The GANIL GTS Trigger Processor

1 Technical specifications

2 Hardware solution : VC707 + RASPBERRY

-VC707 : highlights

-RASPBERRY : highlights

3 Firmware developments

-Block diagram

-Implementation

4 Software developments

- Registers
- Python commands
- High Level commands

5 Early tests

- GTS tree – VC707 optical link
- GTS TP core test

6 Conclusion

The GANIL GTS Trigger Processor

1Technical specifications

GANIL GTS Trigger Processor specifications : STP-439-A

http://wiki.ganil.fr/gap/GAMMA/attachment/wiki/EXOGAM2/Projet/TP/STP-439A%20-%20SP%20Exogam2%20Trigger%20Processor.pdf

Two main constraints:

- **Coupling of the EXOGAM, AGATA detectors with ancillaries such as NEDA, DIAMANT, VAMOS.**
- **Full compatibility with the current GTS system which equips the AGATA detector:**
  - ✓ Optical fiber connection to GTS V3 Root
  - ✓ TR label is delivered on eight bits => TP cannot handle more than 256 TR labels
  - ✓ Aurora protocol message are transmitted at 2 Gb/s

Summary of Physics specifications (EXOGAM, AGATA, NEDA, TRACE):

- **Multiple simultaneous trigger capabilities** for a large set of detectors such as AGATA, EXOGAM, NEDA, DIAMANT, VAMOS ....
- **Multiplicities and prompt or-delayed coincidences** (example: γ-γ-γ OR γ -n)
- Complete freedom **to define the partitions** (channels are split up into partitions)
- No dead time; **continuous coincidence analysis**
- Possibility **to validate data from a detector which does not participate to the trigger decision**
- **Flexibility** (easiness to adjust new trigger conditions)
- Generate an **event pattern** to identify trigger type

The GANIL GTS Trigger Processor

1Technical specifications

Main steps of the trigger processing cycle:

1) SORTING

- To sort the incoming GTS leaves messages according to the assignment TR label - channel number
- To distribute and to store the TS into FIFO according to the channel number

2) MULTIPLICITY

- To issue the individual multiplicity window of each channel when (channel TS) = (reference TS)
- To perform the multiplicity of each partition
- To issue the multiplicity result of each partition according to the multiplicity threshold

3) ACCEPTANCE

- To issue the acceptance window of each partition
- To validate channels in coincidence with the acceptance window

4) COINCIDENCE

- To issue the coincidence window of each partition

5) ANALYSIS

- To combine the coincidence windows into the logical equation
- To source the event validation signal or event reject signal

6) EVENT PARAMETERS

- To build and to register the event parameters (TR pattern, the event number and the event time stamp) block.

7) REPLY

- To send back to each GTS leaves the validation or reject messages

The GANIL GTS Trigger Processor

2. TP hardware : VC707 key features

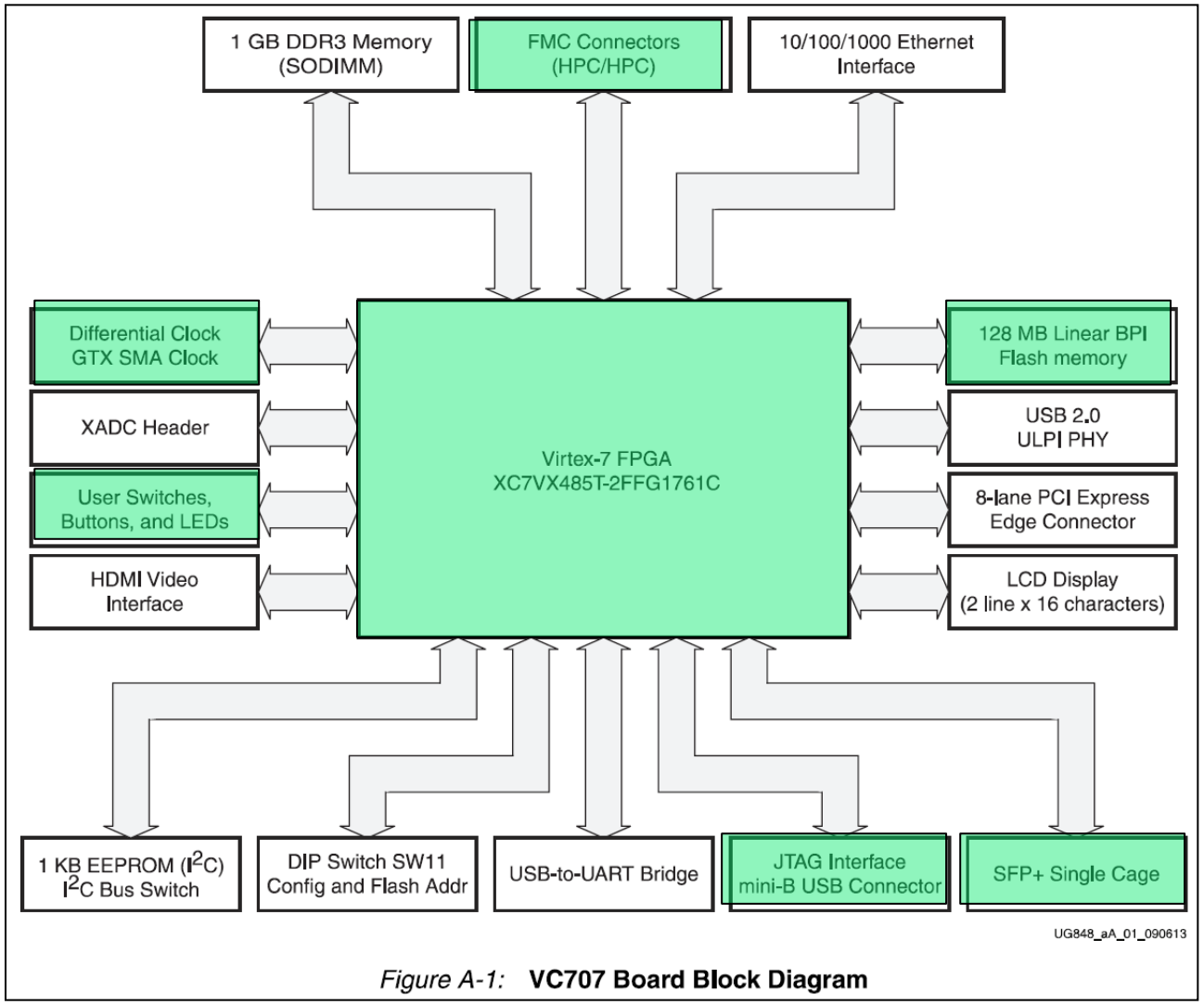
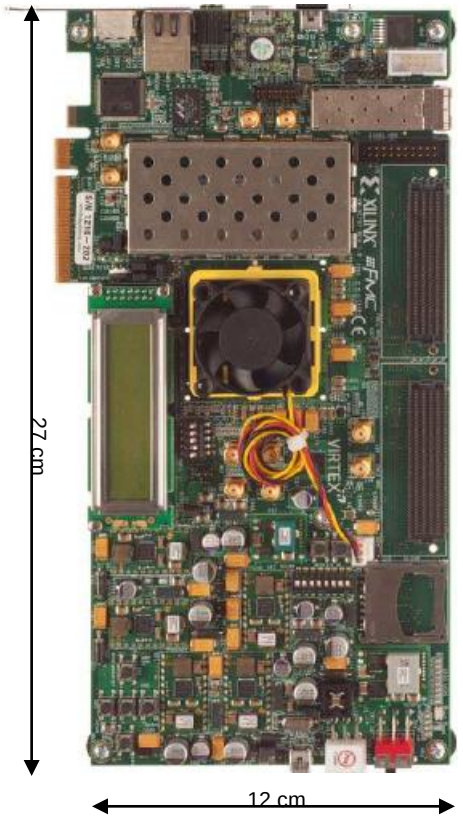


Figure A-1: **VC707 Board Block Diagram**

VC707
Xilinx
Evaluation Board



2. TP hardware : VC707 key features

Comparison of resources:

Virtex XC7VX485T-2FFG1761C - XC5VFX200T (current AGATA TP)
=> Twice more ressources in Vitex7 than in Virtex 5

Device	Logic Cells	Configurable Logic Blocks (CLBs)		RAM Blocks (kb)	GTX	I/O Banks	User I/O
		Slices	Distributed RAM (kb)				
XC7VX485T	485 760	75 900	8 175	37 090	56	14	700
XC5VFX200T	196 608	30 720	2 280	16 416	24	13	960

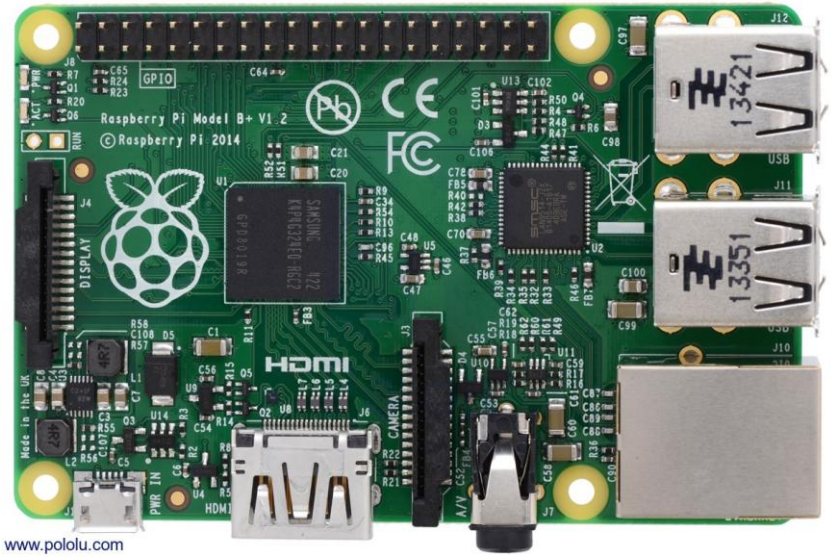
2. TP hardware : RASPBERRY key features

RASPBERRY Hardware features
- 700 MHz ARM11 processor
- 512 MB of RAM
- Ethernet port
- Four USB ports
- HDMI output
- Audio output and composite video output
- 40-pin GPIO header
- Camera interface (CSI)
- Display interface (DSI)
- Micro SD card slot

Software:
- Open source
- RASPBIAN: free Operating System based on Debian LINUX
- PYTHON : powerfull programming language

What you need to add:
- 5V power source
- Micro SD card
- Custom interface card for electric levels adaptation

Low cost:
- RASPBERRY + micro SD card = 50€
- Custom interface card : 50€



RASPBERRY PI Model B

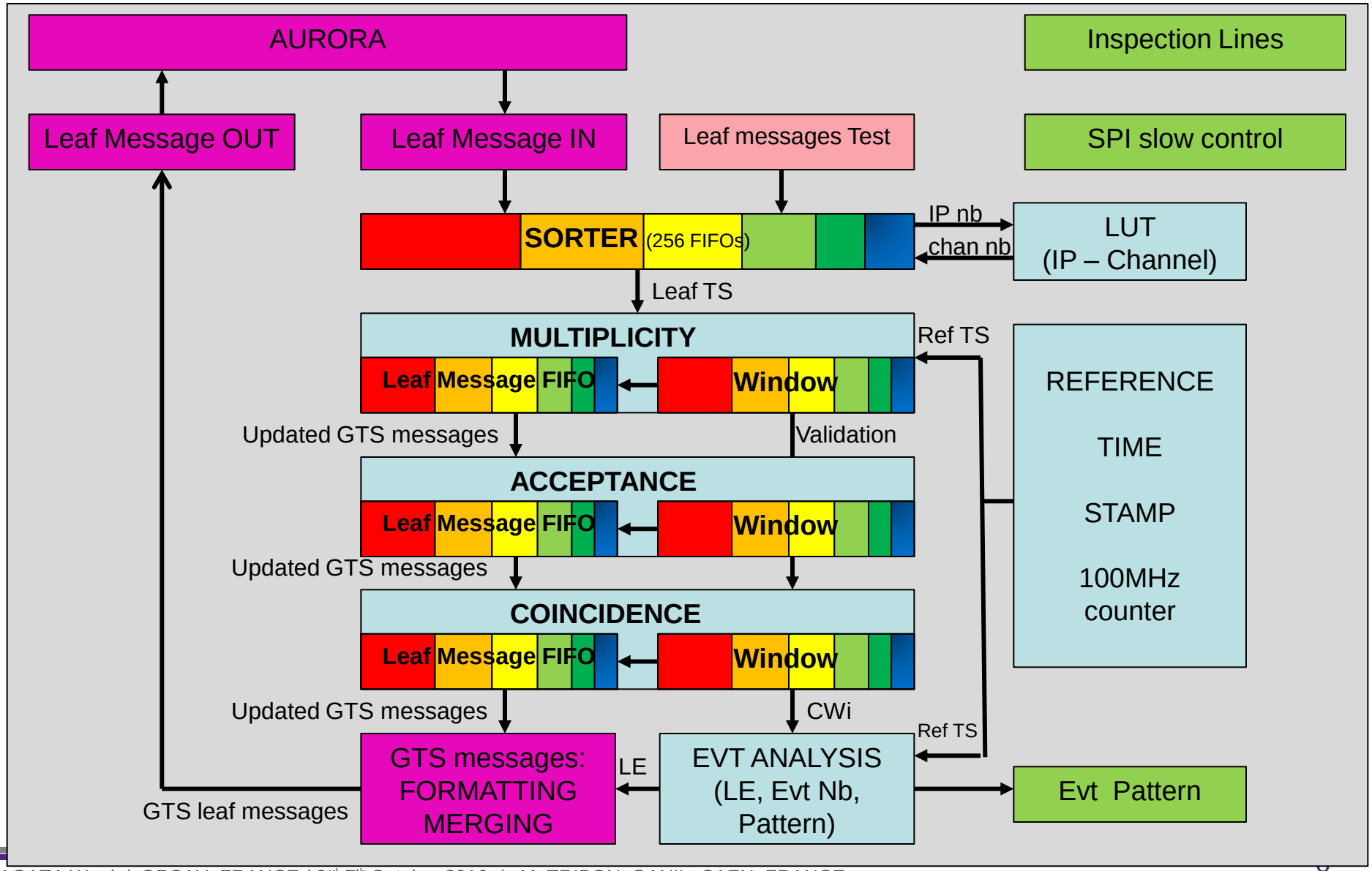
Custom interface

3. Firmware developments: Extra constraints and features

- 256 channels (Trigger Request labels on 8 bits)
- 13 Partitions:
  - P128: 128 channels
  - P64: 64 channels
  - P32: 32 channels
  - P16: 16 channels
  - P8: 8 channels
  - P1_1 to P1_8: one channel in each of 8 partitions
- Window Width range on 10 bits:
  - 13 Multiplicity Windows
  - 13 Acceptance Windows
  - 13 Coincidence Windows
- Window Delay range on 10 bits
  - 13 Coincidence Window Delays
- Validation/Rejection signal issued from one Logic Equation
  - 13 CW logically combined in one Logical Equation
  - LE = OR^{13} ($AND^{13}CW_n$)

The GANIL GTS Trigger Processor

3. Firmware developments: GTS TP simplified block diagram



The GANIL GTS Trigger Processor

3. Firmware developments: VHDL implementation in Virtex7

EXOGAM2_GTS_TP project tree

- Design Sources (6)
  - **exogam2_GTS_TP_top - Behavioral** (exogam2_GTS_TP_top.vhd) (17)
    - inst_pll_200MHz_100MHz - pll_200MHz_100MHz (pll_200MHz_100MHz.xci)
    - INST_IP_SETUP_SPI_TP - IP_SETUP_SPI_TP (IP_SETUP_SPI_TP.v)
    - INST_Partition_LE_setup - Partition_LE_setup - Behavioral (Partition_LE_setup.vhd)
    - inst_test_GTS_TP - test_GTS_TP - Behavioral (test_GTS_TP.vhd) (1)
    - inst_aurora_trigger_processor - aurora_trigger_processor - Behavioral (aurora_trigger_
    - inst_label_sorter - label_sorter - Behavioral (label_sorter.vhd)
    - inst_TS_48b_reference - TS_48b_reference - Behavioral (TS_48b_reference.vhd)
    - inst_multiplicity_window_generator - multiplicity_window_generator - Behavioral (multipli
    - inst_multiplicity_value - multiplicity_value - Behavioral (inst_multiplicity_value.vhd) (5)
    - inst_multiplicity_validation - multiplicity_validation - Behavioral (multiplicity_validation.vhd
    - inst_Acceptance_Coincidence_Windows - Acceptance_Coincidence_Windows - Behavior
    - inst_Part_TS_SubEventPattern - Part_TS_SubEventPattern - Behavioral (Part_TS_SubEv
    - inst_FIFO_evt - FIFO_evt - Behavioral (FIFO_evt.vhd) (25)
    - inst_LUT_CHANnb_IPnb - LUT_CHANnb_IPnb - Behavioral (LUT_CHANnb_IPnb.vhd) (2)
      - inst_rd_wr_LUT_CHANtoIP - RAM_8b_8b (RAM_8b_8b.xci)
      - inst_rd_wr_LUT_IPtoCHAN - RAM_8b_8b (RAM_8b_8b.xci)
    - inst_INSP_LOG - INSPECTION_LINES - Behavioral (INSPECTION_LINES.vhd)
    - inst_EvtParam_SPIrd - EvtParam_SPIrd - Behavioral (EvtParam_SPIrd.vhd)
    - inst_ILA_2 - ila_2 (ila_2.xci)
  - ila_0 (ila_0.xci)
  - ila_1 (ila_1.xci)
  - IP Update Log (3)
- Constraints (1)
  - constrs_1 (1)
    - TP_VC707.xdc
- Simulation Sources (4)
  - sim_1 (4)
    - exogam2_GTS_
    - ila_0 (ila_0.x
    - ila_1 (ila_1.x
    - Waveform Con

FIFO_evt IP tree

- inst_FIFO_evt - FIFO_evt - Behavioral (FIFO_evt.vhd) (25)
  - inst_logic_equation - Logic_Equation - Behavioral (Logic_Equation.vhd)
  - generate_FIFO_Channel_evt_P128.inst_FIFO_Channel_evt_P128 - FIFO_Channel_evt - Behavioral (FIFO_Channel_evt.vhd) (1)
  - generate_FIFO_Channel_evt_P64.inst_FIFO_Channel_evt_P64 - FIFO_Channel_evt - Behavioral (FIFO_Channel_evt.vhd) (1)
  - generate_FIFO_Channel_evt_P32.inst_FIFO_Channel_evt_P32 - FIFO_Channel_evt - Behavioral (FIFO_Channel_evt.vhd) (1)
  - generate_FIFO_Channel_evt_P16.inst_FIFO_Channel_evt_P16 - FIFO_Channel_evt - Behavioral (FIFO_Channel_evt.vhd) (1)
  - generate_FIFO_Channel_evt_P8.inst_FIFO_Channel_evt_P8 - FIFO_Channel_evt - Behavioral (FIFO_Channel_evt.vhd) (1)
  - inst_FIFO_Channel_evt_P1_1 - FIFO_Channel_evt - Behavioral (FIFO_Channel_evt.vhd) (1)
  - inst_FIFO_Channel_evt_P1_2 - FIFO_Channel_evt - Behavioral (FIFO_Channel_evt.vhd) (1)
  - inst_FIFO_Channel_evt_P1_3 - FIFO_Channel_evt - Behavioral (FIFO_Channel_evt.vhd) (1)
  - inst_FIFO_Channel_evt_P1_4 - FIFO_Channel_evt - Behavioral (FIFO_Channel_evt.vhd) (1)
  - inst_FIFO_Channel_evt_P1_5 - FIFO_Channel_evt - Behavioral (FIFO_Channel_evt.vhd) (1)
  - inst_FIFO_Channel_evt_P1_6 - FIFO_Channel_evt - Behavioral (FIFO_Channel_evt.vhd) (1)
  - inst_FIFO_Channel_evt_P1_7 - FIFO_Channel_evt - Behavioral (FIFO_Channel_evt.vhd) (1)
  - inst_FIFO_Channel_evt_P1_8 - FIFO_Channel_evt - Behavioral (FIFO_Channel_evt.vhd) (1)
  - ins_fifo_64x256_paramEVT - fifo_64b_256depth (fifo_64b_256depth.xci)
  - generate_Mux_FIFO64b_8to1_P128_0to7.inst_Mux_FIFO64b_8to1_P128_0to7 - Mux_FIFO64b_8to1 - Behavioral (Mux_FIFO64b_8to1.vhd) (1)
  - generate_Mux_FIFO64b_8to1_P64_0to7.inst_Mux_FIFO64b_8to1_P64_0to7 - Mux_FIFO64b_8to1 - Behavioral (Mux_FIFO64b_8to1.vhd) (1)
  - generate_Mux_FIFO64b_8to1_P32_0to7.inst_Mux_FIFO64b_8to1_P32_0to7 - Mux_FIFO64b_8to1 - Behavioral (Mux_FIFO64b_8to1.vhd) (1)
  - generate_Mux_FIFO64b_8to1_P16_0to7.inst_Mux_FIFO64b_8to1_P16_0to7 - Mux_FIFO64b_8to1 - Behavioral (Mux_FIFO64b_8to1.vhd) (1)
  - inst_Mux_FIFO64b_8to1_P8_0to7 - Mux_FIFO64b_8to1 - Behavioral (Mux_FIFO64b_8to1.vhd) (1)
  - inst_Mux_FIFO64b_8to1_P1_0to7 - Mux_FIFO64b_8to1 - Behavioral (Mux_FIFO64b_8to1.vhd) (1)
  - generate_Mux_FIFO64b_8to1_P128_GTS.inst_Mux_FIFO64b_8to1_P128_GTS - Mux_FIFO64b_8to1 - Behavioral (Mux_FIFO64b_8to1.vhd) (1)
  - inst_Mux_FIFO64b_8to1_P64_GTS - Mux_FIFO64b_8to1 - Behavioral (Mux_FIFO64b_8to1.vhd) (1)
  - inst_Mux_FIFO64b_8to1_P128to32_GTS - Mux_FIFO64b_8to1 - Behavioral (Mux_FIFO64b_8to1.vhd) (1)
  - inst_Mux_FIFO64b_8to1_Pall_GTS - Mux_FIFO64b_8to1 - Behavioral (Mux_FIFO64b_8to1.vhd) (1)

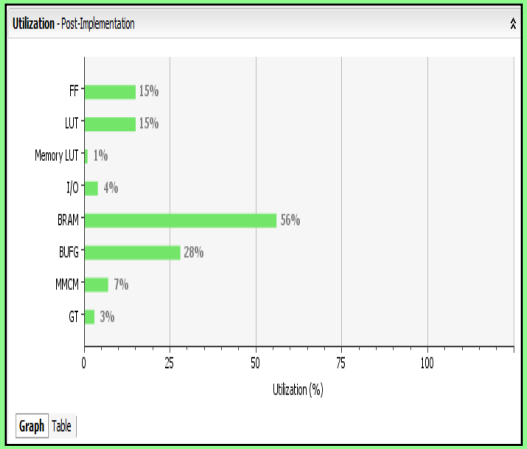
Project Summary

Project Settings

Project name:	EXOGAM2_GTS_TP_MT
Project location:	E:/Xilinx_projects/EXOGAM2_GTS_TP
Product family:	Virtex-7
Project part:	Virtex-7 VC707 Evaluation Platform (xc7vx485tffg1761-2)
Top module name:	exogam2_GTS_TP_top

Board Part

Display name:	Virtex-7 VC707 Evaluation Platform
Board part name:	xilinx.com:vc707:part0:1.1
Repository path:	C:/Xilinx/Vivado/2015.2/data/boards/board_parts
URL:	www.xilinx.com/vc707
Board overview:	Virtex-7 VC707 Evaluation Platform



Power

Total On-Chip Power:	**1.321 W**
Junction Temperature:	**26,5 °C**
Thermal Margin:	58,5 °C (49,3 W)
Effective ϑJA:	1,1 °C/W
Power supplied to off-chip devices:	0.006 W
Confidence level:	Low

Implementation

Conflict nets:	0
Unrouted nets:	0
Partially routed nets:	0
Fully routed nets:	143726

4. Software developments: SPI registers

14 SPI control and status registers

REG_0_CTRL : General Control (reset, TP_on, start_TS_counter, enable self start_TS_counter enable_test)
REG_1_CTRL : TS offset of the root counter
REG_2_CTRL : It sets the label for the readout of the STATUS registers
REG_3_CTRL : LUT : CHANnb to IPnb. (256 sub-registers)
REG_3_STATUS : read LUT (set by REG_2_CTRL)
REG_4_CTRL : start delay of the TS root counter
REG_5_CTRL: Multiplicity Window Width (13 sub-registers)
REG_5_STATUS: Multiplicity Window Width (set by REG_2_CTRL)
REG_6_CTRL: Multiplicity Threshold (13 sub-registers)
REG_6_STATUS: Multiplicity Threshold (set by REG_2_CTRL)
REG_7_CTRL: Acceptance Window Width (13 sub-registers)
REG_7_STATUS: Acceptance Window Width (set by REG_2_CTRL)
REG_8_CTRL:Coincidence Window Width (13 sub-registers)
REG_8_STATUS: Coincidence Window Width (set by REG_2_CTRL)
REG_9_CTRL:Coincidence Window Delay (13 sub-registers)
REG_9_STATUS: Coincidence Window Delay (set by REG_2_CTRL)
REG_10_CTRL: Logic Equation (13 sub-registers)
REG_10_STATUS: Logic Equation (set by REG_2_CTRL)
REG_11_CTRL : Enable multiplicities windows of the 256 channels and multiplicity validations (256 sub-registers)
REG_11_STATUS: Channel number or partition number (set by REG_2_CTRL)
REG_12_CTRL: Logic Inspection (4 sub-registers for connecting ≈300 signals to 4 inspection lines)
REG_12_STATUS: Logic Inspection (set by REG_2_CTRL)
REG_15_CTRL: Event pattern

The GANIL GTS Trigger Processor

4. Software developments: Python commands

```
tripon@ganp292 MINGW64 ~
$ ssh -x pi@193.48.111.229
pi@193.48.111.229's password:
Linux raspberrypi 4.1.13+ #826 PREEMPT Fri Nov 13 20:13:22 GMT 2015 armv6l
The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the individual files in /usr/share/doc/*/copyright.
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent permitted by applicable law.
Last login: Thu Sep 22 08:47:35 2016 from ganp292.in2p3.fr
pi@raspberrypi ~ $ ls
Desktop  indiecity  ocr_pi.png  python_games  python-spi  SPI_prog  test_rgb_mezz.sh  vid.h264  WebIOPi-0.6.0  WebIOPi-0.6.0.tar.gz
WiringPi_prog
pi@raspberrypi ~ $ cd  python-spi
pi@raspberrypi ~/python-spi $ ls
git       rd_spi.py        Register_GTS_TP.pyc  Register.pyc    setup_GTS_TP.txt  spidev_module.c  test_spi.py       test_units.py
TP_EvtParam_rd_spi.py     Units.py  wr_spi.py    _info.txt  Register_GTS_TP.py    Register.py       setup_GTS_TP.py   setup.py
test_rd_spi.py  test_units.help.txt  test_wr_spi.py  TP_test_wr_spi.py     Units.pyc
pi@raspberrypi ~/python-spi $
```

Two basic R/W python command:

test_wr_spi.py

test_rd_spi.py

Python script files:

TP_test_wr_spi.py : TP setup for test

TP_EvtParam_rd_spi.py : read pattern

The GANIL GTS Trigger Processor

4. Software developments: HL commands

REG_1_CTRL: TS offset of the root counter (0ns-42949672950ns):
offset(1)= 0ns

REG_3_CTRL: LUT channel index vs IP address (0-255 or P*.*-.*):
LUT(11)= P128.1-.0, P128.2-.1, P128.18-.18, P128.35-.35, P128.52-.52, P128.69-.69, P128.86-.86, P128.103-.103, P128.120-.120, P1_1-.248, P1_2-.249

REG_4_CTRL: start delay of the TS root counter (0ns-40950ns):
delay(1)= 10230ns

REG_5_CTRL: Multiplicity Window Width (10ns-10us):
MWW(3)= P128-.140ns, P1_1-.210ns, P1_2-.220ns

REG_6_CTRL: Multiplicity Threshold (1-256):
MT(3)= P128-.5, P1_1-.1, P1_2-.1

REG_7_CTRL: Acceptance Window Width (10ns-10us):
AWW(3)= P128-.320ns, P1_1-.10ns, P1_2-.10ns

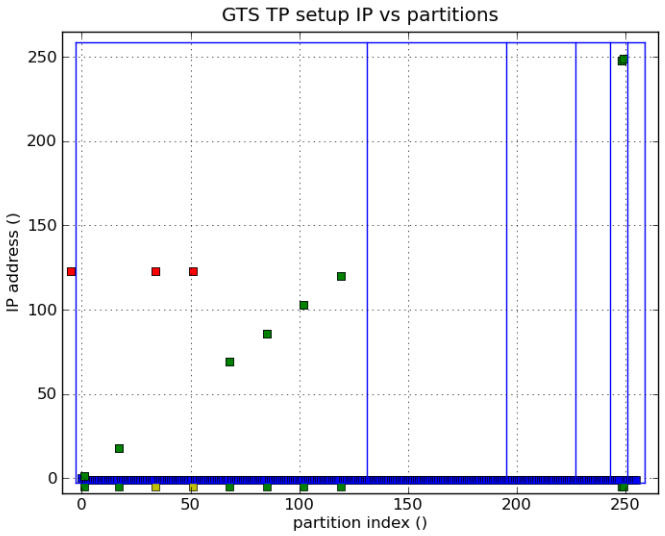
REG_8_CTRL: Coincidence Window Width (10ns-10us):
CWW(3)= P128-.480ns, P1_1-.530ns, P1_2-.540ns

REG_9_CTRL: Coincidence Window Delay (10ns-10us):
CWD(3)= P128-.1120ns, P1_1-.1170ns, P1_2-.1180ns

REG_10_CTRL: Logic Equation LE index vs Operand Number (13):
LE(2)= P128-.0 P1_1-.1, P1_2-.1

REG_11_CTRL: Enable Multiplicity Window (P*.*):
EMW(11)= P128.1, P128.2, P128.18, P128.35, P128.52, P128.69, P128.86, P128.103, P128.120, P1_1, P1_2
REG_11_CTRL: Enable Partitions Multiplicity Validation: (P*) :
EPMV (3)= PMV128, PMV1_1, PMV1_2

REG_12_CTRL: Logic Inspection (0- 307):
LI(4)= 1, 12, 123, 300




```
File  Edit  View  Terminal  Go  Help
EMW #11 (256)
 Enable Multiplicity Window (P*.*) ([channel])
[0..1] or 0
[1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0] dimensionless
CWW #8 (13)
 Coincidence Window Width  (10ns-10us) ([time])
[1..1000] or -1
[48 20 30 40 50 53 54 10 10 10 10 10 10] tic
CWD #9 (13)
 Coincidence Window Delay  (10ns-60us) ([time])
[1..6000] or -1
[112   2   3   4   5 117 118 100 100 100 100 100 100] tic
AWW #7 (13)
 Acceptance  Window Width  (10ns-10us) ([time])
[1..1000] or -1
[32 20 30 40 50  1  1  1  1  1  1  1  1] tic
 -LUT as plot:
duplicate IP addresses 2 [123 123]
           @ indexes 2 [34 51]
```

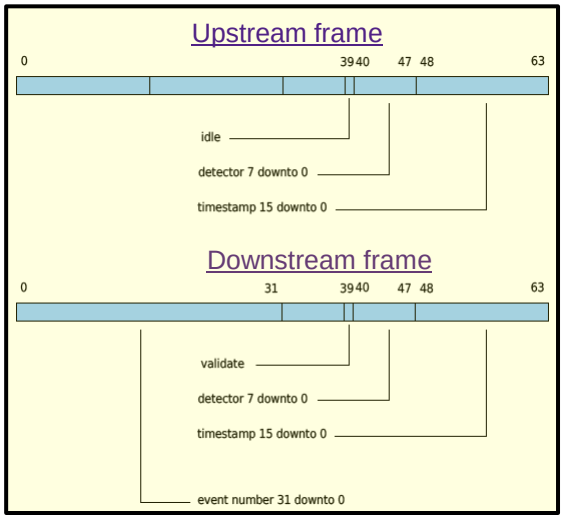
5. Early tests: GTS tree – VC707 optical link

Test aiming at the validation of the optical communication between VC707 and GTS tree.

GTS protocol:
- Aurora 8b/10b encoding
- 64 bits frame

Hardware:
- GTS tree: 1 ROOT V3, 3 FIFO V3, 8 NUMEXO2
- VC707

VC707 firmware:
- AURORA IP: Xilinx GTX IP, 8b/10b encoding, framing.
-TX_CRC, RX_CRC IPs
- Very light TP core: incoming GTS leaf message always accepted



Acquisition run:
- GTS built and aligned
- TP included in the GTS tree
- Acquisition is started => GTS leaf messages are sent to GTS upstream link
- NUMEXO2 inspection lines: Trigger validation is displayed on scope, as expected

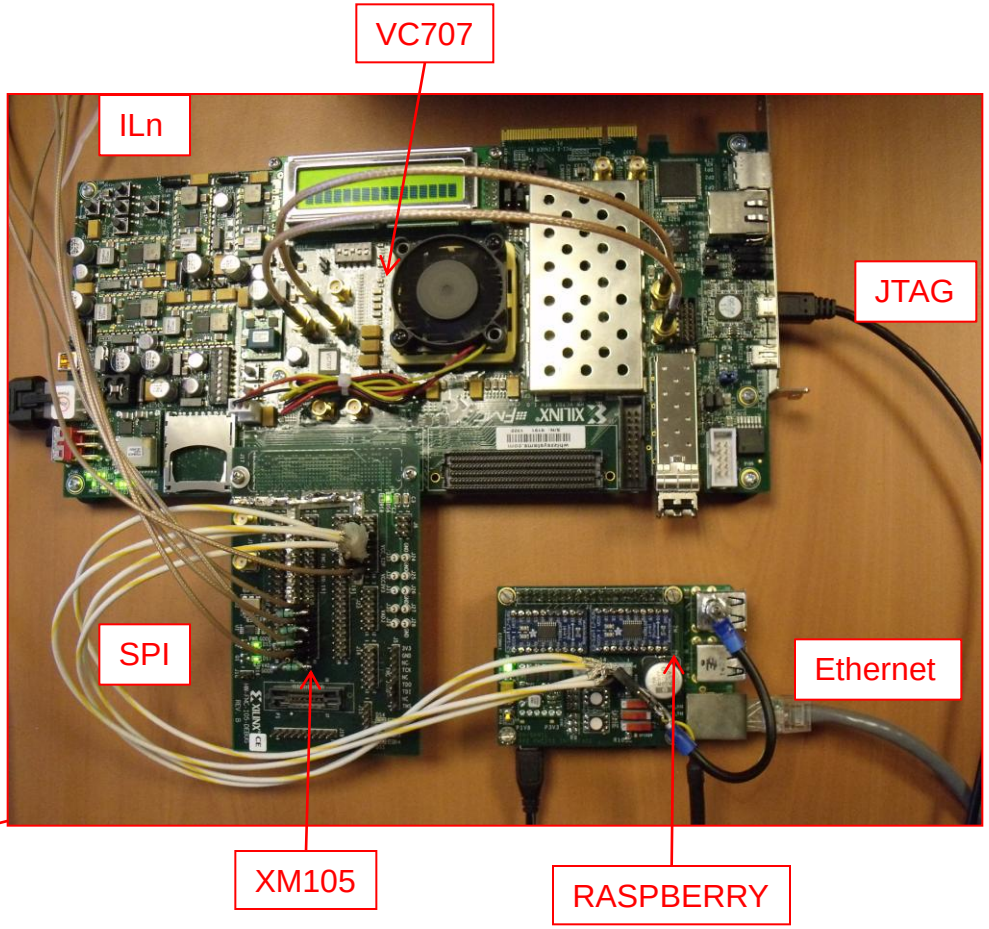
Conclusion of the test: GTS tree – VC707 link has been successfully tested

AURORA+CRC firmware in VC707 is validated

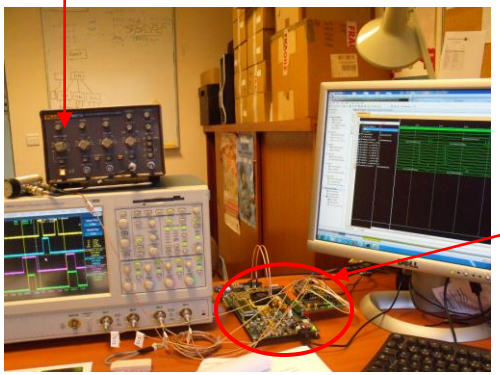
The GANIL GTS Trigger Processor

5. Early tests: GTS TP core test

Test Bench



Pulser aiming
to trig the block
of 10 test GTS messages



The GANIL GTS Trigger Processor

5. Early tests: GTS TP core test

Test aiming at the validation of the TP core in VC707:

- No connection to the GTS tree
- GTS leaf messages are sourced by an inner IP : test_GTS_TP
- GTS TP setup:
  - 3 partitions are enabled : P128, P1_1, P1_2
  - 10 channels are enabled, with IP number = channel number :
    - 8 in P128: 1, 12, 23, 34, 45, 56, 67, 78
    - 1 in P1_1: F8
    - 1 in P1_2: F9
  - Windows Widths

    MWWP128 = 160ns, MWWP1_1 = 210ns MWWP1_2 = 220ns

    AWWP128 = 320ns

    CWWP128 = 480ns, CWWP1_1 = 530ns, CWWP1_2 = 540ns
  - Multiplicity Thresholds:
    - MTHP128 = 5
  - CW Delays

    CWDP128 = 1120ns, CWDP1_1 = 1170ns, CWDP1_2 = 1180ns
  - LE = CWP128 OR (CWP1_1 AND CWP1_2)

```
                                                    ID   TS
test_message_GTS_i <= x"0000000000 01 00F1";
test_message_GTS_i <= x"0000000000 12 00F2";
test_message_GTS_i <= x"0000000000 23 00F3";
test_message_GTS_i <= x"0000000000 34 00F4";
test_message_GTS_i <= x"0000000000 45 00F5";
 test_message_GTS_i <=  x"0000000000 56 0F6";
test_message_GTS_i <= x"0000000000 67 00F7";
test_message_GTS_i <= x"0000000000 78 0148";
test_message_GTS_i <= x"0000000000 F8 0001";
test_message_GTS_i <= x"0000000000 F9 0010";
```

```
pi@raspberrypi ~/python-spi $ ./TP_test_wr_spi.py                          #GTS TP setup
pi@raspberrypi ~/python-spi $ ./test_wr_spi.py 0X00 0X0C 0X01 0X31 0X00 0X02   #OR MW on IL2
pi@raspberrypi ~/python-spi $ ./test_wr_spi.py 0X00 0X0C 0X01 0X33 0X00 0X04   #OR CW on IL3
pi@raspberrypi ~/python-spi $ ./test_wr_spi.py 0X00 0X0C 0X01 0X30 0X00 0X08   #LE on IL4
pi@raspberrypi ~/python-spi $ ./test_wr_spi.py 0X00 0X00 0X00 0X00 0X00 0X1A   # enable test and TP on
pi@raspberrypi ~/python-spi $ ./TP_EvtParam_rd_spi.py                      # SPI  evt pattern read
```

The GANIL GTS Trigger Processor

5. Early tests: GTS TP core test

GTS messages returned to the leaves:

	TS	ID	accept	EvtNb
message_GTS_s (63..0):	0001	F8	8000	000009
message_GTS_s (63..0):	0010	F9	8000	000009
message_GTS_s (63..0):	0148	78	0000	000000
message_GTS_s (63..0):	00F3	23	8000	00000A
message_GTS_s (63..0):	00F5	45	8000	00000A
message_GTS_s (63..0):	00F4	34	8000	00000A
message_GTS_s (63..0):	00F6	56	8000	00000A
message_GTS_s (63..0):	00F1	01	8000	00000A
message_GTS_s (63..0):	00F7	67	8000	00000A
message_GTS_s (63..0):	00F2	12	8000	00000A

Conclusion:

1- GTS messages issued from the TP core matched with the incoming test GTS messages

2- Event Pattern is in accordance with the incoming test GTS messages

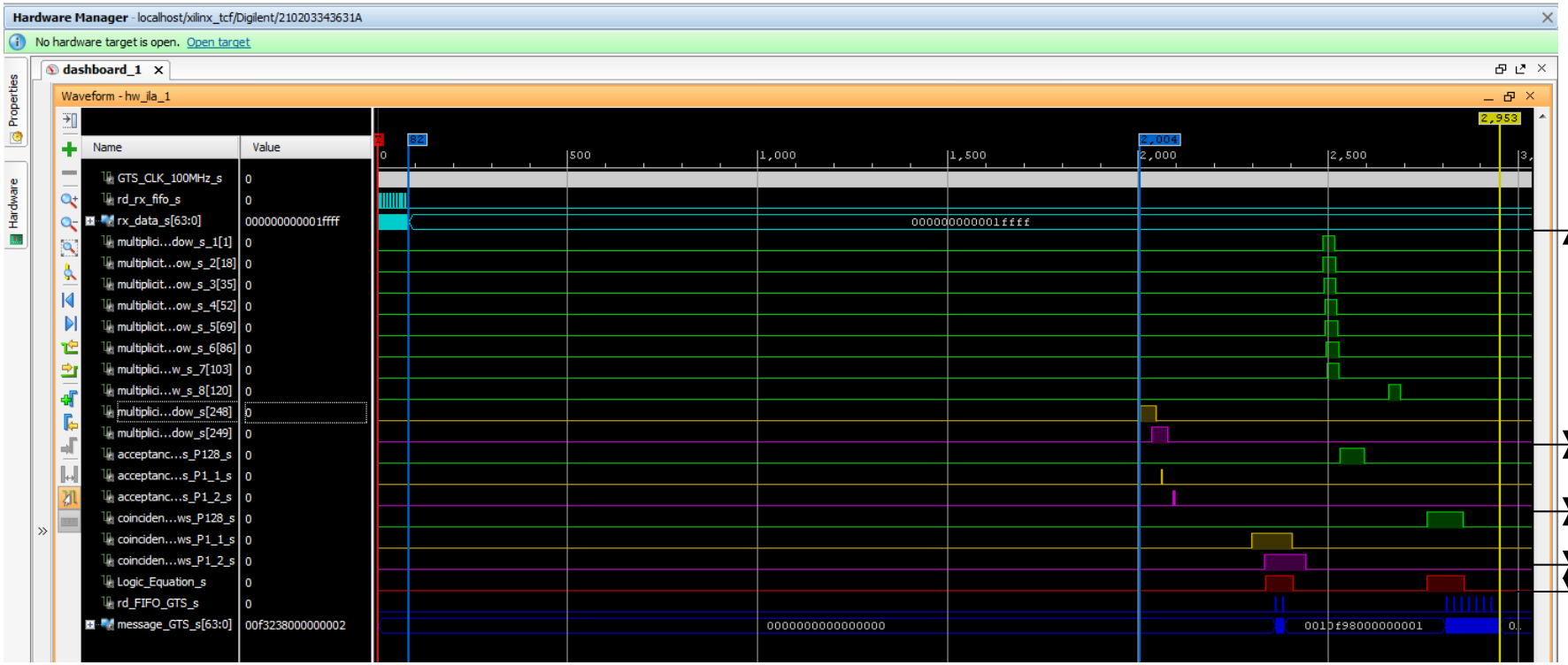
Event pattern

```
pi@raspberrypi ~/python-spi $ ./TP_EvtParam_rd_spi.py
('0x0', '0x0', '0xca', '0xfe', '0xab', '0xcd')
('0x0', '0x0', '0xef', '0x0', '0x0', '0x9')          #event number 9
('0x0', '0x0', '0x0', '0x0', '0x0', '0x0')
('0x0', '0x0', '0x0', '0x0', '0x0', '0x0')
('0x0', '0x0', '0x0', '0x0', '0x0', '0x0')
('0x0', '0x0', '0x0', '0x0', '0x0', '0x0')
('0x0', '0x0', '0x0', '0x0', '0x0', '0x0')
('0x0', '0x0', '0x0', '0x0', '0x0', '0x0')
('0x0', '0x0', '0x3', '0x0', '0x0', '0x0')           #channels F8 and F9
('0x0', '0x0', '0x0', '0x0', '0x0', '0x0')
('0x0', '0x0', '0x0', '0x0', '0x0', '0x70')          #  evt TS
('0x0', '0x0', '0x8c', '0xf4', '0x0', '0xa8')
pi@raspberrypi ~/python-spi $ ./TP_EvtParam_rd_spi.py
('0x0', '0x0', '0xca', '0xfe', '0xab', '0xcd')
('0x0', '0x0', '0xef', '0x0', '0x0', '0xa')          #event number a
('0x0', '0x0', '0x0', '0x4', '0x0', '0x2')           #channels 1 and 12
('0x0', '0x0', '0x0', '0x10', '0x0', '0x8')          #channels 23 and 34
('0x0', '0x0', '0x0', '0x40', '0x0', '0x20')         # channels 45 and 56
('0x0', '0x0', '0x0', '0x0', '0x0', '0x80')          # channels 67
('0x0', '0x0', '0x0', '0x0', '0x0', '0x0')
('0x0', '0x0', '0x0', '0x0', '0x0', '0x0')
('0x0', '0x0', '0x0', '0x0', '0x0', '0x0')
('0x0', '0x0', '0x0', '0x0', '0x0', '0x0')
('0x0', '0x0', '0x0', '0x0', '0x0', '0x70')          #  evt TS
('0x0', '0x0', '0x8c', '0xf4', '0x1', '0x7d')
```

The GANIL GTS Trigger Processor

5. Early tests: GTS TP core test

GTS TP cycle: Signals displayed on Vivado ILA board



- 3 partitions : P128, P1_1, P1_2
- 10 channels :
  - 8 in P128: 1, 12, 23, 34, 45, 56, 67, 78
  - 1 in P1_1: F8
  - 1 in P1_2: F9

LE = CWP128 OR (CWP1_1 AND CWP1_2)

The GANIL GTS Trigger Processor

5. Early tests: GTS TP core test

GTS TP cycle:

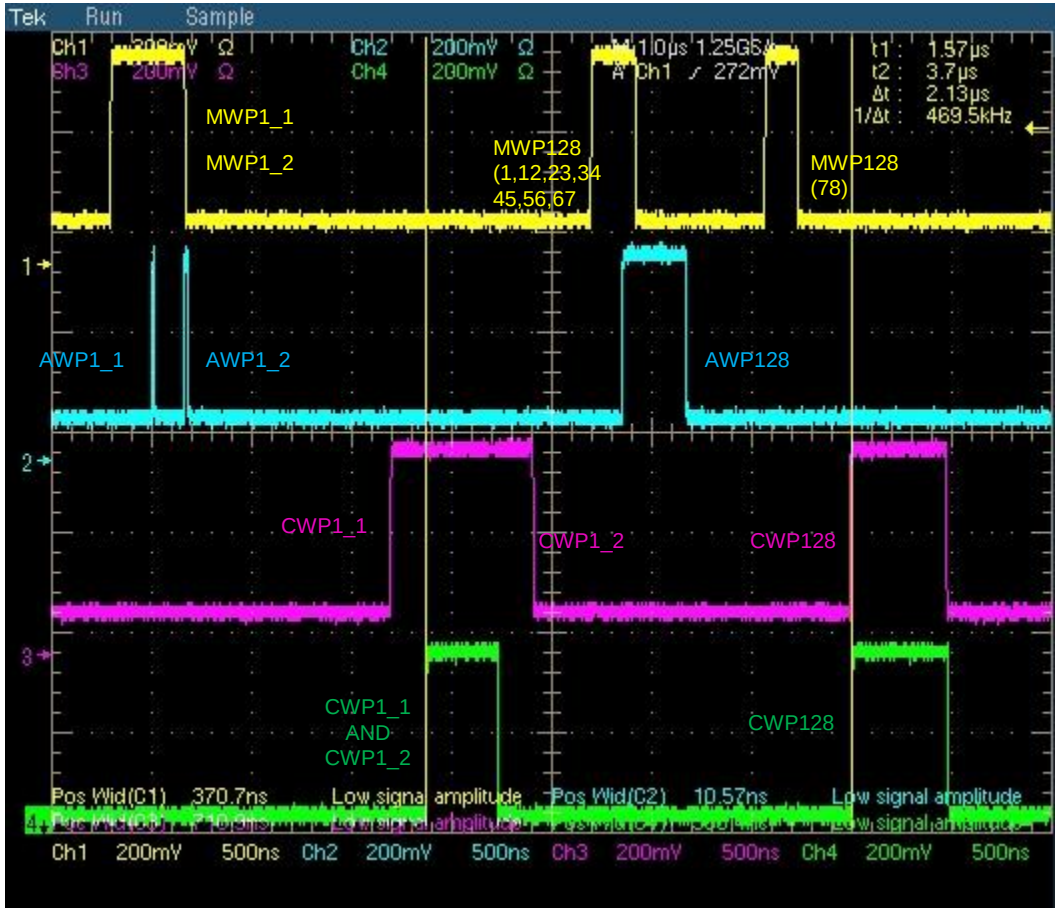
Signals connected to 4 Inspection Lines and displayed on oscilloscope

OR_MW

OR_AW

OR_CW

LE



5. Conclusion:

Early tests:

-GTS V3 tree <=> VC707 communication : OK

- GTS TP core: OK

To do:

- To connect the GTS TP to the GTS tree
- To house the GTS TP in a box (19” case, rack mount, 1 U)
- To debug the GTS TP core
- To implement the GTS TP in detector instrumentation
- To develop the IP protocol (pattern readout)

2. TP hardware

TP hardware components housed in a custom box (19” case, rack mount, 1 U)