquisition, the counter is incremented of 1 each trigger, the current value can be read in *Trigger counter current value* (PCIBAR2+28H).
When the current read value reaches the initial set value, the triggers are blocked and an interrupt is sent.
Note that the set value 0 is reserved for no counting, in this case the counter is transparent and there is no need to reset the trigger filter.

new

6.2.2.5 TRIGGER STAMP COUNTER AND FIFO

A 32 bits counter has been added. This counter runs at 40 MHz. It is reset by the resetFED command. When a trigger occurs, the counter value is loaded in a FIFO. The FIFO is 512 deep.

FIFO management

- FIFO status : Read the *Trigger stamp FIFO status register* (PCIBAR2+74H)
- FIFO clear : Write to *Clear Trigger stamp FIFO register* (PCIBAR2+74H)
- FIFO read : Read *Trigger stamp FIFO content register* (PCIBAR2+78H)

6.2.2.6 RESET TRIGGER FILTER

When the trigger counter is set to a value different from 0, the triggers are blocked when the counter reaches this set value. Then, to go on, a hardware or software Reset Trigger Filter order has to be given.

- **Software reset** : write into *Reset Trigger Filter register* (PCIBAR2+2CH).
- **Hardware reset** : NIM pulse in the Reset Trigger Filter input.
  The hardware reset is active at the high level (-800 mV) of the pulse. Note that the trigger filter is transparent during the high level of the pulse, this is why this reset pulse has to be short. The minimum length is 25 nS. **NO MORE AVIALABLE**

6.3 CLOCKS

The clock source can be choosen between both an ECL external clock input and a 40 Mhz internal clock generator.
The choice is done by writing in *Trigger register control* (PCIBAR2+CH) bit6=1 for external source and bit6=0 for internal source.
At power up the internal source is selected.
If external source is selected, the input has always to be fed, if it is not the case, the 1 nS delay in the DLLs is no more garanteed. A cold start is then necessary.

6.4 SEQUENCER PART

The sequencer has :

- 2 inputs : clock and trigger from the trigger filter.
- 5 outputs : APV clock and trigger, FED clock and trigger, pulser.

The APV trigger output of the sequencer can hold trigger, calibration and reset sequences. Refer to APV user manual for details.

6.4.1 TRIGGER SEQUENCE

This is the trigger input of the sequencer delayed of an entire number of clocks called APV latency. The APV latency can be updated by writing the latency value [1, 65535] in the *APV latency register* (PCIBAR2+18H).

6.4.2 CALIBRATION SEQUENCE

The calibration sequence is a 110 sequence, it comes an entire number of clocks after the trigger input of the sequencer. This number is called Calibration latency.
The Calibration latency can be updated by writing the latency value [1, 65535] in the *Calibration latency register* (PCIBAR2+38H).
First, the output of this Calibration sequence must be enabled by writing bit4=1 in the *trigger register control* (PCIBAR2+0CH).

6.4.3 RESET SEQUENCE

- **Reset sequence hardware trigged**
  The reset sequence is a 101 sequence, it comes an entire number of clocks after the trigger input of the sequencer. This number is called Reset latency.
  The Reset latency can be updated by writing the latency value [1, 65535] in the *Reset latency register* (PCIBAR2+0).
  First, the output of this Reset sequence must be enabled by writing bit5=1 in the *trigger register control* (PCIBAR2+0CH).

- **Reset sequence software trigged**
  A reset sequence can be asked at any time by writing in the *Soft APV reset register* (PCIBAR2+28H).
  Typically, this is for updating slow control values in the APV.

6.4.4 FED CLOCK AND TRIGGER

The TSC can drive up to 10 FED clock and trigger, 4 via the front panel and 6 via J2 connector.

- **FED trigger latency**
  In order to use correctly the FED without header finding, the FED trigger output is to be delayed a certain number (FED latency) of clocks.
  This is done by writing latency value [1, 65535] in *FED trigger latency register* (PCIBAR2+14H).
  Note that a reset sequence 101 (same as the one for APV) is sent each time a soft APV reset is asked.

- **Fine FED clock delay**
  The FED clock is tunable in fine delay from 0 to 24 by steps of 1 nS. The DLL2 (M43) line 0 is used for this purpose.

  Set fine FED clock delay by writing bit[6:5] = 00 and delay value [0,24] in bit[4:0] both in *I2CDLL2 register* (PCIBAR2+4CH).

- **Fine FED trigger delay**
  The FED trigger is also tunable in fine delay from 0 to 24 by steps of 1 nS. The DLL2 (M43) line 1 is used for this purpose.

  Set fine FED trigger delay by writing bit[6:5] = 01 and delay value [0,24] in bit[4:0] both in *I2CDLL2 register* (PCIBAR2+4CH).

In order to reset the FED trigger counter, a reset sequence 101 is sent by writing in *FED soft reset register* (PCIBAR2+68H). In the same time, an APV Soft Trigger Reset is also sent.

All delay lines are set to 0 delay by the FPGA at power on.

6.5 ENCODER PART

This encoder is intended to receive the APV clock and trigger outputs of the sequencer in order to encode them by a biphase modulation. This modulated signal supplies the optical transmitter connected to the FEC.
This part is the same as the TTCvx one, also implemented in ECL technology. This is why heating problems must be solved by a fan.
To produce 40, 80 and 160 Mhz, a PLL (M30) is used. Sometimes it has to be resetted, this is done by writing in *Reset encoder PLL register* (PCIBAR2+30H).

6.6 PULSER

The TSC can drive a pulser. This is the trigger input of the sequencer delayed from 0 to 24 by steps of 1 nS. The DLL2 (M43) line 2 is used for this purpose.
This is done by writing bit[6:5] = 10 and delay value [0, 24] in bit[4:0] both in *I2CDLL2 register* (PCIBAR2+4CH).

6.7 FIRMWARE

The firmware is the FPGA design file. It is loaded in the flash EPROM.
If updates are necessary, the content of the flash EPROM can be updated by the JTAG connector (J8).
In future versions it will be updated through PCI bridge.
The version number can be read in *Firmware version register* (PCIBAR2+7CH).
Bit[3:0] means subsubversion number , bit[7:4] means subversion number , bit[11:8] means version number.

6.8 INTERRUPTS

An interrupt is sent through the PCI bridge each time an end of trigger counter is reached, for this, the interrupts must be enabled by writing bit0=1 in Trigger control register (PCIBAR2+CH).
When the interrupt rises, it has to be catched as soon as possible. Just after, it has to be reset by writing in *RAZ interrupt register* (PCIBAR2+3CH).
Anyway, if it is not seen by the interrupt controler, it is reset automatically after 1.6 millisecond.

6.9 RESETS

- **PCI reset**, **Reset Button (S1)**
  - asks for a FPGA reset.
  - The FPGA returns to a cold start status.
  - The contents of registers are lost.

- **Power on/off cycle**: The PCI bridge loads the configuration parameters from the serial EEPROM M32. It also produces a PCI reset condition, resulting in a reload of FPGA.

6.10 I2C BUSES

2 I2C buses are held by the FPGA :

6.10.1 LOCAL I2C

An I2C bus for local needs, controlling DLL1 and DLL2. It can be accessed by registers 40H through 4CH.

6.10.2 EXTERNAL I2C

An I2C bus connected to outside by J1 and J2.
- 8 lines are connected by J1, they are destinated to hybrids.
- 2 lines are connected by J2, they are destinated to general purpose material.

It can be accessed by registers 54H through 64H.

In both cases, the lines are connected directly to the FPGA. The outputs are 3.3 volts and the inputs are **not 5 volts tolerant**.

Using external I2C

- **Select mode**

The external I2C system assumes 1 through 4 byte, write and read functions plus RAL mode (APV I2C). The mode is selectable in *External I2C Mode register* (PCIBAR2+54H).

- **Select address and channel**

There are 10 external I2C channels switchable (see table 8).
I2C address is specified in bit[7:1] and I2C channel in bit[11:8] of *External I2C Address + Channel register* (PCIBAR2+58H).

- **Writing data**

Whatever mode is used, data is written in *External I2C Data To Send register* (PCIBAR2+5CH). The MS byte(s) are filled with zeros.
Note that MSB of data is sent first in the I2C frame.

- **Reading data**

Whatever mode is used, data is read in *External I2C Data Received register* (PCIBAR2+60H).
Note that MSB of data word is read first in the I2C frame.

- **Reading status**

Reading bit0 of *External I2C Status register* (PCIBAR2+64H) = 0 means busy state (transaction in progress), = 1 means external I2C bus ready.

Write sequence

1) Wait for I2C ready, *External I2C Status register* bit0=1
2) Select mode in *External I2C Mode register*
3) Write data in *External I2C Data To Send register*
4) Write address and channel in *External I2C Address + Channel register.*
This last access triggers the write transaction.

Read sequence

1) Wait for I2C ready, *External I2C Status register* bit0=1
2) Read data in *External I2C Data Received register.*
This last access triggers the read transaction.

I2C channel number	Type	Connector	Pin number
0	SCL	J1	23
	SDA	J1	22
1	SCL	J1	20
	SDA	J1	19
2	SCL	J1	17
	SDA	J1	16
3	SCL	J1	14
	SDA	J1	13
4	SCL	J1	11
	SDA	J1	10
5	SCL	J1	8
	SDA	J1	7
6	SCL	J1	5
	SDA	J1	4
7	SCL	J1	2
	SDA	J1	1
8	SCL	J2	8
	SDA	J2	6
9	SCL	J2	4
	SDA	J2	2

Table 8 Distribution of external I2C channels

7 SOFTWARE

For users who don't want to worry with register accesses and PCI configuration stuff, see directly §7.5

7.1 PCI CONFIGURATION AND ADDRESS MAP

7.1.1 MEMORY REQUIREMENTS

PCI configuration space header type 0

The card has the following PCI space requirements :

PCI Configuration Space 64 bytes
PCI Memory Space 128 bytes

PCI I/O Space is not used.
PCI Interrupts are used.
No Bus Master behavior

All register accesses must be **long word** (32 bits) access, even if one bit, byte or word is necessary. This concerns write and read accesses.

7.1.2 PCI CONFIGURATION REGISTERS

See the table 9 for the list of registers called during the PCI configuration at the driver level.

Register Name	PCI-CFG address	Value	Comments
Vendor ID	00H	10DCH	CERN ID. Value read from serial EPROM
Device ID	02H	CEF0H	Value read from serial EPROM
PCIREV	08H	EPROM defined	PCI revision ID register. Loaded from the serial EPROM, this value is interpreted as the card serial number
PCIBAR0	10H	System defined	Base address register for PCI memory space accesses to PCI9080 bridge local registers
PCIBAR2	18H	System defined	Base address register for PCI memory space accesses to TSC registers
PCIILR	3CH	System defined	PCI Interrupt Line register

Table 9 PCI bridge configuration registers

Vendor ID : CERN number (10DCH) is used.
Device ID : CEF0H

Vendor ID, device ID, revision ID are taken from EPROM at boot time.

7.1.3 BRIDGE INITIALIZATION

PCIBAR0 is remapped to an address during device opening, local configuration register offsets are relative to this address.
See the table 10 for the list of registers called during the local bridge configuration.

Register Name	PCI address offset	Value	Comments
MARBR	08H	1200000H	Mode arbitration register. Bit21 : Local Bus Direct Slave Give up Bus Mode Bit24 : Delayed Transaction Mode
INTCSR	68H	F010100H	Interrupt control/status register Bit 8 : PCI Interrupt Enable Bit11: PCI Local Interrupt Enable Bit16 : Local Interrupt Output Enable Bit27-24 : ! Target abort

Table 10 Local bridge configuration registers

7.2 I/O MAPPING

Address	Read function	Write function
PCIBAR2	Reset latency	Reset latency [1, 65535]
PCIBAR2 + 4H	Minimum number of clocks	Min number of clocks [3, 65535]
PCIBAR2 + 8H	Trigger maximum count	Trigger maximum count [1, 65535]
PCIBAR2 + CH	Trigger register control	Trigger register control
PCIBAR2 + 10H	Internal trigger frequency	Internal trigger frequency
PCIBAR2 + 14H	FED trigger latency	FED trigger latency
PCIBAR2 + 18H	APV latency	APV latency
PCIBAR2 + 1CH		
PCIBAR2 + 20H		
PCIBAR2 + 24H		
PCIBAR2 + 28H	Trigger counter current value	APV soft reset
PCIBAR2 + 2CH		Reset trigger filter
PCIBAR2 + 30H		Reset encoder PLL
PCIBAR2 + 34H		Soft trigger
PCIBAR2 + 38H	Calibration latency	Calibration latency
PCIBAR2 + 3CH		RAZ interrupt
PCIBAR2 + 40H	Local I2C DLL1 (gate)	Local I2C DLL1 (gate)
PCIBAR2 + 44H	Local I2C status	
PCIBAR2 + 48H		Local I2C controller init
PCIBAR2 + 4CH	Local I2C DLL2 (FED, pulser)	Local I2C DLL2 (FED, pulser)
PCIBAR2 + 50H		Gate delay calibration request
PCIBAR2 + 54H	External I2C mode + reset	External I2C mode + reset
PCIBAR2 + 58H	External I2C address + channel	External I2C address + channel
PCIBAR2 + 5CH	External I2C data to send	External I2C data to send
PCIBAR2 + 60H	External I2C data received	
PCIBAR2 + 64H	External I2C status	
PCIBAR2 + 68H		FED soft reset
PCIBAR2 + 6CH	Hybrid Reset	Hybrid Reset
PCIBAR2 + 70H		
PCIBAR2 + 74H	Trigger stamp FIFO status	Clear Trigger stamp FIFO
PCIBAR2 + 78H	Trigger stamp FIFO content	
PCIBAR2 + 7CH	Firmware version	

Table 11 Summary of the register address mapping

All register offsets are in local space 0 and referenced to PCIBAR2.

7.2.1 REGISTERS

PCIBAR2+0H Reset latency R/W

Bit	Description	Read	Write	Value after Reset
15:0	Reset latency value [1, 65535]	Yes	Yes	0
31:16	Reserved	Yes	Yes	0

PCIBAR2+4H Minimum number of clocks R/W

Bit	Description	Read	Write	Value after Reset
15:0	Minimum number of clocks value [3, 65535]	Yes	Yes	0
31:16	Reserved	Yes	Yes	0

PCIBAR2+8H Maximum trigger count R/W

Bit	Description	Read	Write	Value after Reset
15:0	Maximum trigger counter value [0, 65535]	Yes	Yes	0
31:16	Reserved	Yes	Yes	0

PCIBAR2+CH Trigger register control R/W

Bit	Description	Read	Write	Value after Reset
0	Interrupt 1 : Enable 0 : Disable	Yes	Yes	0
1	Inhibit 1 : Enable 0 : Disable	Yes	Yes	0
3:2	00 No trigger 01 Select external trigger 10 Select internal trigger 11 Select soft trigger	Yes	Yes	00
4	Calibration 1 : Enable 0 : Disable	Yes	Yes	0
5	Reset 1 : Enable 0 : Disable	Yes	Yes	0
6	Clock 1 : External 0 : Internal	Yes	Yes	0
7	Gate shortcut 1 : gate pass all 0 : normal gate	Yes	Yes	0
8	Reserved	Yes	Yes	0
9	Inhibit source 1 : FIFO 0 : normal	Yes	Yes	0
10	1 : Use APV emulator 0 : Don't use			
31:11	Reserved	Yes	Yes	0

PCIBAR2+10H Internal trigger frequency R/W

Bit	Description	Read	Write	Value after Reset
15:0	Number of 10 microseconds periods [1, 65535]	Yes	Yes	1
31:16	Reserved	Yes	Yes	0

PCIBAR2+14H FED trigger latency R/W

Bit	Description	Read	Write	Value after Reset
15:0	FED trigger latency value [1, 65535]	Yes	Yes	0
31:16	Reserved	Yes	Yes	0

PCIBAR2+18H APV latency R/W

Bit	Description	Read	Write	Value after Reset
15:0	APV latency value [1, 65535]	Yes	Yes	0
31:16	Reserved	Yes	Yes	0

PCIBAR2+1CH
PCIBAR2+20H
PCIBAR2+24H

PCIBAR2+28H APV Soft reset Write only

Bit	Description	Read	Write	Value after Reset
31:0	APV Soft reset	No	Yes	0

PCIBAR2+28H Trigger counter current value Read only

Bit	Description	Read	Write	Value after Reset
15:0	Trigger counter current value	Yes	No	0
31:16	Reserved	Yes	No	0

PCIBAR2+2CH Reset trigger filter Write only

Bit	Description	Read	Write	Value after Reset
31:0	Reset trigger filter	No	Yes	0

PCIBAR2+2CH Trigger status register (for debug) Read only

Bit	Description	Read	Write	Value after Reset
15:0	Trigger status register (answers DEADH)	Yes	No	DEAD
31:16	Reserved	Yes	No	0

PCIBAR2+30H Reset encoder PLL Write only

Bit	Description	Read	Write	Value after Reset
31:0	Reset encoder PLL	No	Yes	0

PCIBAR2+34H Soft trigger generation Write only

Bit	Description	Read	Write	Value after Reset
31:0	Output one soft trigger pulse	No	Yes	0

PCIBAR2+38H Calibration latency R/W

Bit	Description	Read	Write	Value after Reset
15:0	Calibration latency value [1, 65535]	Yes	Yes	0
31:16	Reserved	Yes	Yes	0

PCIBAR2+3CH RAZ interrupt Write only

Bit	Description	Read	Write	Value after Reset
31:0	RAZ interrupt	No	Yes	0

PCIBAR2+40H Local I2C DLL1 (gate) R/W

Bit	Description	Read	Write	Value after Reset
4 :0	Delay value [0, 24]	No	Yes	0
6 :5	Line 0 : 00 position calibration Line 1 : 01 user gate position Line 2 : 10 width calibration Line 3 : 11 user gate width	No	Yes	00
31:7	Reserved	No	Yes	0

Proper address of DLL1 chip is hardware coded.
The DLL chips are not readable, therefore, this R/W register returns the last write values not necessarily the effective ones that stand in the DLL chip.

PCIBAR2+44H Local I2C status Read only

Bit	Description	Read	Write	Value after Reset
2 :0	Reserved	Yes	No	000
3	0 : busy 1 : DLL ready for new I2C transaction	Yes	No	0
7 :4	Reserved	Yes	No	0000
8	0 : Calibration not done 1 : Calibration done	Yes	No	0
11:9	Reserved	Yes	No	000
19:12	Gated trigger counter value	Yes	No	0
31:20	Reserved	Yes	No	0

PCIBAR2+48H Local I2C controller init (not used) Write only

Bit	Description	Read	Write	Value after Reset
31:0	Reserved	No	Yes	

PCIBAR2+4CH Local I2C DLL2 R/W

Bit	Description	Read	Write	Value after Reset
4 :0	Delay value [0, 24]	No	Yes	0
6 :5	Line 0 : 00 FED clock delay Line 1 : 01 FED trigger delay Line 2 : 10 Pulser delay Line 3 : 11 reserved (connector J2 pin 36)	No	Yes	00
31:7	Reserved	No	Yes	0

PCIBAR2+50H Gate delay calibration request Write only

Bit	Description	Read	Write	Value after Reset
31:0	Gate delay calibration	No	Yes	0

PCIBAR2+54H External I2C mode R/W

Bit	Description	Read	Write	Value after Reset
3:0	0000 Single byte write 0001 Single byte read 0010 Double byte write 0011 Double byte read 0100 Triple byte write 0101 Triple byte read 0110 Quadruple byte write 0111 Quadruple byte read 1001 1 byte write + read (RAL mode)	Yes	Yes	0000
4	1 : Reset external I2C controller, has to be released to 0	Yes	Yes	0
31:4	Reserved	Yes	Yes	0

PCIBAR2+58H External I2C address + channel R/W

Bit	Description	Read	Write	Value after Reset
0	Reserved	Yes	Yes	0
7:1	I2C address	Yes	Yes	0000
11:8	I2C channel [0, 9] (see table 7)			0000
31:12	Reserved	Yes	Yes	0

PCIBAR2+5CH External I2C data to send R/W

Bit	Description	Read	Write	Value after Reset
31:0	Data to send	Yes	Yes	0

PCIBAR2+60H External I2C data received Read only

Bit	Description	Read	Write	Value after Reset
31:0	Data received	Yes	No	0

PCIBAR2+64H External I2C status Read only

new

Bit	Description	Read	Write	Value after Reset
0	0 : busy 1 : Ready for new I2C transaction	Yes	No	0
1	0 : last operation not succeeded 1 : last operation succeeded	Yes	No	0
3:2	Reserved for internal use	Yes	No	000
4	Last acknowledgement bit status	Yes	No	0
31:5	Reserved	Yes	No	0

PCIBAR2+68H FED Soft reset Write only

Bit	Description	Read	Write	Value after Reset
31:0	FED Soft reset	No	Yes	0

PCIBAR2+6CH Hybrid Reset R/W

Bit	Description	Read	Write	Value after Reset
0	RESET_HYB : Pin 25 of connector J1 = Value of Bit 0	Yes	Yes	
31:1	Reserved	Yes	Yes	0

PCIBAR2+70H APV emulator counter (for clock number , multiply by 280)
This counter is incremented (+280) each accepted trigger, is decremented (-1) each clock.
Trigger is stopped if counter > *stop**280, is released if counter < *release**280

Bit	Description	Read	Write	Value after Reset
31:16	Number of triggers for *release*		Yes	0
15 :0	Number of triggers for *stop*		Yes	0

PCIBAR2+74H Clear trigger stamp FIFO Write only

new

Bit	Description	Read	Write	Value after Reset
31:0	Clear trigger stamp FIFO. Sequential 0->1->0 1 Clear active. 0 FIFO free	No	Yes	0

PCIBAR2+74H Trigger stamp FIFO status Read only

new

Bit	Description	Read	Write	Value after Reset
8:0	Number of events in the FIFO	Yes	No	0
11:9	Reserved	Yes	No	0
12	0 : Not empty 1 : Empty	Yes	No	0
13	0 : Not full 1 : Full	Yes	No	0
31:14	Reserved	Yes	No	0

PCIBAR2+78H Trigger stamp FIFO content Read only

new

Bit	Description	Read	Write	Value after Reset
31:0	The counter value	Yes	No	0

PCIBAR2+7CH Firmware version Read only

Bit	Description	Read	Write	Value after Reset
3:0	Sub Sub version number	Yes	No	SubSub vers Numb
7.4	Sub version number	Yes	No	Sub vers. number
11:8	Version number	Yes	No	Version number
31:12	Reserved	Yes	No	0

7.3 SOFTWARE INSTALLATION

Driver and all related software are disponible under CVS.
Contact l.mirabito@ipnl.in2p3.fr

7.4 DRIVER

The TSC is delivered with a Linux driver.

Minimum version : These files must exist in ~Daq_Cms_Like/DeviceDriverTsc_Driver/

- tscdrv.c driver source code
- tcdrv.h header for driver code
- Makefile makefile for tscdrv.o target

In ~Daq_Cms_Like/DeviceDriverTsc_Driver/

Make the module **>make tscdrv.o**

Install tscdrv.o module (root privilege) **>insmod tscdrv.o**

Verify the module is now present in the module list **>lsmod**

Major number is 147

Steps in the driver :

- System initialization
  - Scan the PCI bus until the TSC Vendor and Device ID are found
  - Put addresses, interrupt line in tscpmc9080 structure
- Device openning
  - Remap PCIBAR0 and PCIBAR2 in Local0 and Local2 of tscpmc9080 structure
- Application calls
  - All useful calls are ioctl specifying the keyword

7.5 DEVICES

3 kinds of device must exist to access driver:

- **tscdrv**
  Minor 0 and 1. Full control of registers
- **tscusr**
  Minor 16 to 32. Used to control throttle and reset trigger filter. Only control of trigger register.
- **tsci2c**
  10 devices (0 to 9) to control I2C channels from J1 and J2

In ~Daq_Cms_Like/DeviceDriverTsc_Driver/, prepare devices (root privilege) **>make devices**

The following devices are now installed :

Device name	Minor number
/tscdrv00	0
/tscdrv01	1
/tscusr00	16
/tscusr01	17
/tscusr02	18
/tscusr03	19
/tscusr04	20
/tscusr05	21
/tsci2c00	32

Device name	Minor number
/tsci2c01	33
/tsci2c02	34
/tsci2c03	35
/tsci2c04	36
/tsci2c05	37
/tsci2c06	38
/tsci2c07	39
/tsci2c08	40
/tsci2c09	41

Verify these devices are present in /dev

7.6 USER LEVEL APPLICATIONS

- **C++ classes in the CMS DAQ software**

Refer to Daq distribution

LxTsc9080 : Access to the registers by read and write methods, to base 0, 2 and 3. Byte, short and long accesses are implemented.

TscDevice : A collection of methods similar to TscDescription.

- **Graphical application**

TscDialog : a graphical user interface, developped with qt, to control the main parts of the TSC.

In ~Daq_Cms_Like/Gui/ **>TscDialog**

- In the first thumb index, **open** the device.
- In the second, Set :
  - Minimum Number of Clocks to 3
  - Trigger count to 0 (free run)
  - Internal trigger frequency to 40
  - Trigger control to 8 (Internal source)

The Trigger accepted LED on the front panel should blink.

1. In TscDialog, Set APV latency
2. Run PllDialog and request hard reset , init PLL
3. The APV frame should now appear
4. In TscDialog, Set Trigger control to 40 and set Reset latency
5. The APV frame should now be stable
6. In TscDialog, Set Trigger control to 56 and set Calibration latency
7. Run Apv25Dialog and set APV latency according to the previous setting
8. The calibration pulses should appear on the APV frame

- **Software for gate use**

All software for gate use, particularily gate calibration can be found in :

~Daq_Cms_Like/DeviceDriverTsc_Driver/gate/

The last tar file can be found at the following address :
http://lyoinfo.in2p3.fr/cms/cmstraces/tscweb/software/gate.tar

GETTING STARTED

8.1 HARDWARE VERIFICATIONS

Fanning : This card has to be fanned. Several solutions exist, the best seems to be a fan fixed to the cover of the PC, airflow crossing the cover by a hole. The fan can be supplied by +5 or +12volts from the PC bus supply .

Jumpers S2, S3 and S4 are closed. Others are left open.

Power off, install the card in a PCI slot. Turn power on, the LED D9 should light immediately.

8.2 SOFTWARE TESTS

After power up, the TSC should be visible by the OS. 2 ways to verify the card was installed correctly : >lspci -vv or >cat /proc/bus/pci/devices. In both cases,the Vendor (10DC) and Device (CEF0) IDs should be mentionned.

Install the driver, see §7.4
Install devices, see §7.5
Run application program, see §7.6

9 CONTACTS & ORDERING INFORMATION

TSC ordering requests must be addressed to :

Didier Contardo email : contardo@in2p3.fr

All other questions about TSC can be addressed to :

Michel Ageron email : m.ageron@ ipnl.in2p3.fr

10 VERSION AND HISTORY

December 2001 :
Firmware version **1.4.1**. First delivery of the TSC. All functions described in this documentation are implemented except I2C functions.
February 2002 :
Firmware version **1.5.0**. Clock management improved. External I2C implemented except channel 8 and 9 (from J2).
June 2002 :
Firmware version **1.5.2.** Complete implementation of 10 I2C channels. Add trigger stamp counter and FIFO. Use NIM reserved pin as FIFO full. Output separated ECL APV clock and trigger.
July 2003 :
Firmware version **1.5.3.** Improvement of the external I2C controller robustness. Add an external I2C controller reset. Change external I2C controller speed to 100 khz.
Changes in hardware Reset Trigger Filter (Synchronism).
February 2004 :
Firmware version **1.5.4.** The previous version gate and external trigger part was not stable. The 1.5.4 version retreives good features. Change hardware Reset Trigger Filter to Inhibit trigger. Change internal trigger base period from 26 uS to 10 uS. Add a bit in trigger control which closes the gate.
July 2004 :
Firmware version **2.1.0.** This version proposes an APV emulator (25.10.04 not yet tested)

11 APPENDIX 1 : TPO USER MANUAL

11.1 INTRODUCTION :

Because of less connectivity of the PCI for factor, some signals such as Clock&Trigger and I2C do not appear on the back side of the TSC card. These signals are accessible through J1 and J2 connectors. The TPO (Tracker Parallel Output) is supposed to dispatch the signals from J1 and J2 connectors to 4 hybrid compatible lines, 4 general purpose I2C lines, 6 FED Clock and Trigger and other signals.

The TPO is a 114x114 mm standalone card that houses :
- 4 hybrid compatible lines, including clock&trigger on the same line, I2C for APVs, PLL, Mux and optohybrid.
- 4 general purpose I2C lines.
- 6 FED Clock and Trigger
- Other signals such as Inhibit trigger, Trigger out, NIM FIFO full, APV clock and APV trigger.

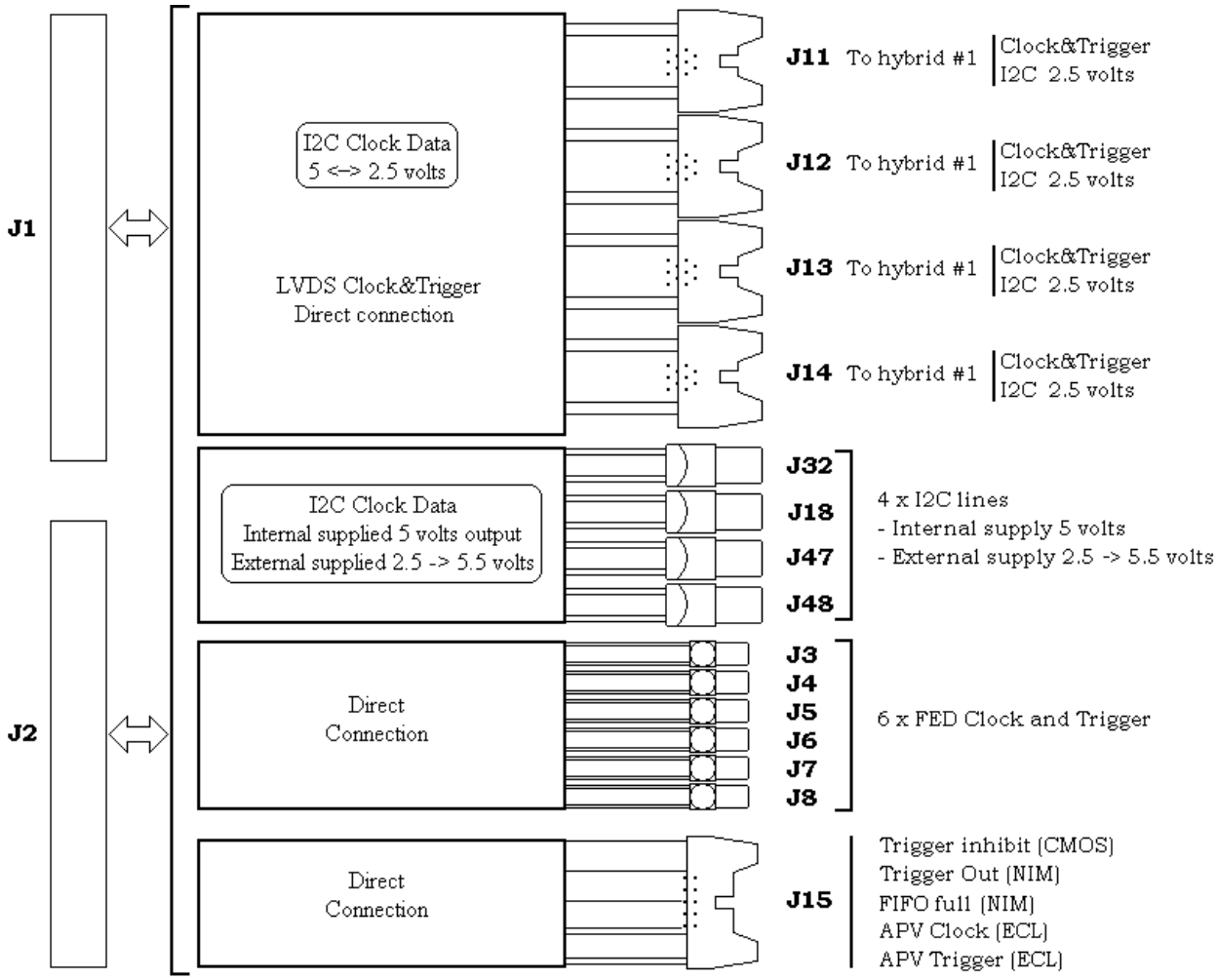

Figure 10 TPO Functionnal block diagram

11.2 HYBRID COMPATIBLE LINES (J11 TO J14)

Hybrids require an I2C, a Clock&trigger and a reset lines.
- I2C line : through a 3.3v to 2.5v adaptors. Refer to table 12 and figure 10 for plug number or device to open.
- Clock&trigger line : the LVDS signal is direcly connected to TSC.
- Reset line : a unique reset from the TSC is level adapted and distributed to hybrids.

TSC				TPO		Device to open /dev/…
I2C channel number	**Type**	**Connector on TSC**	**Pin number on TSC**	**I2C channel assignment**	**Plug number**	
0	SCL	J1	23	Hybrid #1	J14	/tsci2c00
	SDA	J1	22			
1	SCL	J1	20	Hybrid #2	J13	/tsci2c01
	SDA	J1	19			
2	SCL	J1	17	Hybrid #3	J12	/tsci2c02
	SDA	J1	16			
3	SCL	J1	14	Hybrid #4	J11	/tsci2c03
	SDA	J1	13			
4	SCL	J1	11	Not routed		/tsci2c04
	SDA	J1	10			
5	SCL	J1	8	Not routed		/tsci2c05
	SDA	J1	7			
6	SCL	J1	5	I2C line	J47	/tsci2c06
	SDA	J1	4			
7	SCL	J1	2	I2C line	J48	/tsci2c07
	SDA	J1	1			
8	SCL	J2	8	I2C line	J18	/tsci2c08
	SDA	J2	6			
9	SCL	J2	4	I2C line	J32	/tsci2c09
	SDA	J2	2			

Table 12 Summary of TPO connections

11.3 GENERAL PURPOSE I2C LINES (J32, J18, J47 AND J48)

- Jumper location

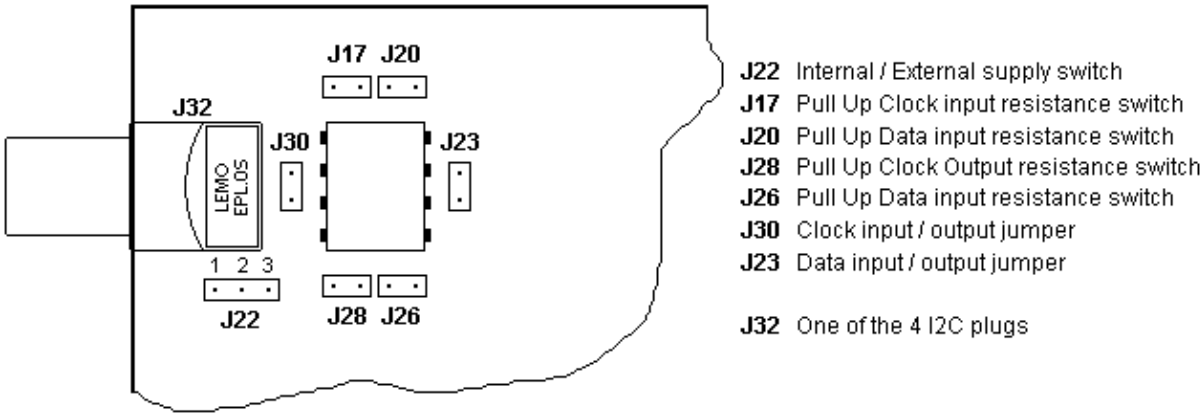

Figure 11 I2C output jumper location

- Internal / External supply

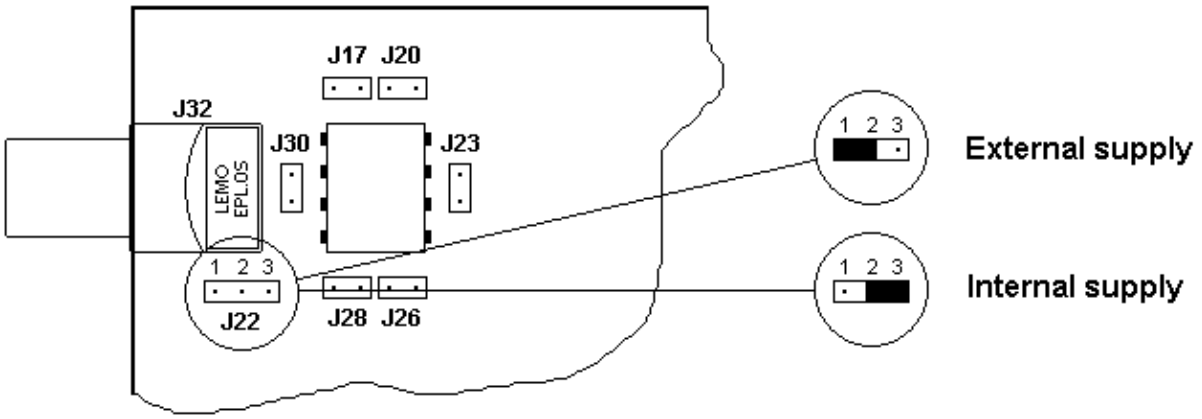

Figure 12 I2C output supply

Limits for external supply : 3.3 volts to 5.5 volts. Note that only the + line is floating, the ground is not.

- Using 82B715

The 82B715 is an I2C bus extender allowing long cables, refer to figure 13 for cell J32, the J18, J47 and J48 cells are identical. It is mounted on a support and therefore, can be unused. Refer to table 13 for different jumper combinations.

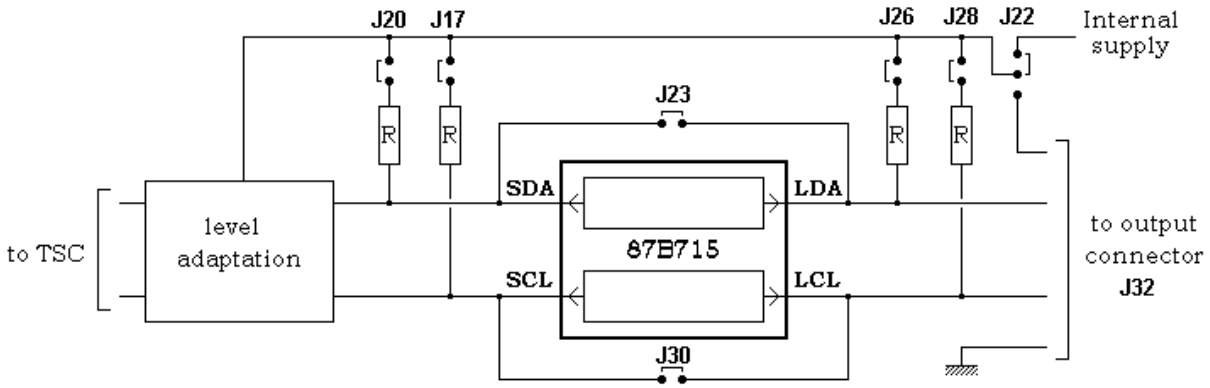

Figure 13 I2C Cell functionnal diagram for J32

	with 87B715		without 87B715	
	with pull up	without pull up	with pull up	without pull up
J20	X	X		
J17	X	X		
J23			X	X
J30			X	X
J26	X		X	
J28	X		X	

Table 13 Different jumper configurations for J32

- Default configuration

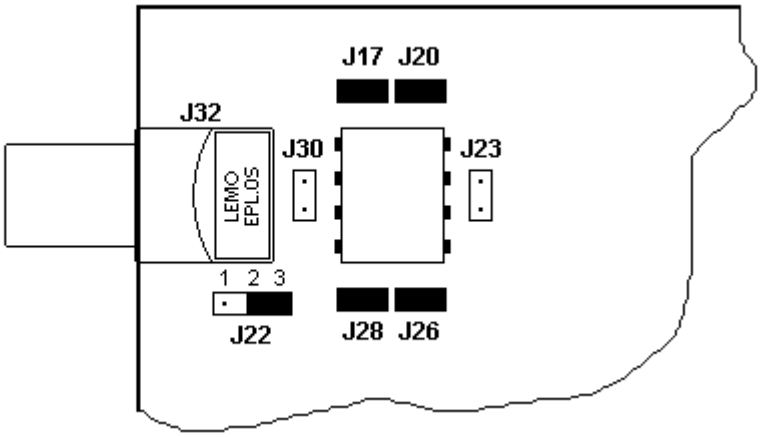

Figure 14 I2C Jumper default configuration

- Internal supply
- Use 87B715
- All pull up resistance swiched

11.4 FED CLOCK AND TRIGGER (J3 TO J8)

On the front pannel, there are 4 FED clock and trigger lines. 6 others on J2, they are routed on TPO. Add some more Lemo plugs to use these lines.

11.5 OTHER SIGNALS (J15)

Some other signals on J15

- Inhibit trigger a CMOS input (non 5 volts tolerent) pulled up to 3.3 volts, when shorted at 0 volts, any trigger output is inhibited
- FIFO full a NIM signal indicating event counter FIFO is full
- APV clock
- APV trigger the only outputs where APV clock and trigger are separated

11.6 SUPPLYING (J16)

This card must be supplied with 5 volts. J16 Pin 1 -> +5 volts
Pin 2 -> GND
An on board regulator supplies 2.5 volts for hybrid interface.

11.7 SCHEMATICS

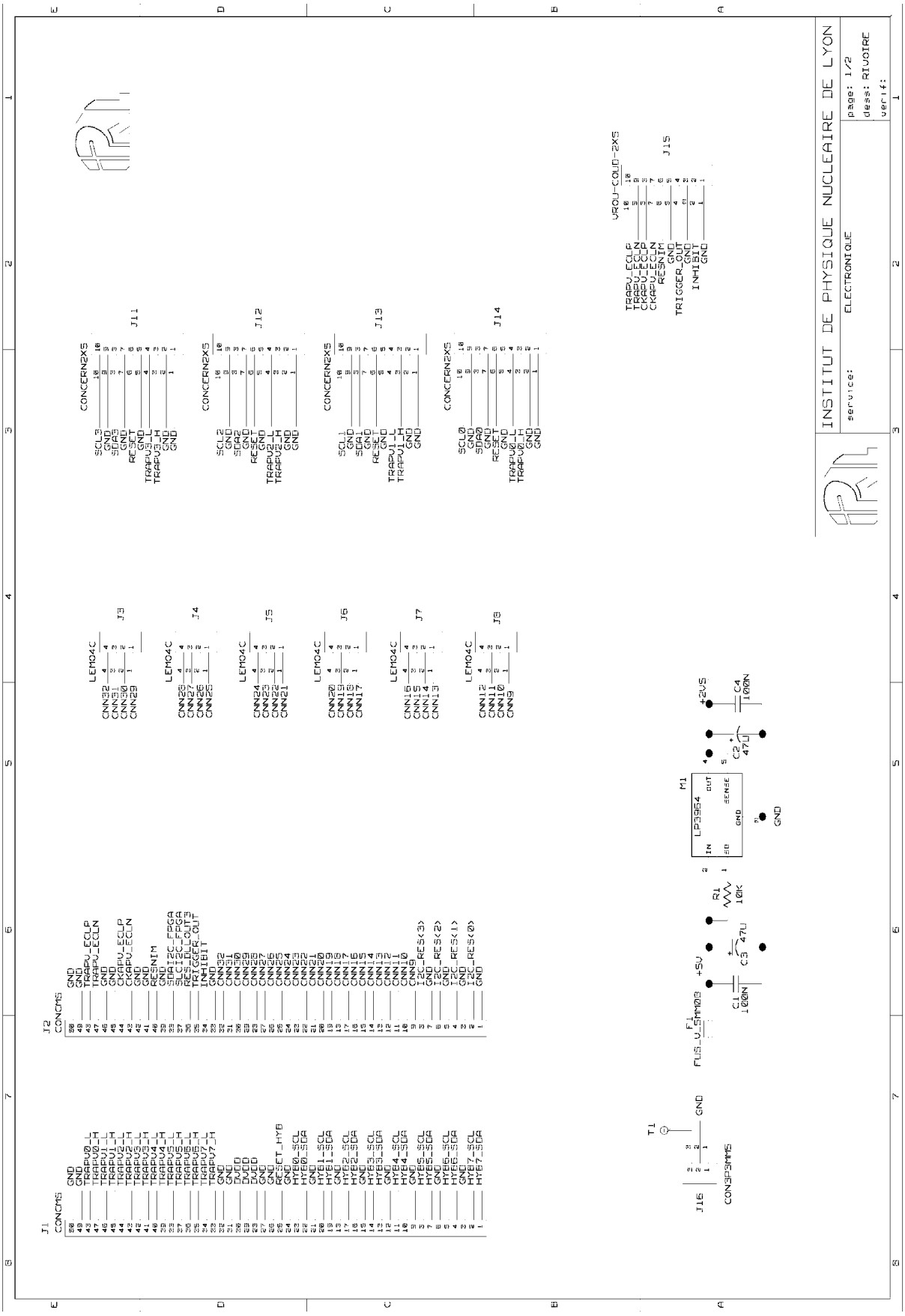

Figure 15. TPO Schematics no1

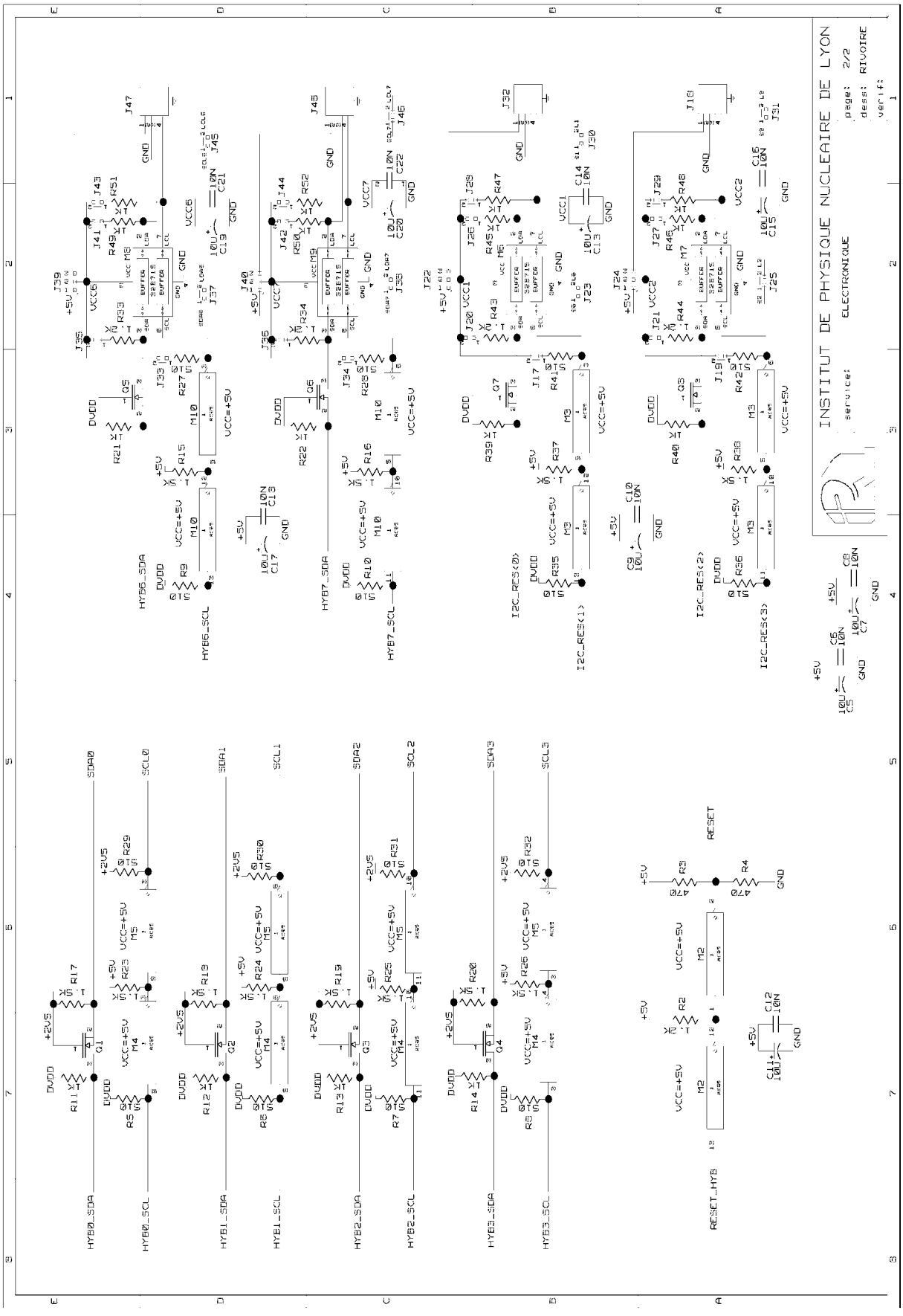

Figure 16. TPO Schematics no2

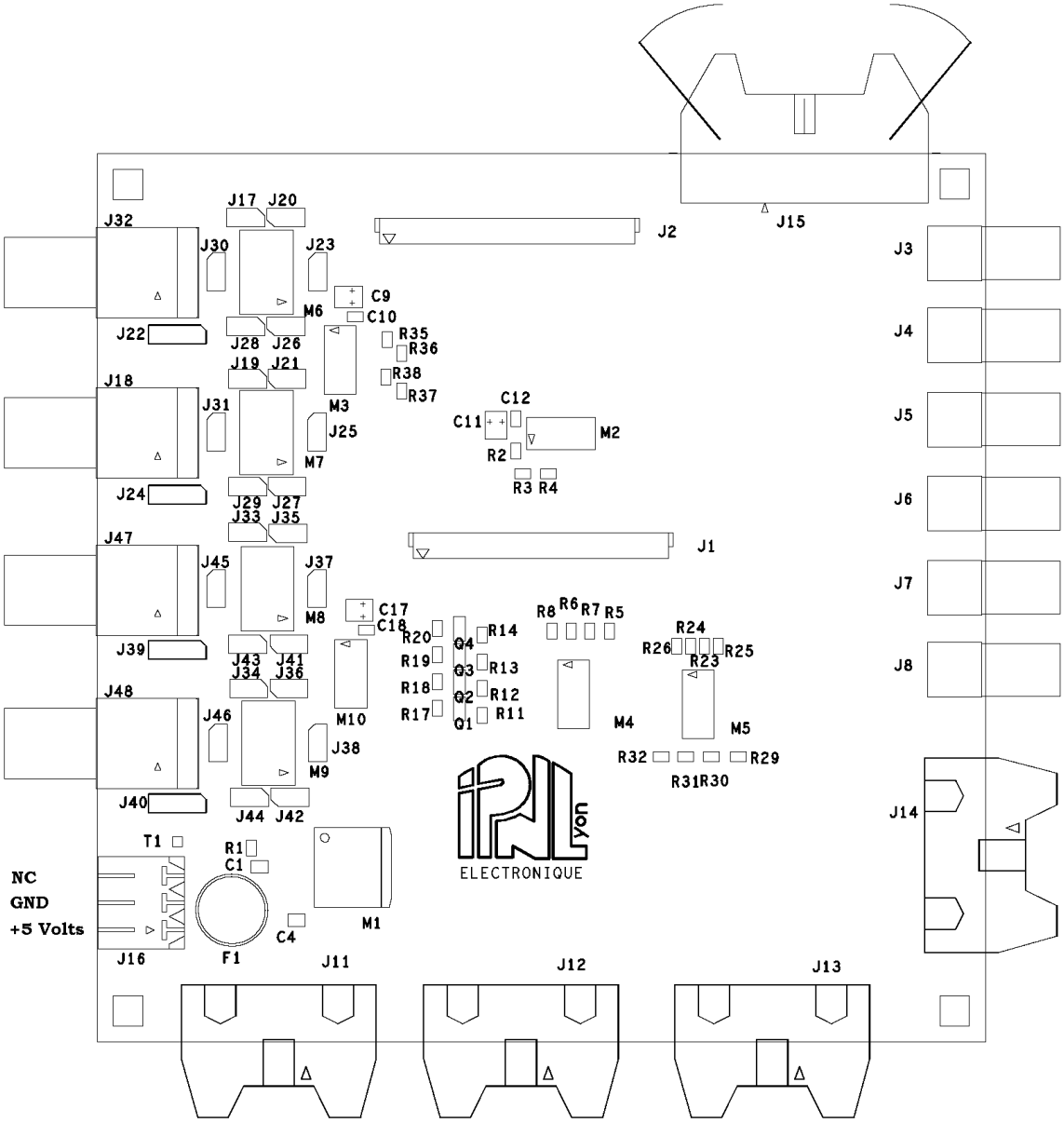

Figure 17. TPO Top view