RF2TTC
TEST PROCEDURE

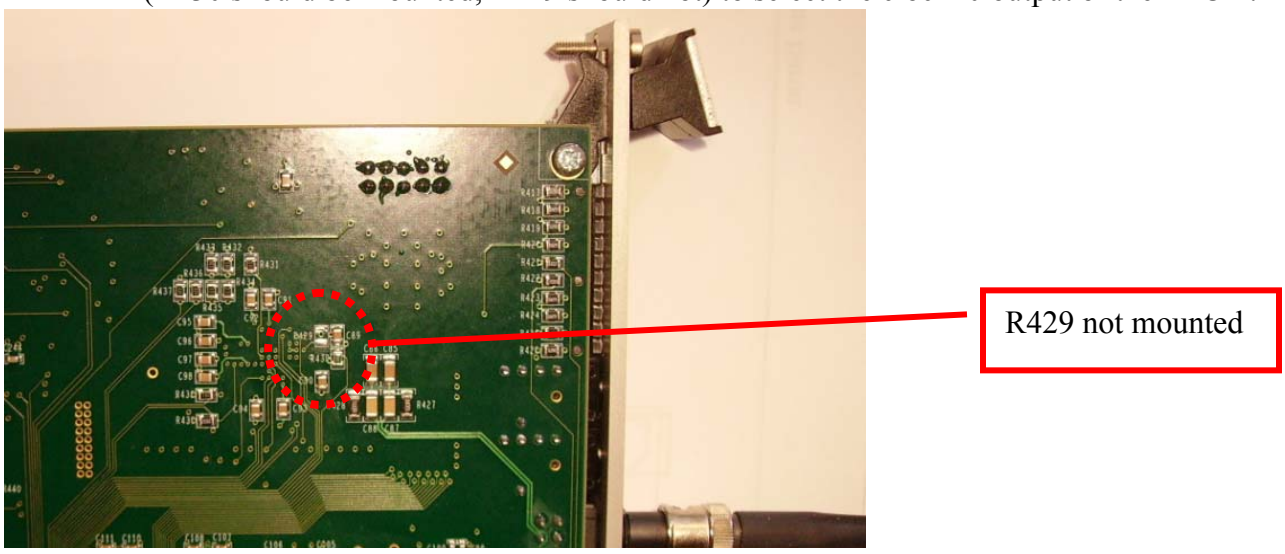
Table of Contents

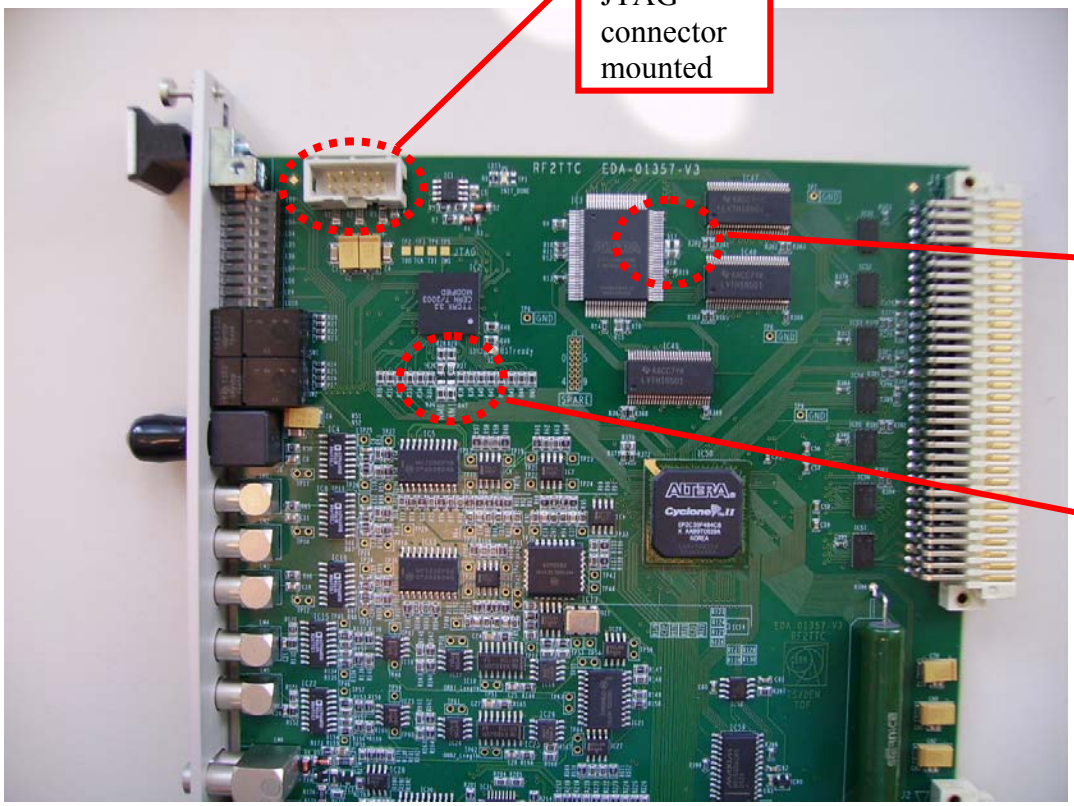
1. Visual inspection

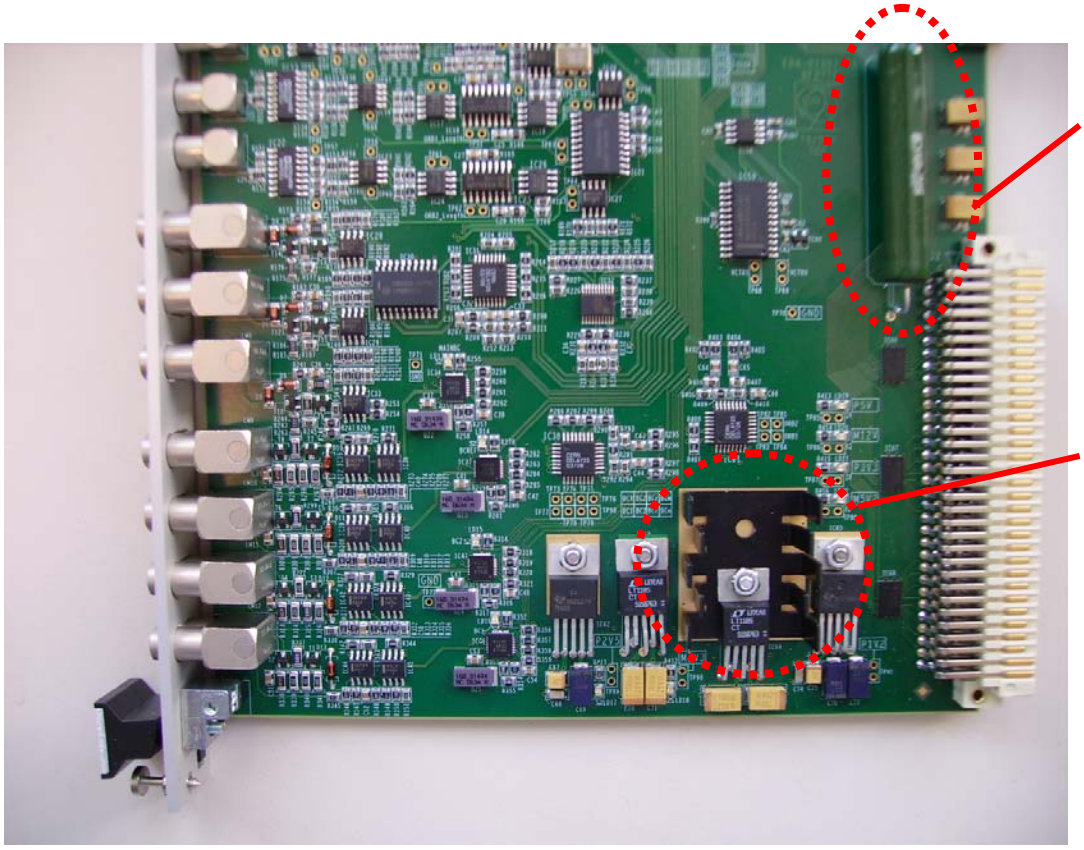
☐ Check the board visually according to the following pictures:

☐ Pay special attention to:

☐ TTCrx clock output resistor setting (bottom side of the TTCrx place):
(R430 should be mounted, R429 should not) to select the clock40 output of the TTCrx.

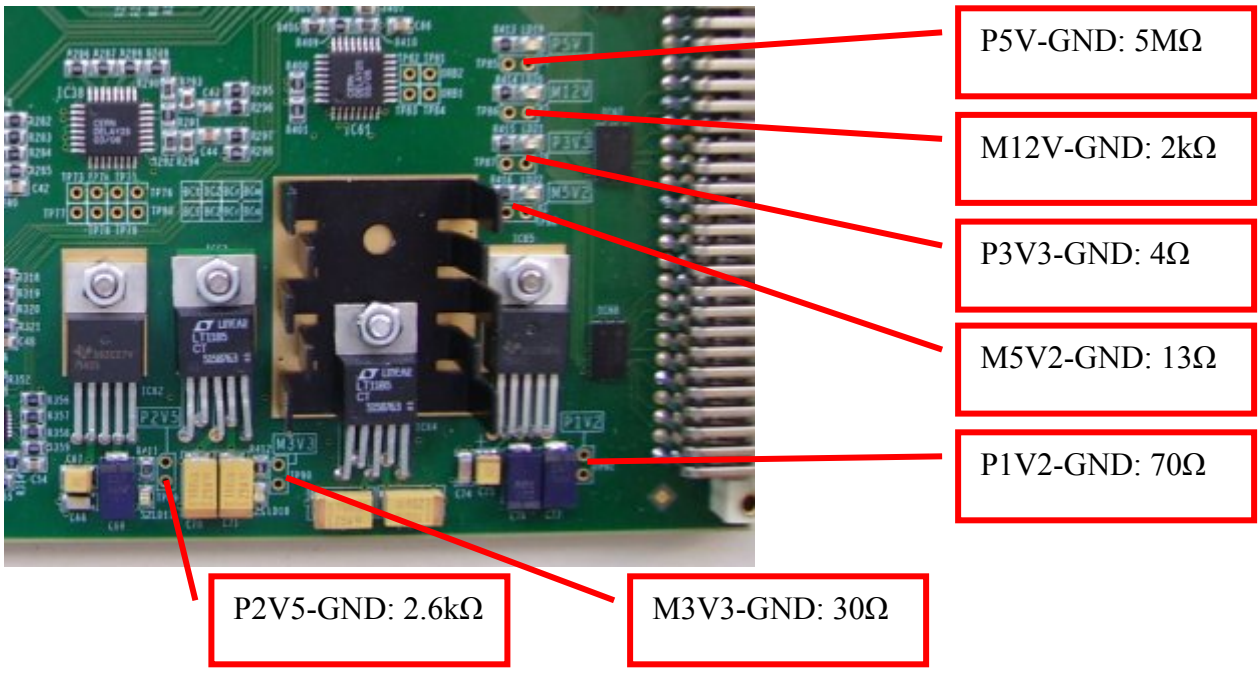



JTAG connector mounted
R17 should not be mounted
R36 and R37 mounted on the up position


Power Resistor body not in contact with the PCB to avoid thermal conduction
Heatsink installed

2. Power Supplies inspection

☐ Check the short circuits between power supply and ground using the test points on the bottom down side of the board:

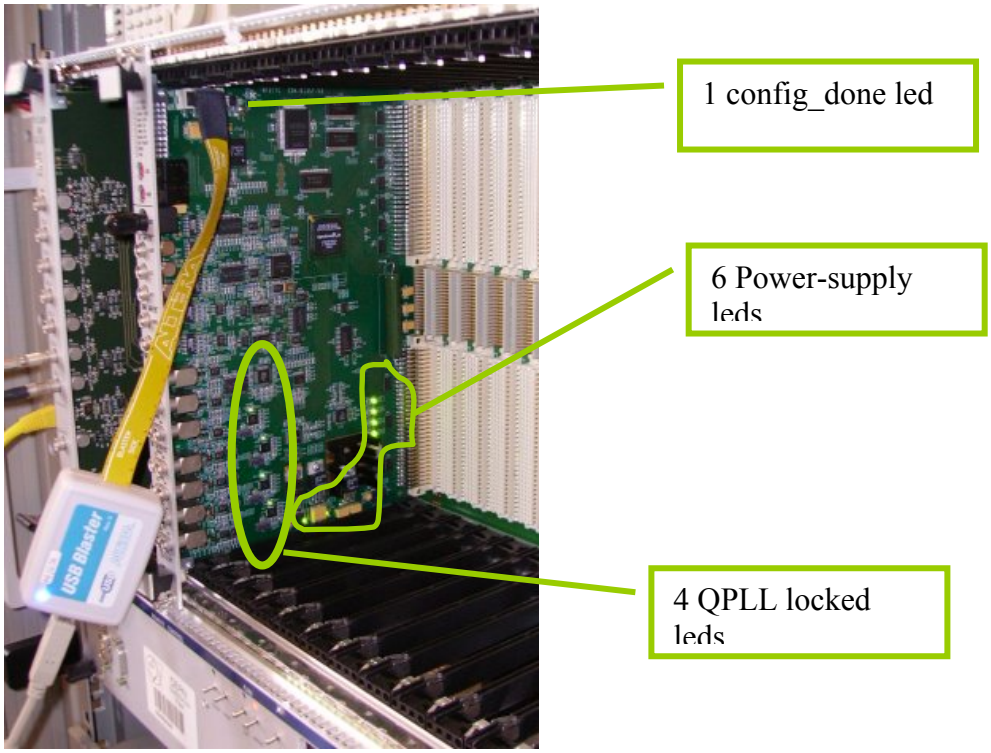


☐ Check the short circuits from one power plan to another using the same test points.

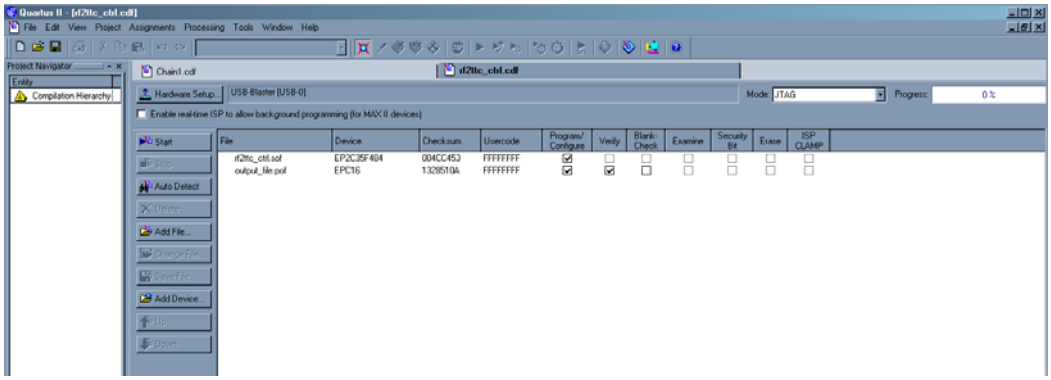
3. FPGA and EPROM configuration

- Insert the board in a 6U VME 64x crate

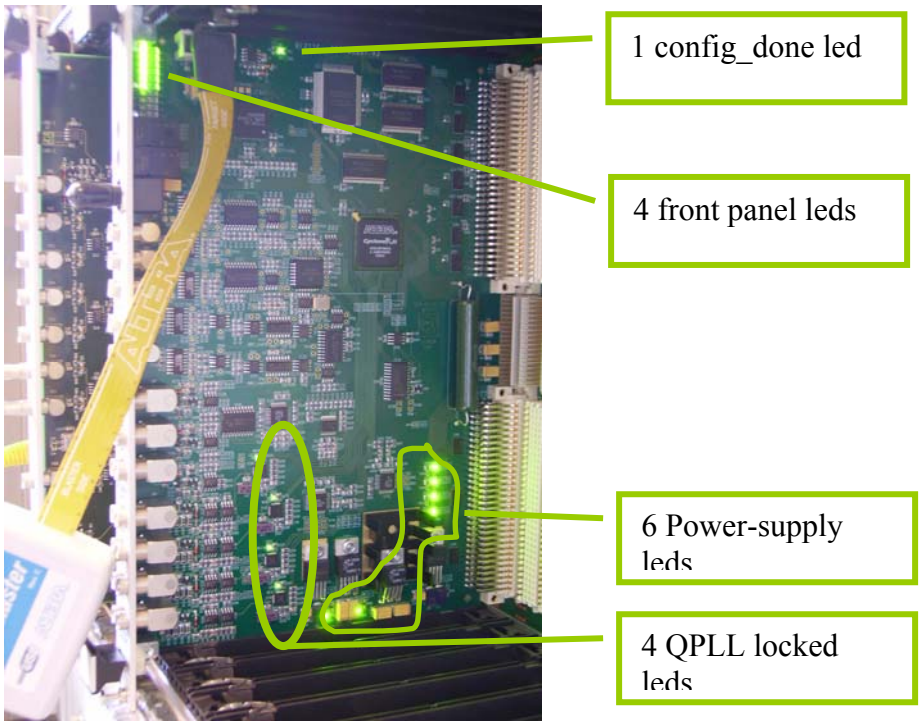
☐ Check the leds which should be on already:



- Connect the USB Blaster on the board, and launch the Quartus programmer.
- Open the rf2ttc_ctrl.cdf programmer file, calling the sof and the pof file ables to program respectively the FPGA and the EPRO. The .sof and .pof files, as well as the cdf file, are available on dfs\Services\cdsusers\sspriet\TTC\TTCupgrade\New_Design\RF2TTC_board\fpga\Quartus6.1\V3 Production firmware\RF2TTC_basic_08_05_2007.
- Press On 'Start'. The full configuration of the FPGA + of the EPROM should take about 5 minutes.
- Checksums:
  - Rf2ttc_ctrl.sof: 004CC453
  - Output_file.pof: 1328510A



☐ Once the programming is finished, the following leds must be ON:



4. Address tests [T1]

All the tests described in sections 4 and above are prepared in the software rf2ttcscope_atlice written by Markus Joos. (available on the CVS repository).

4.1. Geographical address test

5 bits of geographical address must be tested:

- In RF2TTC/bin, type rf2ttcscop_atlice –s 0x0F00000 and then choose 12 and run test 1.
- Exit the program
- Put the board on slot 15 with 0x00 as a Front Panel address.
- Switch on and read the register offset 0x00004 (VME read 0x0F00004).

☐ The value must be 0x0000016B.

☐ Check the VME led, which should light up during a VME access. If it is not working but the board is answering, check the front panel leds soldering: they should all be in the same direction

- In RF2TTC/bin, type rf2ttcscop_atlice –s 0x1000000 and then choose 12 and run test 1
- Exit the program
- Put the board on slot 16 with 0x00 as a Front Panel address.
- Switch on and read the register offset 0x00004 (VME read 0x1000004).

☐ The value must be 0x0000016B.

4.2. Manual address test

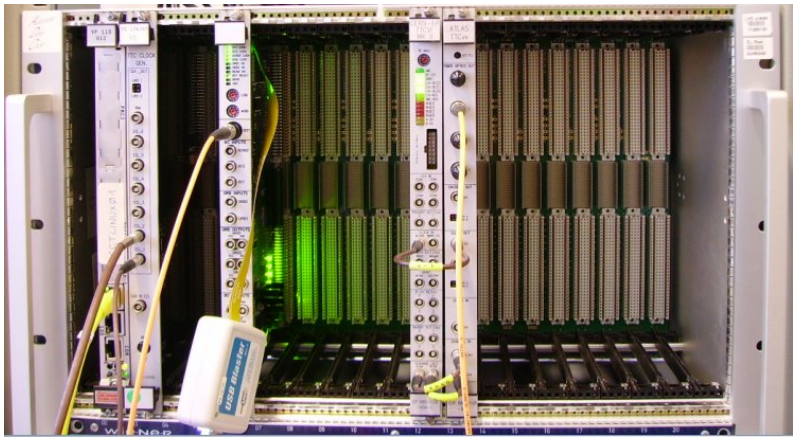
8 bits of manual address must be tested:

- In RF2TTC/bin, type rf2ttcscop_atlice –b 0xFF00000 and then choose 12 and run test 1
- Put the board on slot 5 with 0xFF as a Front Panel (manual) address.
- Switch on and read the register offset 0x00004 (VME read 0xFF00004).

☐ The value must be 0x0000016B.

5. VME registers check

- Connect the board to the TTCvi and the TTCvx as follows:



☐ Check that the 'BST ready' led is ON.

5.1. Board identification [T2]

☐ Check the following read only registers

	Offset	#bits	Default
PROGRAM ID	0x0000C	32	Firmware date number Version date: 08052007
REVISION ID	0x00008	32	Hardware version Production V3 = 0x3 Prototype V2= 0x2
BOARD ID	0x00004	32	0x0000016B
MANUFACTURER ID	0x00000	32	0x00080030 (CERN)

5.2. Registers Read/Write access [T3]

☐ For each of the following register, repeat the sequence:

1. Read + check if =A (this allows to verify the FPGA configuration + the init process)
2. Write (=> B)
3. Read + check if = C
4. Write back A

☐ For one of the access, check that the "VME" led on the front panel lights up during a VME access

	Address	#bits		1.Read A (Default)	2. Write B	3. Read Back C
BC1_MAN_SELECT	0x7FBFC	1	R/W	0	1	1
BC1_BEAM_SELECT	0x7FBF8	1	R/W	1	0	0
BC1_NOBEAM_SELECT	0x7FBF4	1	R/W	0	1	1
BC1_QPLL_MODE*	0x7FBF0	1	R/W	1	0	0
BC1_QPLL_STATUS	0x7FBE8	2	R	1	-	-
BC2_MAN_SELECT	0x7FBCC	1	R/W	0	1	1
BC2_BEAM_SELECT	0x7FBC8	1	R/W	1	0	0
BC2_NOBEAM_SELECT	0x7FBC4	1	R/W	0	1	1
BC2_QPLL_MODE*	0x7FBC0	1	R/W	1	0	0
BC2_QPLL_STATUS	0x7FBB8	2	R	1	-	-
BCref_MAN_SELECT	0x7FBAC	1	R/W	0	1	1
BCref_BEAM_SELECT	0x7FBA8	1	R/W	1	0	0
BCref_NOBEAM_SELECT	0x7FBA4	1	R/W	0	1	1
BCref_QPLL_MODE*	0x7FBA0	1	R/W	1	0	0
BCref_QPLL_STATUS	0x7FB98	2	R	1	-	-
BCmain_MAN_SELECT	0x7FB8C	2	R/W	0	1	1
BCmain_BEAM_SELECT	0x7FB88	2	R/W	1	0	0
BCmain_NOBEAM_SELECT	0x7FB84	2	R/W	0	1	1
BCmain_QPLL_MODE*	0x7FB80	1	R/W	1	0	0
BCmain_QPLL_STATUS	0x7FB7C	2	R	1	-	-
ORB1_MAN_SELECT	0x7FB6C	1	R/W	1	0	0
ORB1_BEAM_SELECT	0x7FB68	1	R/W	0	1	1
ORB1_NOBEAM_SELECT	0x7FB64	1	R/W	1	0	0
ORB1_POLARITY	0x7FB60	1	R/W	0	1	1
ORB1_COARSE_DELAY	0x7FB5C	12	R/W	0	0x10	0x10
ORB1_LENGTH	0x7FB58	8	R/W	0	0x5	0x5
ORB1_INT_PERIOD_SET	0x7FB54	12	R/W	0xdec	0x20	0x20
ORB1_INT_PERIOD_COUNTER	0x7FB50	12	R	Something >0	-	-
ORB1_COUNTER	0x7FB4C	32	R	Something >0	-	-
ORB1_PERIOD_RD	0x7FB48	12	R	0xded	-	-
ORB1_PERIOD_FIFO_STATUS	0x7FB44	2	R	0x2 (full because of necessary delays)	-	-
ORB1_PERIOD_FIFO_RD (twice) (check only the 2nd value)	0x7FB40	16	R	1.Something 2.0xded	-	-

ORB1_DAC	0x7FB3C	8	R/W	0xAA	0x22	0x22
ORB2_MAN_SELECT	0x7FB2C	1	R/W	1	0	0
ORB2_BEAM_SELECT	0x7FB28	1	R/W	0	1	1
ORB2_NOBEAM_SELECT	0x7FB24	1	R/W	1	0	0
ORB2_POLARITY	0x7FB20	1	R/W	0	1	1
ORB2_COARSE_DELAY	0x7FB1C	12	R/W	0	0x10	0x10
ORB2_LENGTH	0x7FB18	8	R/W	0	0x5	0x5
ORB2_INT_PERIOD_SET	0x7FB14	12	R/W	0xdec	0x20	0x20
ORB2_INT_PERIOD_COUNTER	0x7FB10	12	R	Something > 0	-	-
ORB2_COUNTER	0x7FB0C	32	R	Something > 0	-	-
ORB2_PERIOD_RD	0x7FB08	12	R	0xded	-	-
ORB2_PERIOD_FIFO_STATUS	0x7FB04	2	R	0x2	-	-
ORB2_PERIOD_FIFO_RD(twice) (check only the 2nd value)	0x7FB00	16	R	Something 0xded	-	-
ORB2_DAC	0x7FAFC	8	R/W	0xAA	0x22	0x22
ORBmain_MAN_SELECT	0x7FAEC	2	R/W	2	0	0
ORBmain_BEAM_SELECT	0x7FAE8	2	R/W	0	1	1
ORBmain_NOBEAM_SELECT	0x7FAE4	2	R/W	2	0	0
ORBmain_POLARITY	0x7FAE0	1	R/W	0	1	1
ORBmain_COARSE_DELAY	0x7FADC	12	R/W	0	0x10	0x10
ORBmain_LENGTH	0x7FAD8	8	R/W	0	0x5	0x5
ORBmain_INT_PERIOD_SET	0x7FAD4	12	R/W	0xdec	0x20	0x20
ORBmain_INT_PERIOD_COUNTER	0x7FAD0	12	R	Something >0	-	-
ORBmain_COUNTER	0x7FACC	32	R	Something >0	-	-
ORBmain_PERIOD_RD	0x7FAC8	12	R	0xded	-	-
ORBmain_PERIOD_FIFO_STATUS	0x7FAC4	2	R	0x2	-	-
ORBmain_PERIOD_FIFO_RD (twice) Check only the 2nd value	0x7FAC0	16	R	Something 0xded	-	-
TTCrx_status	0x7FAA0	1	R	1	-	-
BST_Machine_Mode	0x7FA9C	32	R	0x0000	-	-
BEAM_NO_BEAM_DEF	0x7FA7C	32	R/W	0x0008	0xCAFE	0xCAFE
WORKING_MODE	0x7FA78	7	R/W	0	0x7F	0x7F
ORB_INT_ENABLE	0x7FA6C	3	R/W	0x7	0x0	0x0
ORB_COUNTER_ENABLE	0x7FA68	3	R/W	0x7	0x0	0x0
PERIOD_COUNTER_ENABLE	0x7FA64	3	R/W	0x7	0x0	0x0

TTCrx_CONTROL REGISTER	0x7E000	9	(R)/W	W 0x7e000 0x3 R 0x7e000 - R 0x7e200 => 0x1ff	W 0x7e000 0x3 W 0x7e004 0xFB	W 0x7e000 0x3 R 0x7e000 - R 0x7e200 => 0x1FB
ORBOUT_DELAY25_ORBmain	0x7D048	9	(R)/W	R 0x7d048 R 0x7d200 =>0x140	W 0x7d048 0x7F	R 0x7d048 R 0x7d200 =>0x17F
ORBOUT_DELAY25_ORB2	0x7D044	9	(R)/W	R 0x7d044 R 0x7d200 =>0x140	W 0x7d044 0x7E	R 0x7d044 R 0x7d200 =>0x17E
ORBOUT_DELAY25_ORB1	0x7D040	9	(R)/W	R 0x7d040 R 0x7d200 =>0x140	W 0x7d040 0x7C	R 0x7d040 R 0x7d200 =>0x17C
ORBIN_DELAY25_ORB2	0x7D024	9	(R)/W	R 0x7d024 R 0x7d200 =>0x140	W 0x7d024 0x7B	R 0x7d024 R 0x7d200 =>0x17B
ORBIN_DELAY25_ORB1	0x7D020	9	(R)/W	R 0x7d020 R 0x7d200 =>0x140	W 0x7d020 0x7A	R 0x7d020 R 0x7d200 =>0x17A
BC_DELAY25_BCmain	0x7D00c	9	(R)/W	R 0x7d00c R 0x7d200 =>0x140	W 0x7d00c 0x79	R 0x7d00c R 0x7d200 =>0x179
BC_DELAY25_BCref	0x7D008	9	(R)/W	R 0x7d008 R 0x7d200 =>0x140	W 0x7d008 0x78	R 0x7d008 R 0x7d200 =>0x178
BC_DELAY25_BC2	0x7D004	9	(R)/W	R 0x7d004 R 0x7d200 =>0x140	W 0x7d004 0x77	R 0x7d004 R 0x7d200 =>0x177
BC_DELAY25_BC1	0x7D000	9	(R)/W	R 0x7d000 R 0x7d200 =>0x140	W 0x7d000 0x76	R 0x7d000 R 0x7d200 =>0x176

* add 500ms delay after the end of the line (after writing back A value)

5.3. Reset commands [T4]

☐ Four different reset actions have to be tested

☐ Delay25 RESET: do the following sequence:

1. read delays values (should be like column 1."Read A (Default)" of the previous table)
2. write 0x01 on the BSET register 0x00010
3. write 0x01 on the BCLEAR register 0x00014
4. read delay values (should all be 0x00)
5. write back 0x040 in all the 8 delay25 registers
6. delay of 500ms before next test

☐ QPLL RESET: do the following sequence:

1. read QPLL status values (should all be 0x01), and check that the 4 QPLL leds are ON

	Address
BC1_QPLL_STATUS	0x7FBE8
BC2_QPLL_STATUS	0x7FBB8
BCref_QPLL_STATUS	0x7FB98
BCmain_QPLL_STATUS	0x7FB7C

2. write 0x02 on the BSET register 0x00010
3. read QPLL status values (should all be 0x00), and the 4 QPLL leds shoulds be off
4. write 0x02 on the BCLEAR register 0x00014
5. wait 500ms
6. read back the QPLL status values (should all be 0x01), and the 4 QPLL leds shoulds be ON

☐ TTCrx RESET: do the following sequence:

1. read TTCrx status value (0x7FAA0). It must be 0x01
2. write 0x04 on the BSET register 0x00010
3. read TTCrx status value (0x7FAA0). It must be 0x00 and the BST ready led must be off.
4. write 0x04 on the BCLEAR register 0x00014
5. wait 2ms
6. read back TTCrx status value (0x7FAA0). It must be 0x01

☐ Global RESET: do the following sequence:

1. write 0x08 on the BSET register 0x00010
2. read the following registers:

	Address	Expected Value
BC1_QPLL_STATUS	0x7FBE8	0x00
TTCrx_status	0x7FAA0	0x00
ORB1_COUNTER	0x7FB4C	0x00
ORB1_PERIOD_RD	0x7FB48	0x00
ORB1_PERIOD_FIFO_STATUS	0x7FB44	0x01
ORB1_PERIOD_FIFO_RD twice	0x7FB40	0x4000

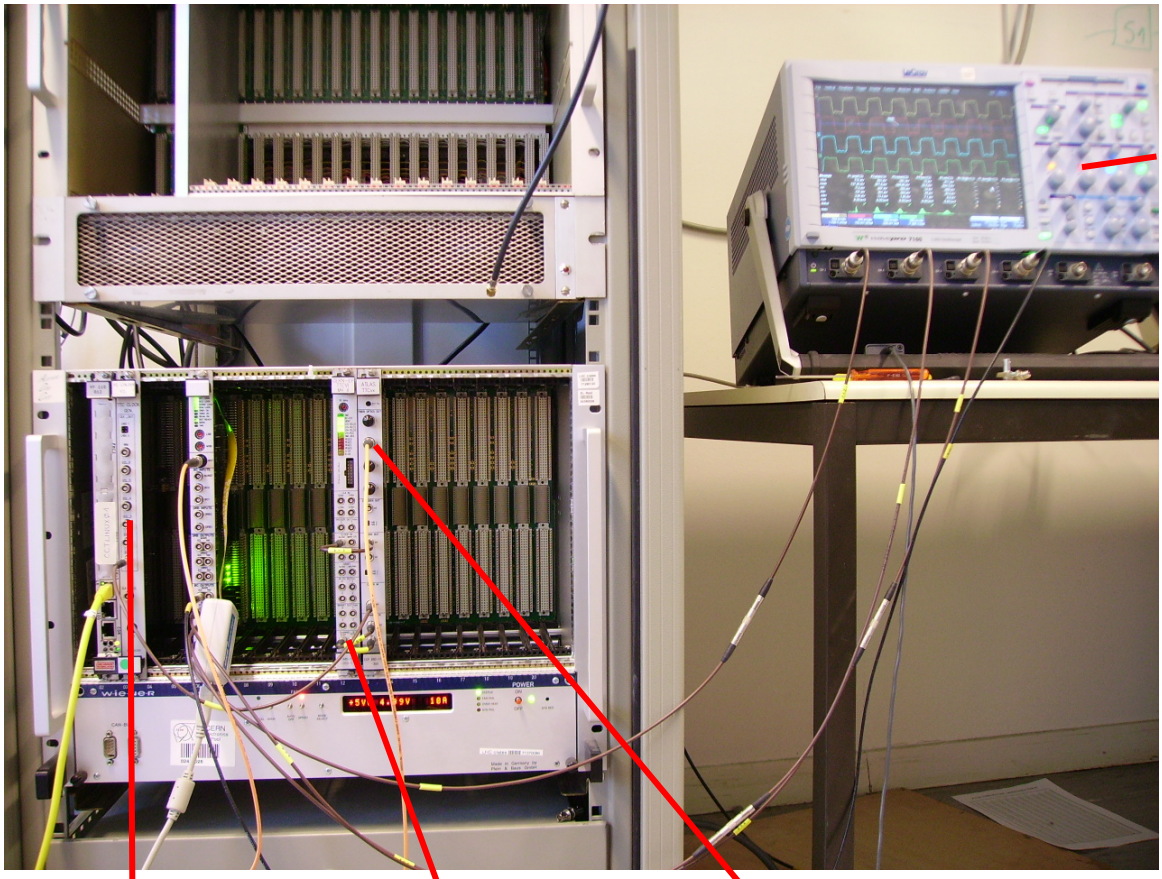
3. write 0x08 on the BCLEAR register 0x00014
4. wait 500ms
5. read back the following registers:
6. reinitialize the delay25 (enable them with 0x40)

	Address	Expected Value
BC_DELAY25_BC1	0x7D000	0x00
BC1_QPLL_STATUS	0x7FBE8	0x01
TTCrx_status	0x7FAA0	0x01
ORB1_COUNTER	0x7FB4C	Something>0
ORB1_PERIOD_RD	0x7FB48	0xded
ORB1_PERIOD_FIFO_STATUS	0x7FB44	0x2
ORB1_PERIOD_FIFO_RD twice	0x7FB40	Something>0

6. Bunch Clock Tests

6.1. Bunch Clock test setup

Connect the board with the following setup.



Lecroy scope 7100
-BC1 ECL out on IN1
-BC2 ECL out on IN2
-BCref ECL out on IN3
-BCmain ECL out on IN4
-recall the setup: "rf2ttc-bctest"

TTC clock gen in INTERNAL mode

TTCvi. Internal address 0xFF10

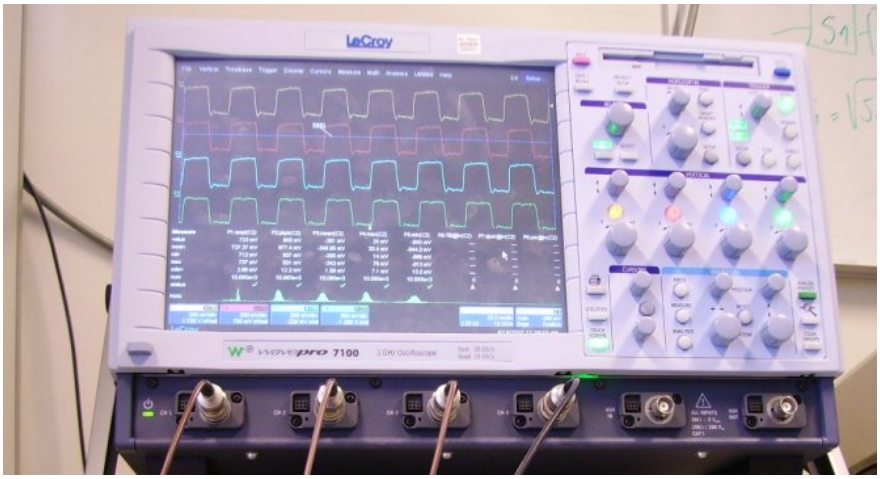
TTCvx.

6.2. Internal Bunch Clocks [T5]

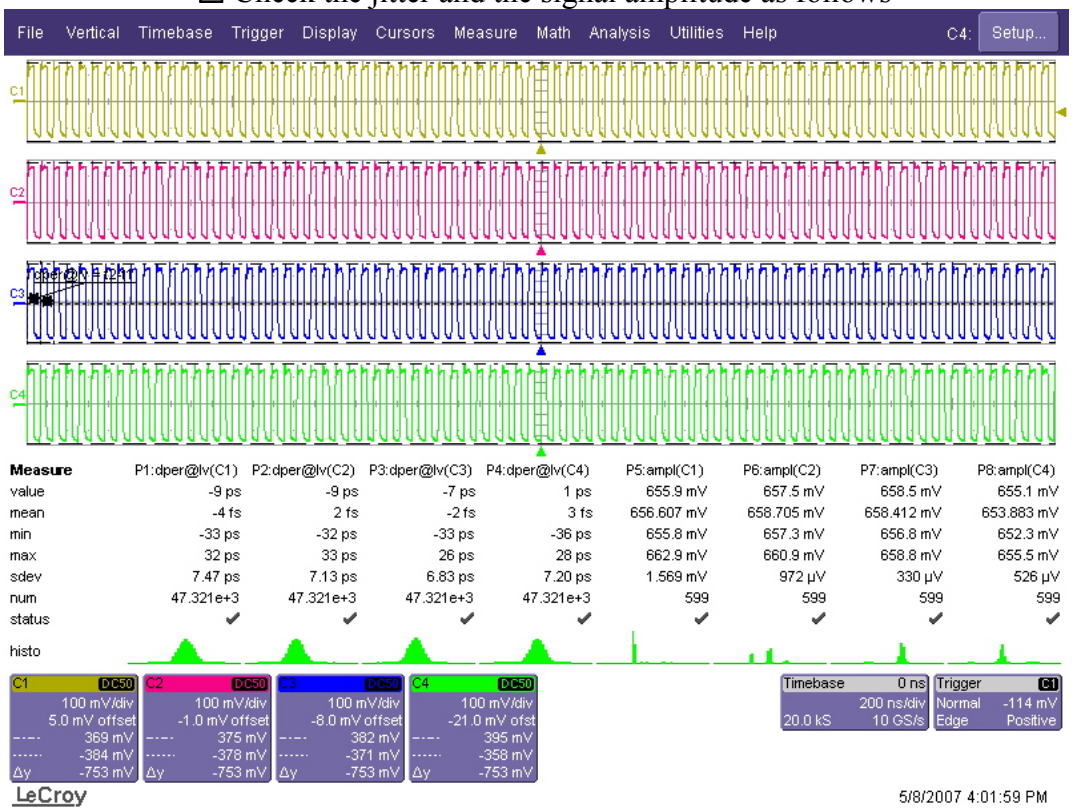
- Do ***not*** connect any external clock on the front panel
- Select all the Bunch Clocks to be INTERNAL
- Ensure that the board is in 'manual' mode.

	Address	Write
WORKING_MODE	0x7FA78	0x00
BC1_MAN_SELECT	0x7FBFC	0x00
BC2_MAN_SELECT	0x7FBCC	0x00
BCref_MAN_SELECT	0x7FBAC	0x00
BCmain_MAN_SELECT	0x7FB8C	0x00

☐ Check that the 4 signals on the scope are synchronized to each other:



☐ Check the jitter and the signal amplitude as follows



6.3. External Bunch Clocks [T6]

- Keep the BC inputs on the front panel UNCONNECTED
- Select all the Bunch Clocks to be EXTERNAL and ensure that the board is in 'manual' mode.

	Address	Write
WORKING_MODE	0x7FA78	0x00
BC1_MAN_SELECT	0x7FBFC	0x01
BC2_MAN_SELECT	0x7FBCC	0x01
BCref_MAN_SELECT	0x7FBAC	0x01
BCmain_MAN_SELECT	0x7FB8C	0x01

☐ Check that the QPLLs lose the lock: status register values should be 0x00 and the leds should be OFF.

	Address
BC1_QPLL_STATUS	0x7FBE8
BC2_QPLL_STATUS	0x7FBB8
BCref_QPLL_STATUS	0x7FB98
BCmain_QPLL_STATUS	0x7FB7C

- Connect TTC Clock Gen outputs on the BC inputs on the front panel, as on the following picture:

☐ Check that the leds are back ON
☐ Check that the 4 signals on the scope are synchronized to each other:
☐ Clear sweep the scope and check the jitter (again around 7ps rms)

☐ Check the NIM BC outputs, change the scope resolution and the trigger level: should be synchronized and their jitter should be reasonably low (around 9-10ps rms).

6.4. BCmain selection [T 7]

- Keep the setup as previously (with the external BC inputs connected).
- ☐ For each of the following case, check that the BCmain is correctly locked to the expected input:

	BC1_MAN_SELECT	BC2_MAN_SELECT	BCref_MAN_SELECT	BCmain_MAN_SELECT	What you should see
☐	0x00 (int)	0x01 (ext)	0x01 (ext)	0x03 (BC1)	BC1-BCmain locked BC2-BCref locked
☐	0x01 (ext)	0x00 (int)	0x01 (ext)	0x02 (BC2)	BC2-BCmain locked BC1-BCref locked
☐	0x01 (ext)	0x01 (ext)	0x00 (int)	0x01 (BCref)	BCref-BCmain locked BC2-BC1 locked
☐	0x01 (ext)	0x00 (int)	0x00 (int)	0x03 (BC1)	BC1-BCmain locked BC2-BCref locked
☐	0x00 (int)	0x01 (ext)	0x00 (int)	0x02 (BC2)	BC2-BCmain locked BC1-BCref locked
☐	0x00 (int)	0x00 (int)	0x01 (ext)	0x01 (BCref)	BCref-BCmain locked BC2-BC1 locked
☐	0x01 (ext)	0x01 (ext)	0x01 (ext)	0x03 (BC1)	All locked

6.5. BC delays [T8]

- Keep the last configuration from previous test
- Increase the horizontal resolution of the scope to see 1 period only
- Set trigger on BC1 and put the cursor on BC rising edges
- Successively write 10 (0xA) on each BC delay
- ☐ check that the rising edge of the corresponding BC on the scope moves from 5ns.

REGISTER NAME	ADDRESS	
BC_DELAY25_BCmain	0x7D00c	W 0x7d00c 0x4A
BC_DELAY25_BCref	0x7D008	W 0x7d008 0x4A
BC_DELAY25_BC2	0x7D004	W 0x7d004 0x4A
BC_DELAY25_BC1	0x7D000	W 0x7d000 0x4A

PS: when BC1 is delayed, all the signals are back to their initial configuration

7. Orbit tests

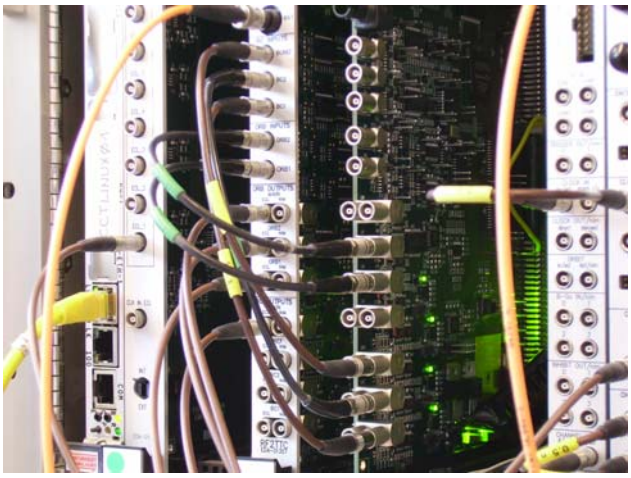
7.1. New test setup (configuration made at the beginning of T9)

- Turn off the crate
- Plug the RF2TTC V2 (tester board) configured with manual address 0x02
- Configure it in manual mode, with internal BC and Orbit (to be used as a source for the board to be tested)
- Set the internal orbit period to Orb1=0xabd, Orb2=0x123, Orbmain=0x456

RF2TTC V2 BOARD – ADDRESS 0x02XXXX (XXXXX= register offset)

REGISTER NAME	OFFSET	VALUE
WORKING_MODE	0x7FA78	0x00
BC1_MAN_SELECT	0x7FBFC	0x00
BC2_MAN_SELECT	0x7FBCC	0x00
BCref_MAN_SELECT	0x7FBAC	0x00
BCmain_MAN_SELECT	0x7FB8C	0x00
ORB1_MAN_SELECT	0x7FB6C	0x01
ORB2_MAN_SELECT	0x7FB2C	0x01
ORBmain_MAN_SELECT	0x7FAEC	0x02 (int)
ORB1_INT_PERIOD_SET	0x7FB54	0xabc
ORB2_INT_PERIOD_SET	0x7FB14	0x123
ORBmain_INT_PERIOD_SET	0x7FAD4	0x456

- Connect the RF2TTC V2 Board to the RF2TTC V3 board to be tested as follows:



7.2. External Orbit test [T9]

- Configure the board to be tested with external BC and Orbit (BCmain to BCref et Orbmain to Orb1) and internal orbit periods to 0xdec

RF2TTC V3 BOARD – ADDRESS 0xFFXXXX (XXXXX= register offset)

REGISTER NAME	OFFSET	WRITE VALUE
WORKING_MODE	0x7FA78	0x00
BC1_MAN_SELECT	0x7FBFC	0x01
BC2_MAN_SELECT	0x7FBCC	0x01
BCref_MAN_SELECT	0x7FBAC	0x01
BCmain_MAN_SELECT	0x7FB8C	0x01 (BCref)
ORB1_MAN_SELECT	0x7FB6C	0x00
ORB2_MAN_SELECT	0x7FB2C	0x00
ORBmain_MAN_SELECT	0x7FAEC	0x00 (orb1)
ORB1_INT_PERIOD_SET	0x7FB54	0xdec
ORB2_INT_PERIOD_SET	0x7FB14	0xdec
ORBmain_INT_PERIOD_SET	0x7FAD4	0xdec
ORB1_DAC	0x7FB3C	0xAA
ORB2_DAC	0x7FAFC	0xAA

☐ Check the orbit LEDS on the front panel (should be ON)
☐ Check that the 3 orbit counters are increasing (soft)

REGISTER NAME	OFFSET
ORB1_COUNTER	0x7FB4C
ORB2_COUNTER	0x7FB0C
ORBmain_COUNTER	0x7FACC

- Reset the orbit period counters

REGISTER NAME	OFFSET	WRITE VALUE
PERIOD_COUNTER_RESET	0x7FA48	0xF

☐ Check the following registers (soft)

REGISTER NAME	OFFSET	EXPECTED VALUE
ORB1_PERIOD_RD	0x7FB48	0xabd
ORB1_PERIOD_FIFO_RD (twice)	0x7FB40	First: anything Second: 0xabd
ORB2_PERIOD_RD	0x7FB08	0x124
ORB2_PERIOD_FIFO_RD(twice)	0x7FB00	First: anything Second: 0x124
ORBmain_PERIOD_RD	0x7FAC8	0xabd
ORBmain_PERIOD_FIFO_RD (twice)	0x7FAC0	First: anything Second: 0xabd

- Keep the above configuration, set the RF2TTC V2 board tester orbit periods to 0x100:

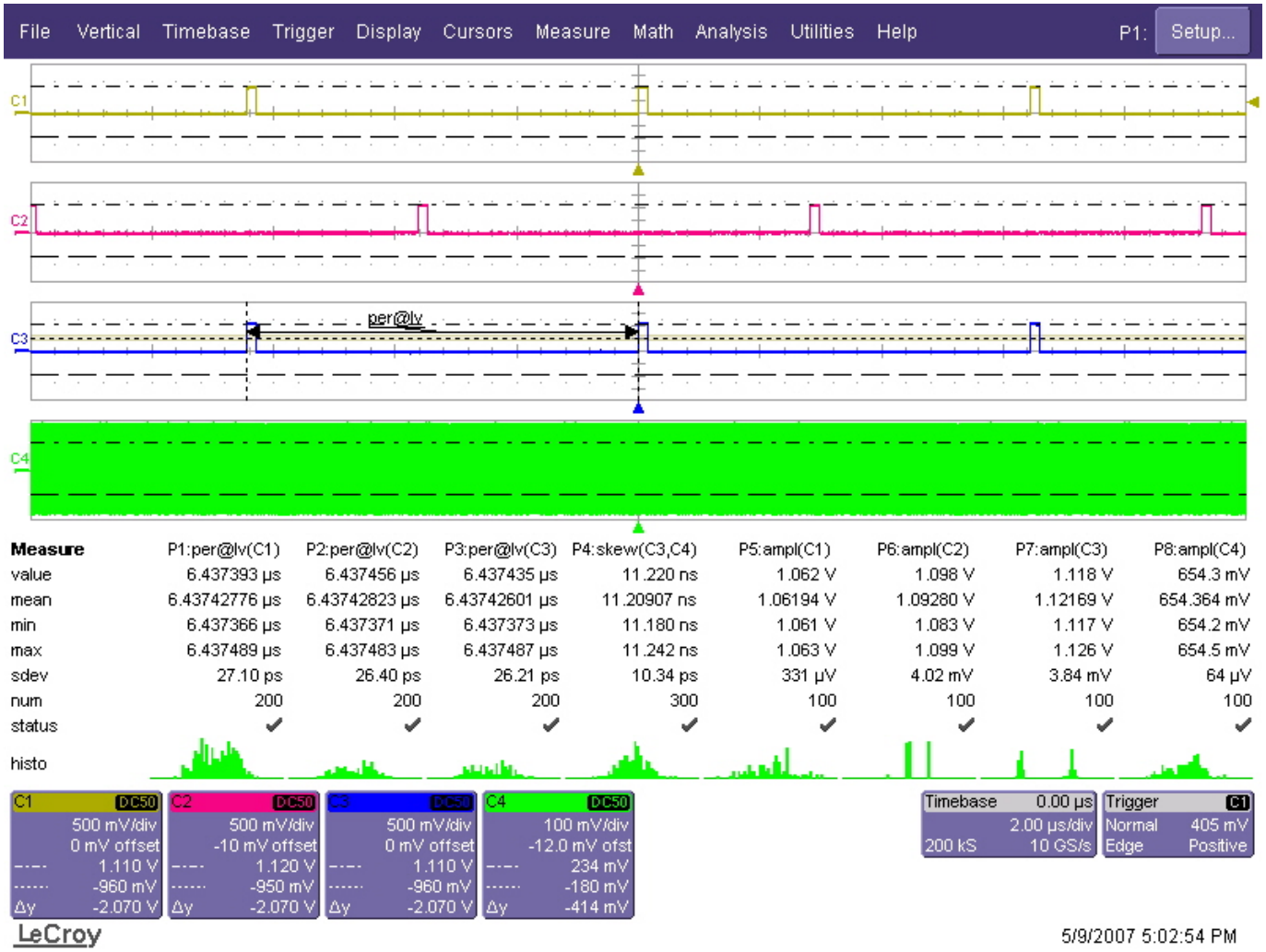
REGISTER NAME	OFFSET	WRITE VALUE
ORB1_INT_PERIOD_SET	0x7FB54	0x100
ORB2_INT_PERIOD_SET	0x7FB14	0x100
ORBmain_INT_PERIOD_SET	0x7FAD4	0x100

7.3. Orbit outputs quality [T9 continued]

- Connect the oscilloscope as follows:

- Orb1 out => scope IN1
- Orb2 out => scope IN2
- MainOrb out => scope IN3
- BCmain out => scope IN4
- Recall “rf2ttc-orbit-test” setup

☐ Control the Jitter and the amplitude as follows:



Typical: jitter 25ps rms, amplitude 1.1V.
With very good parameters (vertical and horizontal resolution), the jitter goes down to 15ps rms.

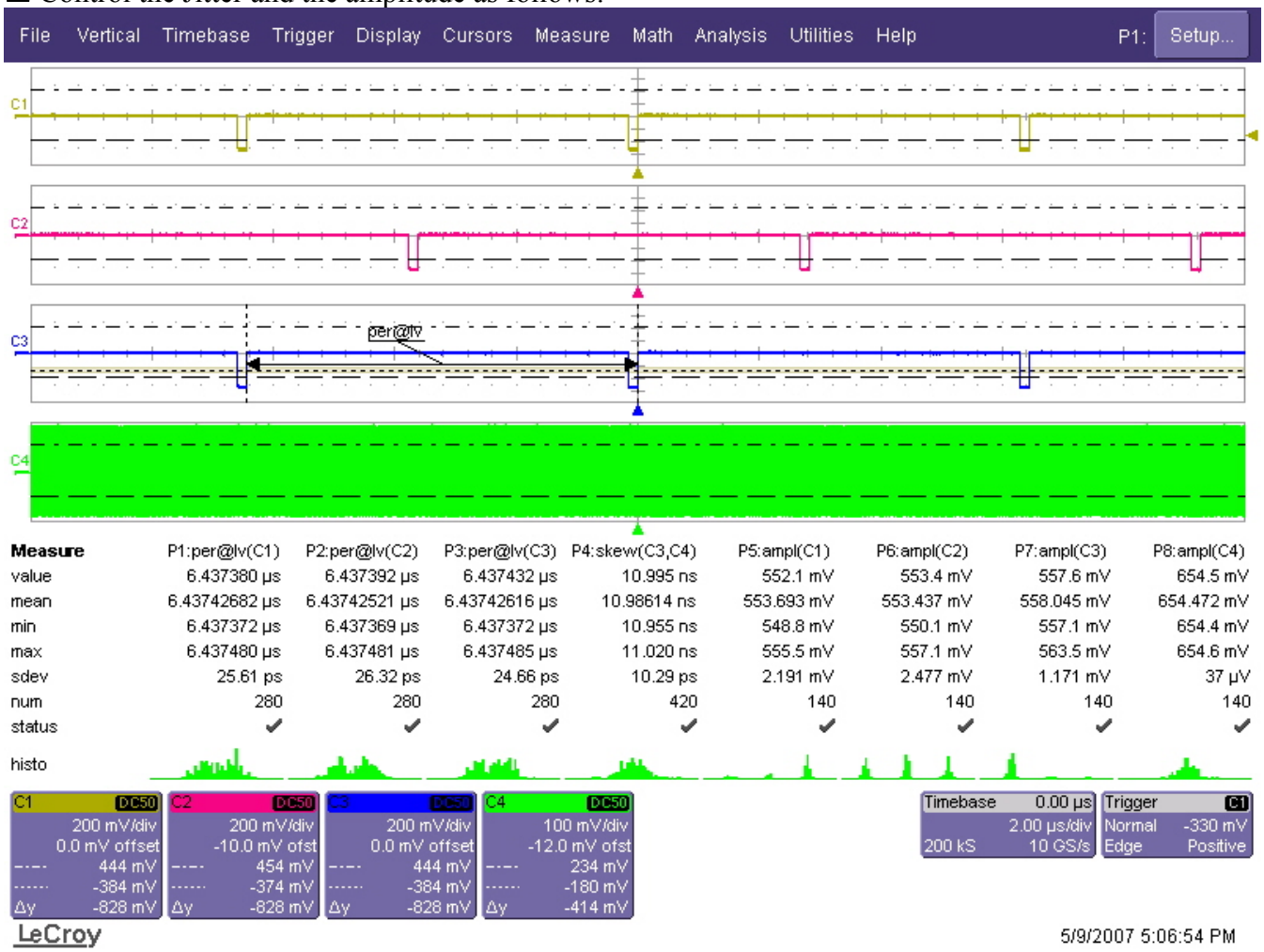
☐ Check the corresponding NIM outputs

7.4. Orbit Polarity [T10]

- Invert the orbit polarity:

REGISTER NAME	OFFSET	WRITE VALUE
ORB1_POLARITY	0x7FB60	0x1
ORB2_POLARITY	0x7FB20	0x1
ORBmain_POLARITY	0x7FAE0	0x1

☐ Control the Jitter and the amplitude as follows:



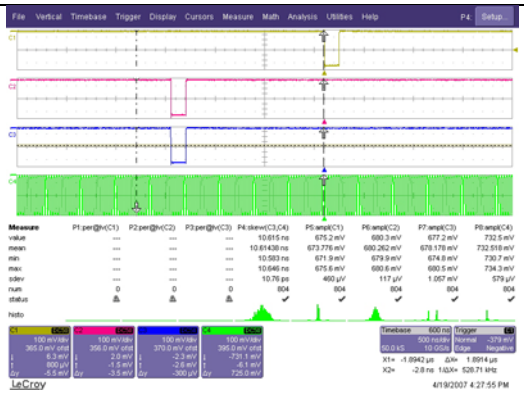
Typical: jitter 25ps rms, amplitude 560mV.

7.5. Coarse Delay test [T11]

- o Increase the scope horizontal resolution to 500ns/div
- o Put horizontal cursors between orbit edges
- o Set Orbitmain to orb2 and write 0xF in the coarse delay of orb2, and then to ORBmain

☐ check the effect on the scope (the IN2 and IN3 should successively be shifted by 375ns Bunch clocks compared to the cursors position)

REGISTER NAME	OFFSET	WRITE VALUE
ORBmain_MAN_SELECT	0x7FAEC	0x01 (orb2)
ORB2_COARSE_DELAY	0x7FB1C	0x0F
ORBmain_COARSE_DELAY	0x7FADC	0x0F



o Keep this configuration and write 0xF in the coarse delay of orb1
☐ check the effect on the scope. All the pulses should be back to their original position:

REGISTER NAME	OFFSET	WRITE VALUE
ORB1_COARSE_DELAY	0x7FB5C	0x0F

7.6. Orbit Length Adjustment test [T12]

o Keep this configuration and write 0xF in the orbit length registers
☐ check the effect

REGISTER NAME	OFFSET	WRITE VALUE
ORB1_LENGTH	0x7FB58	0x0F
ORB2_LENGTH	0x7FB18	0x0F
ORBmain_LENGTH	0x7FAD8	0x0F

7.7. Orbit input threshold test [T13]

o Run the calibrate voltage threshold application from RF2TTCscope for orbit1
☐ check that the result gives a window between about [0x80 and 0xD5], and advises to use about 0xAA as a DAC value

o Run the calibrate voltage threshold application from RF2TTCscope for orbit2
☐ check that the result gives a window between about [0x80 and 0xD5], and advises to use about 0xAA as a DAC value

7.8. Fine shift of orbit inputs test [T14]

o Run the calibrate orbit application from RF2TTCscope for orbit1
☐ check that the result gives a window of about 48 steps

- Run the calibrate orbit application from RF2TTCscope for orbit2

☐ check that the result gives a window of about 48 steps

7.9. Fine delay of orbit output [T15]

- Increase the scope horizontal resolution to 25ns/div
- Put horizontal cursors on orbit1 edge
- Set Orbitmain to orb2 and, after having put horizontal cursor on the right orbit edge, write successively 0xA in the fine delay of orbmain, orb2 and finally orb1

☐ check the effect on the scope for each orbit (the pulse should successively be shifted by 5ns compared to the cursors position). Rem: for Orb1, put the trigger on orb2 if you want to actually see the delay on the orb1 signal.

REGISTER NAME	OFFSET	WRITE VALUE
ORBOUT_DELAY25_ORBmain	0x7D048	0x4A
ORBOUT_DELAY25_ORB2	0x7D044	0x4A
ORBOUT_DELAY25_ORB1	0x7D040	0x4A

7.10. Internal orbit test [T16]

- Set the scope resolution back to 2us/div
- Before running the test, measure the period of the orbits (should be 6.4 us)
- Configure the board to be tested with external BC and internal Orbit (BCmain to BCref) and internal orbit periods to 0xdec

RF2TTC V3 BOARD – ADDRESS 0xFFXXXX (XXXXX= register offset)

REGISTER NAME	OFFSET	VALUE
WORKING_MODE	0x7FA78	0x00
BC1_MAN_SELECT	0x7FBFC	0x01
BC2_MAN_SELECT	0x7FBCC	0x01
BCref_MAN_SELECT	0x7FBAC	0x01
BCmain_MAN_SELECT	0x7FB8C	0x01
ORB1_MAN_SELECT	0x7FB6C	0x01
ORB2_MAN_SELECT	0x7FB2C	0x01
ORBmain_MAN_SELECT	0x7FAEC	0x02 (internal)
ORB1_INT_PERIOD_SET	0x7FB54	0xdec
ORB2_INT_PERIOD_SET	0x7FB14	0xdec
ORBmain_INT_PERIOD_SET	0x7FAD4	0xdec

ORB1_DAC	0x7FB3C	0xAA
ORB2_DAC	0x7FAFC	0xAA

☐ check that the 3 signals are synchronized on the scope
☐ The period should be about 89us (to be able to see it on the scope, you have to change the sampling rate from 10GS/s down to 250MS/s or so.)

8. BST/TTCrx test

8.1. Configuration [T17] – not a test, only the configuration setup

- Ensure that the TTCvi and the TTCvx are connected as described in the previous setups
- Configure the board to analyse the various LHC machine mode transmitted by the BST:

REGISTER NAME	OFFSET	VALUE
BC1_BEAM_SELECT	0x7FBF8	External 1
BC1_NOBEAM_SELECT	0x7FBF4	Internal 0
BC2_BEAM_SELECT	0x7FBC8	External 1
BC2_NOBEAM_SELECT	0x7FBC4	Internal 0
BCref_BEAM_SELECT	0x7FBA8	External 1
BCref_NOBEAM_SELECT	0x7FBA4	Internal 0
BCmain_BEAM_SELECT	0x7FB88	BCref 1
BCmain_NOBEAM_SELECT	0x7FB84	Internal 0
ORB1_BEAM_SELECT	0x7FB68	External 0
ORB1_NOBEAM_SELECT	0x7FB64	Internal 1
ORB2_BEAM_SELECT	0x7FB28	External 0
ORB2_NOBEAM_SELECT	0x7FB24	Internal 1
ORBmain_BEAM_SELECT	0x7FAE8	Orb1 0
ORBmain_NOBEAM_SELECT	0x7FAE4	Internal 2
TTCrx pointer to the [control] register	0x7E000	0x03
TTCrx pointer to the data [of the control register]	0x7E004	0xFF
WORKING_MODE	0x7FA78	0x01

8.2. No Beam machine mode [T18]

- Set the TTCvi to send the “NO BEAM” mode

ACTION	ADDRESS	VALUE
WRITE	0xFF10C0	0x8001
WRITE	0xFF10C2	0x1B00

☐ check that the “BST ready” led is ON
☐ check that the “BEAM ON” led is OFF

☐ check the BST mode register (done by the software)

REGISTER NAME	OFFSET	EXPECTED VALUE
BST_Machine_Mode	0x7FA9C	0x00

☐ check that the signals are following the "NOBEAM" configuration (all internal) (optional)

8.3. Filling machine mode [T19]

- Set the TTCvi to send the "FILLING" mode

ACTION	ADDRESS	VALUE
WRITE	0xFF10C0	0x8001
WRITE	0xFF10C2	0x1B01

☐ check that the "BST ready" led is ON
☐ check that the "BEAM ON" led is OFF

☐ check the BST mode register (soft)

REGISTER NAME	OFFSET	EXPECTED VALUE
BST_Machine_Mode	0x7FA9C	0x01

☐ check that the signals are following the "NOBEAM" configuration (all internal) - optional

8.4. Ramping machine mode [T20]

- Set the TTCvi to send the "RAMPING" mode

ACTION	ADDRESS	VALUE
WRITE	0xFF10C0	0x8001
WRITE	0xFF10C2	0x1B02

☐ check that the "BST ready" led is ON
☐ check that the "BEAM ON" led is OFF

☐ check the BST mode register (soft)

REGISTER NAME	OFFSET	EXPECTED VALUE
BST_Machine_Mode	0x7FA9C	0x02

☐ check that the signals are following the "NOBEAM" configuration (all internal) - optional

8.5. Physics machine mode [T22]

- Set the TTCvi to send the "PHYSICS" mode

ACTION	ADDRESS	VALUE
WRITE	0xFF10C0	0x8001
WRITE	0xFF10C2	0x1B03

☐ check that the "BST ready" led is ON
☐ check that the "BEAM ON" led is ON

☐ check the BST mode register (soft)

REGISTER NAME	OFFSET	EXPECTED VALUE
BST_Machine_Mode	0x7FA9C	0x03

☐ check that the signals are following the "BEAM" configuration (all external, BCmain=BCref, Orbmain=Orb1) - optional