



BROOKHAVEN
NATIONAL LABORATORY



L'unam l'université
nantes
angers
le mans
PÔLE DE RECHERCHE ET D'ENSEIGNEMENT SUPÉRIEUR

STAR SSD UPGRADE

Technical implementation

Version 1.6 – February 02nd, 2012
Incomplete DRAFT

Abstract:

The main technical goal of this Upgrade is to increase the event rate capability of the SSD from less than 150Hz to more than 1kHz.

This document describes the Silicon Strip Detector (SSD) electronics after the upgrade and the tools developed for this upgrade. It does not describe the physics goal of this upgrade.

Latest version of this master document and many helpfull files can be found in [0].

1 Contents

1.1 Index

1	Contents	2
1.1	Index	2
1.2	Index of equations	4
1.3	Index of figures	4
1.4	Index of tables	6
2	About this documentation	10
2.1	Typographic conventions	10
2.2	Versions	10
3	Introduction	19
3.1	SSD-Upgrade overview	19
3.2	SSD-Upgrade power supplies	20
3.3	SSD-Upgrade cables	20
3.4	SSD-Upgrade Slow-Control	20
3.5	SSD-Upgrade clocks	21
3.6	SSD-Upgrade readout time	23
4	The upgraded Readout card (rdoU)	24
4.1	rdoU Overview	24
4.2	rdoU Power supply management	24
4.3	rdoU Clock management	24
4.4	rdoU FPGA configuration	25
4.5	rdoU Trigger interface	27
4.6	Overview	27
4.6.1	rdoU Trigger connector	27
4.6.2	rdoU Trigger messages	29
4.6.3	rdoU Trigger Protocol	31
4.7	rdoU DAQ interface (DDL using SIU daughter card)	32
4.7.1	rdoU DAQ connector	32
4.7.2	rdoU DAQ Raw data	32
4.7.3	rdoU SIU daughter card	36
4.8	rdoU Slow-Control interface	38
4.8.1	Overview	38
4.8.2	VME	39
4.9	rdoU-VME-FPGA	40
4.9.1	Overview	40
4.10	rdoU-Master-FPGA	45
4.10.1	Overview	45
4.10.2	rdoU-Master-FPGA TRG part	46
4.10.3	rdoU-Master-FPGA ACQ part	48
4.11	Master<->slave serial Interface (MSSI)	48
4.11.1	overview	48
4.11.2	MSSI rdoU-Master to rdoU-Slave direction	49
4.11.3	MSSI rdoU-Slave to rdoU-Master direction	50
4.12	rdoU-Slave-FPGA	53
4.12.1	Overview	53
4.12.2	EPSC4	54
4.12.3	JTAG	55
4.13	Giga Bit Interface (GBIC)	56
4.13.1	Overview	56

4.13.2	GBIC: RDO-slave to Ladder direction.....	58
4.13.3	GBIC: Ladder to RDO-slave direction.....	58
5	The ladder card.....	60
5.1	Overview.....	60
5.2	Power management.....	60
5.3	Power connector	62
5.4	Grounding	62
5.5	Ladder card switches	63
5.6	Power-up reset (SMT1813 Open Drain, Active-Low with Push- Button Detect).....	63
5.7	Clock management	64
5.8	Ladder readout sequence	64
5.9	Slow-Control.....	65
5.10	analog part of the ladder card.....	69
5.10.1	Overview	69
5.10.2	Connectors for flex cables (30 pins).....	69
5.10.3	Level-shifter (from Alice128 ^{[4][5]} to ADC).....	69
5.10.4	ADC (Analog Devices AD7356 dual 12 bit ADC with serial output) ^[20]	72
5.11	Ladder-FPGA.....	73
5.11.1	Ladder-FPGA configuration.....	73
5.11.2	Ladder-FPGA detection of configuration error due to SEU (Single Event Upset).....	73
5.11.3	Ladder-FPGA event controller.....	74
5.11.4	Ladder-FPGA data packer.....	74
5.12	Debug interface	75
5.12.1	Debug interface: edge connector on ladder card.....	76
5.12.2	Debug interface: FPGA configuration (FPGA-Blaster).....	76
5.12.3	Debug interface: JTAG slow-control.....	76
5.12.4	Debug interface: 8 bit FIFO	76
5.13	Ladder card printed circuit.....	76
5.13.1	Ladder card printed circuit cross-section	77
5.13.2	Ladder card printed circuit board dimensions.....	78
5.13.3	Ladder card printed circuit board placement top (inside, cooled).....	78
5.13.4	Ladder card printed circuit board placement bottom (outside, not cooled)	78
5.13.5	Flex length adaptation.....	79
6	Annexes.....	81
6.1	Pin swapping rules for Altera ^[8] FPGA.....	81
6.1.1	Power pins.....	81
6.1.2	Dedicated pins.....	81
6.1.3	user I/O pins	81
6.2	Simulations.....	81
6.2.1	VHDL simulations.....	81
6.2.2	Level shifter simulation (from Alice128 to ADC)	83
6.3	Complete DAQ raw data.....	85
6.4	SSD before Upgrade.....	90
6.4.1	SSD power supply management before Upgrade.....	90
6.4.2	SSD acquisition and slow-control before Upgrade	91
6.4.3	SSD numbering before Upgrade	91
6.4.4	SSD ADC before Upgrade	93
6.4.5	Hybrid power consumption	95
6.4.6	Alice128C ^{[4][5]} timing.....	95
6.4.7	Alice128 ^{[4][5]} internal shaper output signal.....	96
6.4.8	Test of the readout frequency with old readout card.....	96
6.5	Simulation of JTAG master ACT8990 emulator (slave-FPGA) ^[37]	98

6.5.1	Test 1: Write 2 words to write buffer, separated by 25µs (independent writes)	98
6.5.2	Test 2: Overlapping writes	99
6.6	Tests equipment	100
6.6.1	Mechanical validation	100
6.6.2	Ladder card prototype	105
6.6.3	Debug card	107
6.6.4	Debug isolator (EVAL-ADUM4160EBZ USB isolator card)	110
6.6.5	Static Fake Hybrid card	110
6.6.6	Dynamic Fake Hybrid card	112
6.6.7	Quick RDO (QRDO)	112
6.6.8	FPGADC_star_v1	114
6.7	Tests of the ladder card	116
6.7.1	Tests using debug card	116
6.7.2	Tests using GBIC interface	117
6.8	Planning	118
7	References	120

1.2 Index of equations

Équation 1: <i>level shifter, first stage, negative input</i>	70
Équation 2: <i>level shifter, first stage to second stage</i>	70
Équation 3: <i>level shifter, second stage, negative inputs</i>	70
Équation 4: <i>level shifter, first stage, inputs</i>	70
Équation 5: <i>level shifter, second stage, inputs</i>	70
Équation 6: <i>level shifter, second stage, outputs</i>	70
Équation 7: <i>level shifter, first stage, output</i>	70
Équation 8: <i>level shifter, ADC, negative input ADC_ANALOG_NEG</i>	70
Équation 9: <i>level shifter, ADC, positive input ADC_ANALOG_POS</i>	70
Équation 10: <i>ADC(in+)</i>	70
Équation 11: <i>ADC(in-)</i>	70
Équation 12: <i>Output voltage of DAC with respect to its analog ground vs. register value^[22]</i>	71
Équation 13: <i>Output voltage of DAC-B with respect to analog ground vs. register value</i>	71
Équation 14: <i>Output voltage of DAC-A with respect to analog ground vs. register value</i>	72
Équation 15: <i>ADC internal reference voltage.^[20]</i>	72
Équation 16: <i>ADC resolution when 12 bits are used.^[20]</i>	72
Équation 17: <i>ADC(diff)^[20]</i>	73
Équation 18: <i>ADC(com)^[20]</i>	73
Équation 19: <i>ADC value (12 bits) vs. ADC differential inputs.^[20]</i>	73
Équation 20: <i>ADC(diff)</i>	73
Équation 21: <i>ADC(com)</i>	73
Équation 22: <i>ADC value (12 bits) vs. DAC-A, DAC-B, Vin, VDD_P2V, VSS_M2V and Vref.</i>	73
Équation 23: <i>LSB value in the ADC before upgrade</i>	94

1.3 Index of figures

Figure 1: <i>SSD-Upgrade to do list overview</i>	19
Figure 2: <i>SSD-Upgrade overview</i>	19
Figure 3: <i>SSD-Upgrade power supply chain</i>	20
Figure 4: <i>SSD-Upgrade cables</i>	20
Figure 5: <i>SSD-Upgrade Slow-Control</i>	21
Figure 6: <i>SSD-Upgrade Clock tree</i>	21
Figure 7: <i>SSD-Upgrade Clock chain</i>	22
Figure 8: <i>rdoU trial layout (by BNL October 19, 2010)</i>	24

Figure 9: <i>rdoU FPGA configuration scheme (SUBATECH proposition)</i>	26
Figure 10: <i>rdoU FPGA configuration scheme (BNL proposition)</i>	26
Figure 11: <i>rdoU FPGA configuration scheme for master FPGA</i>	27
Figure 12: <i>rdoU FPGA configuration scheme for slave FPGAs</i>	27
Figure 13: <i>BNL recommendation for Trigger interface</i>	28
Figure 14: <i>rdoU Slow-Control</i>	39
Figure 15: <i>rdoU master FPGA block diagram</i>	46
Figure 16: <i>rdoU master FPGA event state machines</i>	46
Figure 17: <i>rdoU master FPGA MSSI cabling</i>	49
Figure 18: <i>RDO slave-FPGA VME protocol to load data into EPCS4</i>	55
Figure 19: <i>RDO slave-FPGA VME protocol to configure ladder FPGA with data stored in EPCS4</i>	55
Figure 20: <i>DS90UR241^[14] 5-43 MHz DC-Balanced 24-Bit LVDS serializer block diagram</i>	56
Figure 21: <i>DS90UR241^[14] 5-43 MHz DC-Balanced 24-Bit LVDS serializer timing</i>	56
Figure 22: <i>DS90UR241^[14] 5-43 MHz DC-Balanced 24-Bit LVDS serializer pinout (top view)</i>	57
Figure 23: <i>DS90UR124^[14] 5-43 MHz DC-Balanced 24-Bit LVDS deserializer block diagram</i>	57
Figure 24: <i>DS90UR124^[14] 5-43 MHz DC-Balanced 24-Bit LVDS deserializer timing</i>	58
Figure 25: <i>DS90UR124^[14] 5-43 MHz DC-Balanced 24-Bit LVDS deserializer pinout (top view)</i>	58
Figure 26: <i>Nicomatic power connector cabling</i>	62
Figure 27: <i>ground connection for the carbon-epoxy ladder (old ADC card)</i>	63
Figure 28: <i>Ladder card switches</i>	63
Figure 29: <i>schematics of the latch up detection in the ladder card^{[17][18][19]}</i>	69
Figure 30: <i>schematics of the analog part in the ladder card^{[20][21][22]}</i>	69
Figure 31: <i>Diagram of operation for level-shifter DAC serial interface.^[22]</i>	71
Figure 32: <i>Control codes for operation of level-shifter DAC serial interface.^[22]</i>	71
Figure 33: <i>Diagram of operation for serial ADC.^[20]</i>	72
Figure 34: <i>Timing of operation for serial ADC.^[20]</i>	72
Figure 35: <i>ladder-FPGA event controller state machine</i>	74
Figure 36: <i>ladder-FPGA data packer</i>	75
Figure 37: <i>ladder card preproduction dimensions (in mm)</i>	78
Figure 38: <i>ladder card preproduction circuit top</i>	78
Figure 39: <i>ladder card preproduction circuit bottom</i>	78
Figure 40: <i>component arrangement on the ladder (old SSD)</i>	79
Figure 41: <i>VHDL simulation of a complete ladder side (acquisition of ladder-FPGA + 16 hybrids) using ModelSim-Altera^[12] 6.4a</i>	82
Figure 42: <i>VHDL simulation of a complete ladder side (slow-control of ladder-FPGA + 16 hybrids) using ModelSim-Altera^[12] 6.4a</i>	82
Figure 43: <i>VHDL simulation of a readout card connected to complete ladder sides (L0+L2a) using ModelSim-Altera^[12] starter edition 6.6d</i>	83
Figure 44: <i>Level shifter (first simulation schematics using Tina)</i>	83
Figure 45: <i>Level shifter (first simulation result using Tina)</i>	84
Figure 46: <i>Level shifter (second simulation schematics using Cadence)</i>	84
Figure 47: <i>Level shifter (second simulation output using Cadence PSPICE)</i>	85
Figure 48: <i>Level shifter (second simulation output using Cadence PSPICE)</i>	85
Figure 49: <i>SSD Power supply chain before Upgrade</i>	90
Figure 50: <i>Power supply and sense pigtails (AWG24) and connectors for ADC card</i>	91
Figure 51: <i>SSD acquisition and slow-control diagram before Upgrade</i>	91
Figure 52: <i>SSD numbering at SSD level before Upgrade^[31]</i>	92
Figure 53: <i>SSD numbering at ladder level before Upgrade^[31]</i>	92
Figure 54: <i>SSD numbering at module level before Upgrade^[31]</i>	93
Figure 55: <i>SSD numbering at Wafer level before Upgrade^[31]</i>	93
Figure 56: <i>SSD ADC mode of operation before Upgrade^[32]</i>	94
Figure 57: <i>AliceI28C timing^[5]</i>	95

Figure 58: Alice128 internal shaper output signal (calculated by Stéphane Bouvier)	96
Figure 59: VHDL simulation (1xrdo+10xadc+10xc2d2+10x16xhybrids) of a hybrid readout frequency test using ModelSim-Altera ^[12] 6.4a	97
Figure 60: Superposition of test signals, for different readout speeds, recorded with an oscilloscope. ^[36]	98
Figure 61: First word = x"ABCD". Value is loaded into write shift register, which is the source for the outgoing TDIO. Note that changes occur on falling edge of TCK, as required. TDIO is looped back to TDO (input to read shift register). At the end of the process, x"ABCD" is loaded into READ BUFFER, where it can be read by the host.....	98
Figure 62: Second word (x"EF01") is loaded into WRITE BUFFER. It is shifted as in Fig.1, and reappears in the READ BUFFER at the end of the process.....	99
Figure 63: At t=210µs, the first word appears at the output of the WRITE BUFFER (FIFO). The word is loaded into the write shift register at t=212 µs. At 222µs, the second word appears at the FIFO output. At t=238µs, write_valid_bits = 0, and load_write_xfer is asserted. The new value (EF01) appears in the write shift register. At the same time, the first word (ABCD) appears reassembled at the output of the READ shift register.....	99
Figure 64: These waveforms cover the entire duration of the shifting of both words. Note that TAP state = Shift-IR from 216µs to 263µs, without any pauses while the second word is being loaded into the WRITE shift register.....	99
Figure 65: Two successive commands: read READ BUFFER return the words ABCD, EF01 written at the beginning of this sequence. The valid bit is returned to the host on each read command.	100
Figure 66: mechanical prototype of ladder card in place on a ladder (CAD drawing by Gérard Guilloux)....	100
Figure 67: optical transceiver CAD model for mechanical prototype (CAD drawing by Gérard Guilloux)	101
Figure 68: Mechanical prototype of ladder card	101
Figure 69: Mechanical prototype of ladder card fixed on a real ladder.....	102
Figure 70: ladder card mechanical prototype dimensions (in mm)	102
Figure 71: Mechanical test of prototype ladder card fixed on a fake ladder end and equipped with a debug card	103
Figure 72: fake ladder (CAD drawing by Gérard Guilloux).....	103
Figure 73: mechanical CAD drawing of ladders on the support, with the shroud on top. NO SCALE INDICATED!!!	104
Figure 74: ladder card mechanical prototype with 45° rotated optical transceiver.....	104
Figure 75: ladder card prototype with 45° rotated optical transceiver	104
Figure 76: ladder card preprod with 30° rotated optical transceiver	104
Figure 77: ladder card prototype dimensions (in mm).....	105
Figure 78: ladder card prototype circuit top.....	106
Figure 79: ladder card prototype circuit bottom.....	107
Figure 80: STAR-SSD slow-control cable from PC parallel DB25 to STAR-SSD JTAG HE10	108
Figure 81: debug card V2 equipped with UM245R daughter card.....	109
Figure 82: debug card V3.....	110
Figure 83: EVAL-ADUM4160EBZ USB isolator card.....	110
Figure 84: Static Fake Hybrid (SF-Hyb) card schematics proposed by Subatech.	111
Figure 85: Static Fake Hybrid (SF-Hyb) card schematics implemented by BNL.	111
Figure 86: Static Fake Hybrid (SF-Hyb) card implemented by BNL.	112
Figure 87: Dynamic Fake Hybrid (DF-Hyb) card schematics.	112
Figure 88: QRDO placement.	113
Figure 89: QRDO partially assembled.	113
Figure 90: FPGADC_V1 pre-placement.....	114
Figure 91: FPGADC_V1 level shifter using AD5449 ^[28]	116

1.4 Index of tables

Table 1: Typographic conventions used in tables and figures	10
---	----

Table 2: <i>trigger connector</i>	28
Table 3: <i>trigger message</i>	29
Table 4: <i>trigger message commands and associated action</i>	29
Table 5: <i>trigger message token</i>	30
Table 6: <i>trigger protocol</i>	31
Table 7: <i>Raw Data organization</i>	32
Table 8: <i>Raw Data organization RDO master header</i>	32
Table 9: <i>Raw Data organization RDO master total words</i>	32
Table 10: <i>Raw Data organization RDO master ID</i>	33
Table 11: <i>Raw Data organization RDO trigger message</i>	33
Table 12: <i>Raw Data organization RDO-slave block</i>	33
Table 13: <i>Raw Data organization RDO-slave header</i>	33
Table 14: <i>Raw Data organization RDO-slave total words</i>	34
Table 15: <i>Raw Data organization RDO-Slave mode</i>	34
Table 16: <i>Raw Data organization ladder status</i>	34
Table 17: <i>Raw Data organization hybrid total words</i>	35
Table 18: <i>Raw Data organization hybrid block ADC data</i>	35
Table 19: <i>Raw Data organization hybrid block pedestal memory</i>	35
Table 20: <i>Raw Data organization hybrid block pedestal subtracted</i>	35
Table 21: <i>Raw Data organization hybrid block zero suppressed</i>	35
Table 22: <i>Raw Data organization RDO-slave block when slave is not used</i>	36
Table 23: <i>SIU event transaction</i>	37
Table 24: <i>SIU write pedestal transaction</i>	37
Table 25: <i>SIU control words (commands from SIU to RDO)</i>	38
Table 26: <i>SIU control words (parameter field in STBWR command)</i>	38
Table 27: <i>SIU data word (set pedestal value)</i>	38
Table 28: <i>SIU control words (status from RDO to SIU)</i>	38
Table 29: <i>rdoU VME addresses</i>	39
Table 30: <i>rdoU VME registers</i>	40
Table 31: <i>rdoU VME-FPGA VME registers</i>	41
Table 32: <i>ladder-FPGA slow-control register (roboclock_phase)^[1] effect on Clock36MHz-slv4 and clock36MHz-slv3</i>	41
Table 33: <i>ladder-FPGA slow-control register (roboclock_phase)^[1] effect on Clock36MHz-slv2 and clock36MHz-slv1</i>	41
Table 34: <i>ladder-FPGA slow-control register (roboclock_phase)^[1] effect on feed-back and clock36MHz-slv041</i>	41
Table 35: <i>ladder-FPGA slow-control register (roboclock_phase)^[1] effect on Clock36MHz-vme FPGA</i>	41
Table 36: <i>rdoU VME-FPGA pinout</i>	45
Table 37: <i>rdoU master-FPGA TRG part VME registers</i>	47
Table 38: <i>rdoU master-FPGA ACQ part VME registers</i>	48
Table 39: <i>RDO-Master to RDO-Slave interface steering bits and associated action and payload</i>	49
Table 40: <i>RDO-Master to RDO-Slave interface RDO-Slave mode</i>	50
Table 41: <i>RDO-Slave to RDO-Master interface</i>	50
Table 42: <i>RDO-Slave to RDO-Master interface steering bits and associated payload</i>	50
Table 43: <i>RDO-Slave to RDO-Master interface payload</i>	50
Table 44: <i>RDO-Slave to RDO-Master interface payload “RDO-slave header”</i>	51
Table 45: <i>RDO-Slave to RDO-Master interface payload “RDO-slave total words”</i>	51
Table 46: <i>rdoU-Slave to rdoU-Master interface payload “ladder status”</i>	51
Table 47: <i>rdoU-Slave to rdoU-Master interface payload “hybrid total words”</i>	52
Table 48: <i>rdoU-Slave to rdoU-Master interface payload “hybrid block zero suppressed (mode=0)”</i>	52
Table 49: <i>rdoU-Slave to rdoU-Master interface payload “hybrid block pedestal subtracted (mode=1)”</i>	52
Table 50: <i>rdoU-Slave to rdoU-Master interface payload “hybrid block raw ADC data (mode=2)”</i>	52
Table 51: <i>rdoU-Slave to rdoU-Master interface payload “hybrid block pedestal memory (mode=3)”</i>	53

Table 52: <i>rdoU slave-FPGA VME registers</i>	54
Table 53: <i>rdoU slave-FPGA VME FPGA_config_cmd register cmd bits and associated commands</i>	54
Table 54: <i>GBIC: RDO-slave to ladder interface</i>	58
Table 55: <i>GBIC: ladder to RDO-slave interface</i>	58
Table 56: <i>GBIC: ladder to RDO-slave interface payload "acquisition"</i>	59
Table 57: <i>GBIC: ladder to RDO-slave interface payload "status"</i>	59
Table 58: <i>ladder card typical power requirement (normal operation)</i> ^{[9][14][15][16][17][18][19][20][21][22]}	60
Table 59: <i>ladder card maximum power requirement (normal operation)</i> ^{[9][14][15][16][17][18][19][20][21][22]}	61
Table 60: <i>ladder card typical power requirement (debug operation without hybrid, without RDO)</i> ^{[9][14][15][16][17][18][19][20][21][22][23][24]}	61
Table 61: <i>ladder-end current requirement for low voltage power supplies</i>	62
Table 62: <i>ladder current requirement for bias power supply</i>	62
Table 63: <i>Alice128^{[4][5]} readout order</i>	64
Table 64: <i>ladder-FPGA slow-control registers</i>	65
Table 65: <i>ladder-FPGA slow-control register (roboclock_phase)^[1]</i>	65
Table 66: <i>ladder-FPGA slow-control register (roboclock_phase)^[1] effect on clock40MHz_fpga</i>	65
Table 67: <i>ladder-FPGA slow-control register (roboclock_phase)^[1] effect on clock40MHz_ser</i>	66
Table 68: <i>ladder-FPGA slow-control register (roboclock_phase)^[1] effect on clock80MHz-adc0 and clock80MHz-adc1</i>	66
Table 69: <i>ladder-FPGA slow-control register (roboclock_phase)^[1] effect on clock80MHz-adc2 and clock80MHz-adc3</i>	66
Table 70: <i>ladder-FPGA slow-control register (roboclock_phase)^[1] effect on clock80MHz-adc4 and clock80MHz-adc5</i>	66
Table 71: <i>ladder-FPGA slow-control register (roboclock_phase)^[1] effect on clock80MHz-adc6 and clock80MHz-adc7</i>	66
Table 72: <i>ladder-FPGA slow-control register (status)</i>	66
Table 73: <i>ladder-FPGA slow-control register (config)</i>	67
Table 74: <i>ladder-FPGA slow-control register (level-shifter DAC values)</i>	67
Table 75: <i>ladder-FPGA slow-control register (etat alim)</i>	67
Table 76: <i>ladder-FPGA slow-control register (mux ref latchup)</i>	67
Table 77: <i>ladder-FPGA slow-control register (rallumage)</i>	67
Table 78: <i>ladder-FPGA slow-control register (bypass)</i>	67
Table 79: <i>ladder-FPGA slow-control register (version)</i>	67
Table 80: <i>ladder-FPGA slow-control register (temperature)</i>	67
Table 81: <i>ladder-FPGA slow-control register (identity)</i>	68
Table 82: <i>ladder-FPGA slow-control register (bypass)</i>	68
Table 83: <i>JTAG slow-control chain when "jtag chain with selected hybrid" is asserted</i>	68
Table 84: <i>Alice128^{[4][5]} slow-control registers (identite)</i>	68
Table 85: <i>Costar^[27] slow-control registers (identite)</i>	68
Table 86: <i>JTAG transaction timing</i>	68
Table 87: <i>ladder-FPGA data packer flux compactor</i>	75
Table 88: <i>ladder-FPGA data packer FIFO21 data organization</i>	75
Table 89: <i>Edge connector to debug interface</i>	76
Table 90: <i>ladder card cross section</i>	77
Table 91: <i>flex cable length difference and related resistive difference</i>	79
Table 92: <i>Flex length difference adaptation for +2V (VDD) and -2V (VSS) on ladder card prototype printed circuit</i>	80
Table 93: <i>complete raw data</i>	90
Table 94: <i>currents needed by the Front-End Electronics (FEE)</i>	95
Table 95: <i>number of ladders read out and equivalent readout speed for the hybrids</i>	97
Table 96: <i>FPGA-Blaster connector</i>	108
Table 97: <i>JTAG slow-control connector</i>	108

Table 98: <i>debug interface connector</i>	109
Table 99: <i>Edge connector to debug interface</i>	115
Table 100: <i>SCSI68 connector to Front-End interface</i>	115

INCOMPLETE DRAFT

2 About this documentation

2.1 Typographic conventions

This document uses the typographic conventions shown below.

	<i>meaning</i>
<i>Black (r=0;g=0;b=0) italic text</i>	Table header
Blue (r=0;g=0;b=255) text	comment
Light blue (r=51;g=102;b=255) background	Data from VME processor
Medium blue (r=153;g=204;b=255) background	Data from central DAQ system
Aquatic green (r=51;g=204;b=204) background	Data from central trigger system
Turquoise (r=51;g=204;b=204) background	Data from USB debug system
Grey 20% (r=204;g=204;b=204) background	Data from VME-P2 backplane system
Pink (r=255;g=0;b=255) background	Data from RDO-master FPGA
(r=255;g=128;b=128) background	Data from ladder FPGA
Orange (r=255;g=102;b=0) background	Data from RDO-VME FPGA
Yellow (r=255;g=255;b=0) background	Data from RDO-slave FPGA
Red (r=255;g=0;b=0) background	Data from hybrid
Brown (r=153;g=51;b=0) background	Data from RDO printed circuit
Light orange (r=255;g=153;b=0) background	Data from ladder card printed circuit
Green (r=0;g=255;b=0) text	Free unassigned bits. Optional name indicates the FPGA that could assign the bits

Table 1: Typographic conventions used in tables and figures

the icon "⚠" indicates a warning

2.2 Versions

Version 0.1: March 31st 2009 - August 18th, 2009

Version 0.2: August 21st, 2009

- Changed order of status words in **Table 43: GBIC ladder to RDO-slave interface payload “status”**
- Included slow-control registers description in **Table 46: ladder-FPGA slow-control registers** and **Table 67: Costar slow-control registers (identite)**
- Included information from Mike about strip readout order in § 5.4

Version 0.3: August 24th, 2009 – August 29th 2009

- Included VHDL simulation output in **Figure 16: VHDL simulation of a complete ladder side (acquisition of a ladder-FPGA + 16 hybrids) using ModelSim-Altera 6.4a** and **Figure 17: VHDL simulation of a complete ladder side (slow-control of a ladder-FPGA + 16 hybrids) using ModelSim-Altera 6.4a**
- Included new Level-shifter simulations in **Figure 20: Level shifter (second simulation schematics using Cadence)**, **Figure 21: Level shifter (second simulation output using Cadence PSPICE)** and **Figure 22: Level shifter (second simulation output using Cadence PSPICE)**
- Modified status words in **Table 43: GBIC ladder to RDO-slave interface payload “status”**, **Table 16: Raw Data organization ladder status**, **Table 33: RDO-slave to RDO-master interface payload “ladder status”** and **Table 66: complete raw data**
- Updated **Figure 3: SSD-Upgrade power supply chain** and **Figure 23: SSD power supply chain before Upgrade**

- Added details about ladder_fpga_sc_roboclock_phase slow-control register in Table 47: ladder-FPGA slow-control register (roboclock_phase), Table 48: ladder-FPGA slow-control register (roboclock_phase) effect on clock40MHz_fpga, Table 49: ladder-FPGA slow-control register (roboclock_phase) effect on clock40MHz_ser, Table 50: ladder-FPGA slow-control register (roboclock_phase) effect on clock80MHz_adc0 and clock80MHz_adc1, Table 51: ladder-FPGA slow-control register (roboclock_phase) effect on clock80MHz_adc2 and clock80MHz_adc3, Table 52: ladder-FPGA slow-control register (roboclock_phase) effect on clock80MHz_adc4 and clock80MHz_adc5 and Table 53: ladder-FPGA slow-control register (roboclock_phase) effect on clock80MHz_adc6 and clock80MHz_adc7
- Changed TEST signal protocol in Figure 11: ladder-FPGA event controller state machine and Table 6: trigger protocol

Version 0.4: September 2nd, 2009 – September 11th, 2009

- Added power consumption on the ladder card in Table 44: ladder card power requirement
- Added temperature sensor position on the ladder card in Table 61: ladder-FPGA slow-control register (temperature)
- Added level-shifter schematics in Figure 10: level shifter schematics
- Modified Busy_Back in Table 2: trigger connector
- Added new BNL recommendations in Figure 13: BNL recommendation for Trigger interface
- Modified number of free bits in Table 16: Raw Data organization ladder status
- Changed two MSB from “master free” to slave free” in Table 14: Raw Data organization RDO-slave total words, Table 16: Raw Data organization ladder total words, Table 16: Raw Data organization ladder status, Table 17: Raw Data organization hybrid total words, Table 18: Raw Data organization hybrid block ADC data, Table 19: Raw Data organization hybrid block pedestal memory, Table 20: Raw Data organization hybrid block pedestal subtracted and Table 21: Raw Data organization hybrid block zero suppressed
- Added information in Table 1: Typographic conventions used in tables
- Moved steering bits from bits 30-31 to bits 32-33 in Table 28: RDO-slave to RDO-master interface, Table 31: RDO-slave to RDO-master interface payload “RDO-slave header”, Table 32: RDO-slave to RDO-master interface payload “RDO-slave total words”, Table 33: RDO-Slave to RDO-Master interface payload “ladder total words”, Table 34: RDO-slave to RDO-master interface payload “ladder status”, Table 35: RDO-slave to RDO-master interface payload “hybrid total words”, Table 38: RDO-slave to RDO-master interface payload “hybrid block raw ADC data (mode 2)”, Table 39: RDO-slave to RDO-master interface payload “hybrid block pedestal memory (mode 3)”, Table 37: RDO-slave to RDO-master interface payload “hybrid block pedestal subtracted (mode 1)” and Table 36: RDO-slave to RDO-master interface payload “hybrid block zero suppressed (mode 0)”
- Modified number of words for ladder status and adjusted the position in Table 12: Raw Data organization RDO-slave block, Table 13: Raw Data organization RDO-slave header, Table 30: RDO-slave to RDO-master interface payload and Table 31: RDO-slave to RDO-master interface payload “RDO-slave header”
- Added RDO-Slave mode description in Table 15: Raw Data organization RDO-Slave mode and Table 27: RDO-master to RDO-slave interface RDO-slave mode
- Added ladder card dimensions in Figure 13: ladder card dimensions (in mm)

Version 0.5: September 12th, 2009 – September 16th, 2009

- Updated information in Table 9: Raw Data organization RDO master total words, Table 12: Raw Data organization RDO-slave block, Table 13: Raw Data organization RDO-slave header, Table 14: Raw Data organization RDO-slave total words, Table 17: Raw Data organization hybrid total words, Table 34: RDO-slave to RDO-master interface payload “hybrid total words”, Table 31: RDO-slave to RDO-master interface payload “RDO-slave header”, Table 34: RDO-Slave to RDO-Master interface payload “Ladder total words” and Table 32: RDO-slave to RDO-master interface payload “RDO-slave total words”

- Suppressed **Table16: Raw Data organization Ladder total words** and **Table34: RDO-Slave to RDO-Master interface payload “Ladder total words”**
- Added ladder card cross-section in **Table 65: ladder card cross section**
- Added ladder card placement in **Figure 14: ladder card placement top** and **Figure 15: ladder card placement bottom**

Version 0.6: September 16th, 2009 – September 25th, 2009

- updated **Table 10: Raw Data organization RDO master ID**
- Steering bits outside of payload in **Table 41: RDO-Slave to RDO-Master interface**
- merged **Table 39: GBIC RDO-slave to ladder interface** and **Table 40: GBIC RDO-slave to ladder interface payload**
- specified clock for HOLD delay in **Table 39: RDO-Master to RDO-Slave interface steering bits and associated action and payload**
- updated and added a comment in **Table 6: trigger protocol[]**
- updated **Table 26: RDO-master to RDO-slave interface steering bits and associated action and payload**
- suppressed RDO # in **Table 21: Raw Data organization hybrid block zero suppressed** and **Table 48: rdoU-Slave to rdoU-Master interface payload “hybrid block zero suppressed (mode=0)”**
- updated **Table 25: SIU control words (commands from SIU to RDO)**
- added **Table 26: SIU control words (parameter field in STBWR command)**

Version 0.7: September 29th, 2009 – October 16th, 2009

- updated **Figure 10: level-shifter schematics**[**Figure 30: schematics of the analog part in the ladder card**]
- added **Table 69: currents needed by the Front-End Electronics (FEE)**[**Table 94: currents needed by the Front-End Electronics (FEE)**]
- updated **Figure 1: SSD-Upgrade to do list overview**[**Figure 1: SSD-Upgrade to do list overview**]
- added **Figure 16: Ladder card on a ladder (CAD drawing by Gérard Guilloux)**[: **optical transceiver CAD model for mechanical prototype (CAD drawing by Gérard Guilloux)**]
- added **Figure 26: component arrangement on the ladder**[**Figure 40: component arrangement on the ladder**] and **Table 70: flex cable length difference and related resistive difference**[**Table 91: flex cable length difference and related resistive difference**]
- updated **Figure 4: SSD-Upgrade cables**[**Figure 4: SSD-Upgrade cables**]
- added **Table 66: flex length difference adaptation for +2V (VDD) and -2V (VSS)**[**Table 92: Flex length difference adaptation for +2V (VDD) and -2V (VSS)**]

Version 0.8: October 16th, 2009 – February 09th, 2010

- updated **Table 14: Raw Data organization RDO-slave total words**[**Table 14: Raw Data organization RDO-slave total words**]
- added **Figure 27: Alice128 internal shaper output signal**[**Figure 58: Alice128 internal shaper output signal**]
- updated **Table 65: ladder card cross section**[**Table 90: ladder card cross section**]
- updated **Figure 10: level-shifter schematics**[**Figure 30: schematics of the analog part in the ladder card**]
- updated **Table 44: ladder card power requirement**[**Table 58: ladder card typical power requirement**]
- updated **Table 55: ladder-FPGA slow-control register (config)**[**Table 73: ladder-FPGA slow-control register (config)**]
- added **Table 56: ladder-FPGA slow-control register (level-shifter DAC values)**[**Table 74: ladder-FPGA slow-control register (level-shifter DAC values)**]
- updated **Table 46: ladder-FPGA slow-control registers**[**Table 64: ladder-FPGA slow-control registers**]
- updated **Figure 13: ladder card dimensions (in mm)**[**Figure 37: ladder card preproduction dimensions (in mm)**]
- updated **Figure 14: ladder card placement top**[**Figure 38: ladder card preproduction circuit top**]
- updated **Figure 15: ladder card placement bottom**[**Figure 39: ladder card preproduction circuit bottom**]

- added 6.7: Test of the readout frequency with old readout card[6.4.8 Test of the readout frequency with old readout card], Table 73: number of ladders readout and equivalent readout speed for the hybrids[Table 95: *number of ladders read out and equivalent readout speed for the hybrids*], and Figure 28: VHDL simulation (rdo+10adc+10c2d2+10x16hybrid) of a hybrid readout frequency test using ModelSim-Altera 6.4a[*VHDL simulation (1xrdo+10xadc+10xc2d2+10x16xhybrids) of a hybrid readout frequency test using ModelSim-Altera[12] 6.4a*]
- updated Table 43: GBIC ladder to RDO-slave interface payload “status”[Table 57: *GBIC: ladder to RDO-slave interface payload “status”*] and Table 54: ladder-FPGA slow-control register (status)[Table 72: *ladder-FPGA slow-control register (status)*]
- updated Figure 1: SSD-Upgrade to do list overview[Figure 1: *SSD-Upgrade to do list overview*]
- added Figure 10: Ground connection for the carbon-epoxy ladder[Figure 27: *ground connection for the carbon-epoxy ladder*]

Version 0.9: February 09th, 2010 – March 16th, 2010

- updated Table 68: complete raw data[Table 93: *complete raw data*]
- added Table 27: RDO VME addresses[Table 29: *rdoU VME addresses*]
- updated Table 56: ladder-FPGA slow-control register (config)[Table 73: *ladder-FPGA slow-control register (config)*]
- moved 5.6: Ladder-FPGA configuration[] to 5.10.1: Ladder-FPGA configuration[5.11.1 Ladder-FPGA configuration]
- added 5.10.2: Ladder-FPGA detection of configuration error due to SEU (Single Event Upset)[5.11.2 Ladder-FPGA detection of configuration error due to SEU (Single Event Upset)]
- updated Table 55: ladder-FPGA slow-control register (status)[Table 72: *ladder-FPGA slow-control register (status)*]
- added Figure 30: Superposition of test signals, for different readout speeds, recorded with an oscilloscope[Figure 60: *Superposition of test signals, for different readout speeds, recorded with an oscilloscope.*]

Version 1.0: March 17th, 2010 – July 2nd, 2010

- updated Table 27: RDO VME addresses[Table 29: *rdoU VME addresses*]
- added Figure 26: Power supply and sense pigtailed (AWG24) and connectors for ADC card[Figure 50: *Power supply and sense pigtailed (AWG24) and connectors for ADC card*]
- updated Table 6: trigger protocol[Table 6: *trigger protocol*]
- updated Table 44: GBIC ladder to RDO-slave interface payload[Table 57: *GBIC: ladder to RDO-slave interface payload “status”*]
- updated Table 59: ladder-FPGA slow-control register (status)[Table 72: *ladder-FPGA slow-control register (status)*]
- updated Figure 31: Superposition of test signals, for different readout speeds, recorded with an oscilloscope [Figure 60: *Superposition of test signals, for different readout speeds, recorded with an oscilloscope.*]
- updated Figure 14: ladder card dimensions (in mm)[Figure 37: *ladder card preproduction dimensions (in mm)*]
- updated Figure 15: ladder card placement top[Figure 38: *ladder card preproduction circuit top*]
- updated Figure 16: ladder card placement bottom[Figure 39: *ladder card preproduction circuit bottom*]

Version 1.1: July 3rd, 2010 – November 18th, 2010

- updated Table 41: GBIC: RDO-slave to ladder interface[Table 54: *GBIC: RDO-slave to ladder interface*] and Table 42: GBIC: ladder to RDO-slave interface[Table 55: *GBIC: ladder to RDO-slave interface*]
- added Table 67: Edge connector to debug interface[Table 89: *Edge connector to debug interface*]
- updated Table 1: Typographic conventions used in tables[Table 1: *Typographic conventions used in tables*]
- added Figure 8: rdoU FPGA configuration scheme[Figure 9: *rdoU FPGA configuration scheme (SUBATECH proposition)*]

- updated Figure 15: ladder card dimensions (in mm)[Figure 37: *ladder card preproduction dimensions (in mm)*], Figure 16: ladder card placement top[Figure 38: *ladder card preproduction circuit top*] and Figure 17: ladder card placement bottom[Figure 39: *ladder card preproduction circuit bottom*]
- added Figure 8: rdoU trial layout (by BNL October 12, 2010)[Figure 8: *rdoU trial layout (by BNL October 19, 2010)*]
- added Figure 10: rdoU configuration scheme (BNL proposition)[Figure 10: *rdoU FPGA configuration scheme (BNL proposition)*]
- updated Table 4: trigger message commands and associated action[Table 4: *trigger message commands and associated action*]
- updated Table 10: Raw Data organization RDO master ID[Table 10: *Raw Data organization RDO master ID*]
- added Table 28: RDO VME-FPGA VME registers[Table 31: *rdoU VME-FPGA VME registers*], Table 29: RDO master-FPGA TRG part VME registers [Table 37: *rdoU master-FPGA TRG part VME registers*], Table 30: RDO master-FPGA ACQ part VME registers [Table 38: *rdoU master-FPGA ACQ part VME registers*] and Table 44: RDO slave-FPGA VME registers [Table 52: *rdoU slave-FPGA VME registers*]
- updated Table 27: RDO VME addresses[Table 29: *rdoU VME addresses*]
- added Figure 22: Mechanical prototype of ladder card[Figure 68: *Mechanical prototype of ladder card*], Figure 23: Mechanical prototype of ladder card on a real ladder[Figure 69: *Mechanical prototype of ladder card fixed on a real ladder*] and Figure 24: fake ladder (CAD drawing by Gérard Guilloux)[Figure 72: *fake ladder (CAD drawing by Gérard Guilloux)*]

Version 1.2: November 18th, 2010 – March 21th, 2011

- updated Table 44: RDO slave-FPGA VME registers[Table 52: *rdoU slave-FPGA VME registers*]
- added Table 45: RDO slave-FPGA VME FPGA_config_cmd register cmd[Table 53: *rdoU slave-FPGA VME FPGA_config_cmd register cmd*]
- added Figure 13: RDO slave-FPGA VME protocol to load data into EPCS4[Figure 18: *RDO slave-FPGA VME protocol to load data into EPCS4*]
- fractioned Table 27: RDO VME addresses[Table 29: *rdoU VME addresses*] into Table 27: RDO VME addresses[Table 29: *rdoU VME addresses*] and Table 28: RDO VME registers[Table 30: *rdoU VME registers*]
- updated Table 30: RDO master-FPGA TRG part VME registers[Table 37: *rdoU master-FPGA TRG part VME registers*] and attached comments
- updated Table 10: Raw Data organization RDO master ID [Table 10: *Raw Data organization RDO master ID*], Table 30: RDO master-FPGA TRG part VME registers [Table 37: *rdoU master-FPGA TRG part VME registers*] and Table 31: RDO master-FPGA ACQ part VME registers [Table 38: *rdoU master-FPGA ACQ part VME registers*]
- added 6.7 Simulation of JTAG emulator (slave FPGA)[6.5 Simulation of JTAG master ACT8990 emulator (slave-FPGA)]
- updated Figure 19: ladder-card circuit top [Figure 38: *ladder card preproduction circuit top*] and Figure 20: ladder-card circuit bottom [Figure 39: *ladder card preproduction circuit bottom*]
- added Figure 25: Mechanical test of pre-serial ladder card fixed on a fake ladder end[Figure 71: *Mechanical test of prototype ladder card fixed on a fake ladder end*]
- added Table 51: ladder-card typical power requirement[Table 58: *ladder card typical power requirement*]
- updated Table 52: ladder-card maximum power requirement[Table 59: *ladder card maximum power requirement*]
- updated Table 29: RDO VME-FPGA VME registers[Table 31: *rdoU VME-FPGA VME registers*]
- added Figure 14: Nicomatic power connector cabling[Figure 26: *Nicomatic power connector cabling*], Figure 19: debug card V2 equipped with UM254R daughter card[Figure 81: *debug card V2 equipped with UM245R daughter card*] and Figure 20: debug card V3[Figure 82: *debug card V3*]
- added Table 30: rdoU VME-FPGA pinout[Table 36: *rdoU VME-FPGA pinout*]

- updated Figure 6: SSD-Upgrade clock tree [Figure 6: *SSD-Upgrade Clock tree*] and Figure 7: SSD-Upgrade Clock chain [Figure 7: *SSD-Upgrade Clock chain*]
- Version 1.3: March 21th, 2011 – June 27th, 2011
- updated Table 31: rdoU master-FPGA TRG part VME registers [Table 37: *rdoU master-FPGA TRG part VME registers*]
- added Figure 13: rdoU master-FPGA block diagram [Figure 15: *rdoU master FPGA block diagram*] and Figure 14: rdoU master-FPGA event state machines [Figure 16: *rdoU master FPGA event state machines*]
- added Figure 50: QRDO placement [Figure 88: *QRDO placement.*]
- added Table 26: SIU data word (set pedestal value) [Table 27: *SIU data word (set pedestal value)*]
- updated Table 10: Raw Data organization RDO master ID [Table 10: *Raw Data organization RDO master ID*]
- added 6.9: Test of the ladder card [6.7 Tests of the ladder card]
- added Table 74: JTAG slow-control chain when "jtag chain with selected hybrid" is asserted [Table 83: *JTAG slow-control chain when "jtag chain with selected hybrid" is asserted*]
- updated Table 36: RDO-Slave to RDO-Master interface [Table 41: *RDO-Slave to RDO-Master interface*]
- added Figure 16: RDO slave-FPGA VME protocol to configure ladder-FPGA with data stored in EPCS4 [Figure 19: *RDO slave-FPGA VME protocol to configure ladder FPGA with data stored in EPCS4*]
- Version 1.4: June 27th, 2011 – July 01st, 2011
- upgraded Figure 1: SSD-Upgrade to do list overview [Figure 1: *SSD-Upgrade to do list overview*]
- upgraded Figure 3: SSD-Upgrade power supply chain [Figure 3: *SSD-Upgrade power supply chain*]
- upgraded Figure 4: SSD-Upgrade cables [Figure 4: *SSD-Upgrade cables*]
- added Figure 22: Diagram of operation for level-shifter DAC serial interface [Figure 31: *Diagram of operation for level-shifter DAC serial interface.*] and Figure 23: Control codes for operation of level-shifter DAC serial interface [Figure 32: *Control codes for operation of level-shifter DAC serial interface.*]
- added Figure 24: Diagram of operation for serial ADC [Figure 33: *Diagram of operation for serial ADC.*]
- Version 1.5: July 01st, 2011 – August 23rd, 2011
- added Equation 1: level shifter, first stage, negative input [Équation 1: *level shifter, first stage, negative input*], Equation 2: level shifter, first stage to second stage [Équation 2: *level shifter, first stage to second stage*], Equation 3: level shifter, second stage, negative inputs [Équation 3: *level shifter, second stage, negative inputs*], Equation 4: level shifter, first stage, inputs [Équation 4: *level shifter, first stage, inputs*], Equation 5: level shifter, second stage, inputs [Équation 5: *level shifter, second stage, inputs*], Equation 6: level shifter, second stage, outputs [Équation 6: *level shifter, second stage, outputs*], Equation 7: level shifter, first stage, output [Équation 7: *level shifter, first stage, output*], Equation 8: level shifter, ADC, negative input ADC_ANALOG_NEG [Équation 8: *level shifter, ADC, negative input ADC_ANALOG_NEG*], Equation 9: level shifter, ADC, positive input ADC_ANALOG_POS [Équation 9: *level shifter, ADC, positive input ADC_ANALOG_POS*], Equation 10: ADC(in+) [Équation 10: *ADC(in+)*] and Equation 11: ADC(in-) [Équation 11: *ADC(in-)*]
- added §1.2 Index of equations [§1.2 Index of equations]
- added Equation 12: Output voltage of DAC with respect to analog ground vs. register value^[22] [Équation 12: *Output voltage of DAC with respect to its analog ground vs. register value^[22]*], Equation 13: Output voltage of DAC-B with respect to analog ground vs. register value [Équation 13: *Output voltage of DAC-B with respect to analog ground vs. register value*] and Equation 14: Output voltage of DAC-A with respect to analog ground vs. register value [Équation 14: *Output voltage of DAC-A with respect to analog ground vs. register value*]
- added Figure 24: Timing of operation for serial ADC^[20] [Figure 34: *Timing of operation for serial ADC.*^[20]]

- added Equation 15: ADC internal reference voltage^[20] [Équation 15: ADC internal reference voltage.^[20]], Equation 16: ADC resolution when 12 bits used^[20] [Équation 16: ADC resolution when 12 bits are used.^[20]], Equation 17: ADC(diff)^[20] [Équation 17: ADC(diff)^[20]], Equation 18: ADC(com)^[20] [Équation 18: ADC(com)^[20]], Equation 19: ADC value (12 bits) vs. ADC differential inputs^[20] [Équation 19: ADC value (12 bits) vs. ADC differential inputs.^[20]], Equation 20: ADC(diff) [Équation 20: ADC(diff)], Equation 21: ADC(com) [Équation 21: ADC(com)] and Equation 22: ADC value (12 bits) vs. DAC-A, DAC-B, Vin, VDD_P2V, VSS_M2V and Vref [Équation 22: ADC value (12 bits) vs. DAC-A, DAC-B, Vin, VDD_P2V, VSS_M2V and Vref.]
- added §6.3.3 SSD ADC before Upgrade [§6.4.3 SSD]
- added Equation 23: LSB value in the ADC before upgrade [Équation 23: LSB value in the ADC before upgrade]
- updated §7 References [§7 References]
- updated Table 41: rdoU-Slave to rdoU-Master interface payload "ladder status" [Table 46: rdoU-Slave to rdoU-Master interface payload "ladder status"]
- updated Table 47: rdoU-Slave-FPGA VME registers [Table 52: rdoU slave-FPGA VME registers]
- created §6.5 Test equipment [§6.6 Tests equipment]
- moved §6.5.1 Mechanical validation [§6.6.1 Mechanical validation]
- moved §6.5.2 Debug card [§6.6.3 Debug card]
- moved §6.5.3 Debug isolator (EVAL-ADUM4160EBZ isolator card [§6.6.4 Debug isolator (EVAL-ADUM4160EBZ USB isolator card)]
- moved §6.5.4 Static Fake Hybrid card [§6.6.5 Static Fake Hybrid card]
- moved §6.5.5 Dynamic Fake Hybrid card [§6.6.6 Dynamic Fake Hybrid card]
- moved §6.5.6 Quick RDO (QRDO) [§6.6.7 Quick RDO (QRDO)]
- updated Table 65: ladder FPGA slow-control register (config) [Table 73: ladder-FPGA slow-control register (config)]
- added Figure 15: rdoU master-FPGA MSSI cabling [Figure 17: rdoU master FPGA MSSI cabling]
- updated Table 29: rdoU VME registers [Table 30: rdoU VME registers]
- added Figure 59: EVAL-ADUM4160EBZ USB isolator card [Figure 83: EVAL-ADUM4160EBZ USB isolator card]
- added Figure 20: schematics of the latch up detection in the ladder card^{[17][18][19]} [Figure 29: schematics of the latch up detection in the ladder card^{[17][18][19]}]
- updated Figure 21: schematics of the analog part in the ladder card^{[20][21][22]} [Figure 30: schematics of the analog part in the ladder card^{[20][21][22]}]
- updated Table 53: ladder card typical power requirement (normal operation)^{[9][14][15][16][17][18][19][20][21][22]} [Table 58: ladder card typical power requirement (normal operation)^{[9][14][15][16][17][18][19][20][21][22]}] and Table 54: ladder card maximum power requirement (normal operation)^{[9][14][15][16][17][18][19][20][21][22]} [Table 59: ladder card maximum power requirement (normal operation)^{[9][14][15][16][17][18][19][20][21][22]}]
- added Table 55: ladder card typical power requirement (debug operation without hybrid, without RDO)^{[9][14][15][16][17][18][19][20][21][22][23][24]} [Table 60: ladder card typical power requirement (debug operation without hybrid, without RDO)^{[9][14][15][16][17][18][19][20][21][22][23][24]}]
- Version 1.6: August 23rd, 2011 – February 02nd, 2012
- added Table 22: Raw Data organization RDO-slave block when slave is not used [Table 22: Raw Data organization RDO-slave block when slave is not used], Table 61: ladder-end current requirement for low voltage power supplies [Table 61: ladder-end current requirement for low voltage power supplies], Table 62: ladder current requirement for bias power supply [Table 62: ladder current requirement for bias power supply], Table 82: ladder-FPGA slow-control register (bypass) [Table 82: ladder-FPGA slow-control register (bypass)], Table 98: debug interface connector [Table 98: debug interface connector], Table 99: edge connector to debug interface [Table 99: Edge connector to debug interface] and Table 100: SCSI68 connector to Front-End interface [Table 100: SCSI68 connector to Front-End interface]

- updated Table 29: rdoU VME addresses [Table 29: rdoU VME addresses], Table 30: rdoU VME registers [Table 30: rdoU VME registers], Table 31: rdoU VME-FPGA VME registers [Table 31: rdoU VME-FPGA VME registers], Table 32: ladder-FPGA slow-control register (roboclock phase)^[1] effect on Clock36MHz-slz4 and clock36MHz-slz3 [Table 32: ladder-FPGA slow-control register (roboclock_phase)^[1] effect on Clock36MHz-slz4 and clock36MHz-slz], Table 33: ladder-FPGA slow-control register (roboclock phase)^[1] effect on Clock36MHz-slz2 and clock36MHz-slz1 [Table 33: ladder-FPGA slow-control register (roboclock_phase)^[1] effect on Clock36MHz-slz2 and clock36MHz-slz], Table 34: ladder-FPGA slow-control register (roboclock phase)^[1] effect on feed-back and clock36MHz-slz0 [Table 34: ladder-FPGA slow-control register (roboclock_phase)^[1] effect on feed-back and clock36MHz-slz0], Table 35: ladder-FPGA slow-control register (roboclock phase)^[1] effect on clock36MHz-vme FPGA [Table 35: ladder-FPGA slow-control register (roboclock_phase)^[1] effect on Clock36MHz-vme FPGA], Table 37: rdoU master-FPGA TRG part VME registers [Table 37: rdoU master-FPGA TRG part VME registers], Table 56: GBIC: ladder to RDO-slave interface payload "acquisition" [Table 56: GBIC: ladder to RDO-slave interface payload "acquisition"] and Table 60: ladder-FPGA slow-control register [Table 64: ladder-FPGA slow-control registers]
- added Figure 11: rdoU-FPGA configuration scheme for master FPGA [Figure 11: rdoU FPGA configuration scheme for master FPGA], Figure 12: rdoU-FPGA configuration scheme for slave FPGAs [Figure 12: rdoU FPGA configuration scheme for slave FPGAs], Figure 20: DS90UR124^[14] 5-43 MHz DC-Balanced 24-Bit LVDS serializer block diagram [Figure 20: DS90UR241^[14] 5-43 MHz DC-Balanced 24-Bit LVDS serializer block diagram], Figure 21: DS90UR124^[14] 5-43 MHz DC-Balanced 24-Bit LVDS serializer timing [Figure 21: DS90UR241^[14] 5-43 MHz DC-Balanced 24-Bit LVDS serializer timing], Figure 22: DS90UR124^[14] 5-43 MHz DC-Balanced 24-Bit LVDS serializer pinout (top view) [Figure 22: DS90UR241^[14] 5-43 MHz DC-Balanced 24-Bit LVDS serializer pinout (top view)], Figure 23: DS90UR124^[14] 5-43 MHz DC-Balanced 24-Bit LVDS deserializer block diagram [Figure 23: DS90UR124^[14] 5-43 MHz DC-Balanced 24-Bit LVDS deserializer block diagram], Figure 24: DS90UR124^[14] 5-43 MHz DC-Balanced 24-Bit LVDS deserializer timing [Figure 24: DS90UR124^[14] 5-43 MHz DC-Balanced 24-Bit LVDS deserializer timing], Figure 25: DS90UR124^[14] 5-43 MHz DC-Balanced 24-Bit LVDS deserializer pinout (top view) [Figure 25: DS90UR124^[14] 5-43 MHz DC-Balanced 24-Bit LVDS deserializer pinout (top view)], Figure 28: ladder card switches [Figure 28: Ladder card switches], Figure 37: ladder card preproduction dimensions (in mm) [Figure 37: ladder card preproduction dimensions (in mm)], Figure 38: ladder card preproduction circuit top [Figure 38: ladder card preproduction circuit top], Figure 39: ladder card preproduction circuit bottom [Figure 39: ladder card preproduction circuit bottom], Figure 43: VHDL simulation of a readout card connected to complete ladder sides (L0+L2a) using ModelSim-Altera^[12] starter edition 6.6d [Figure 43: VHDL simulation of a readout card connected to complete ladder sides (L0+L2a) using ModelSim-Altera^[12] starter edition 6.6d], Figure 52: SSD numbering at SSD level before Upgrade^[31] [Figure 52: SSD numbering at SSD level before Upgrade^[31]], Figure 53: SSD numbering at ladder level before Upgrade^[31] [Figure 53: SSD numbering at ladder level before Upgrade^[31]], Figure 54: SSD numbering at module level before Upgrade^[31] [Figure 54: SSD numbering at module level before Upgrade^[31]], Figure 55: SSD numbering at Wafer level before Upgrade^[31] [Figure 55: SSD numbering at Wafer level before Upgrade^[31]], Figure 57: Alice128C timing^[5] [Figure 57: Alice128C timing^[5]], Figure 67: optical transceiver CAD model for mechanical prototype (CAD drawing by Gérard Guilloux) [Figure 67: optical transceiver CAD model for mechanical prototype (CAD drawing by Gérard Guilloux)], Figure 70: ladder card mechanical prototype dimensions (in mm) [Figure 70: ladder card mechanical prototype dimensions (in mm)], Figure 73: mechanical CAD drawing of ladders on the support, with the shroud on top. NO SCALE INDICATED!!! [Figure 73: mechanical CAD drawing of ladders on the support, with the shroud on top. NO SCALE INDICATED!!!], Figure 74: ladder card mechanical prototype with 45° rotated optical transceiver [Figure 74: ladder card mechanical prototype with 45° rotated optical transceiver], Figure 74: ladder card prototype with 45° rotated optical transceiver [Figure 75: ladder card prototype with 45° rotated optical transceiver], 74: ladder card preprod prototype with 30° rotated optical transceiver [Figure 76: ladder card preprod with 30° rotated optical transceiver], Figure 89: QRDO partially assembled [Figure 89: QRDO partially assembled.], Figure 90: FPGADC_V1 pre-

placement [Figure 90: FPGADC_V1 pre-placement.] and Figure 91: FPGADC_V1 level shifter using AD5449^[28] [Figure 91: FPGADC_V1 level shifter using AD5449^[28].]

- updated Figure 16: rdoU master-FPGA event state machines [Figure 16: rdoU master FPGA event state machines]
- moved Figure 77: ladder card prototype dimensions (in mm) [Figure 77: ladder card prototype dimensions (in mm)] and Figure 78: ladder card prototype circuit top [Figure 78: ladder card prototype circuit top] and Figure 79: ladder card prototype circuit bottom [Figure 79: ladder card prototype circuit bottom] to §6.6.2 Ladder card prototype [§6.6.2 Ladder card prototype]
- added §6.6.7 FPGADC_star_v1 [§6.6.8 FPGADC_star_v1], §6.8 Planning [§6.8 Planning] and §6.6.2 Ladder card prototype [§6.6.2 Ladder card prototype]
- updated §6.1 Pin swapping rules for Altera^[8] FPGA [§6.1 Pin swapping rules for Altera^[8] FPGA] and §7 References [§7 References]

3 Introduction

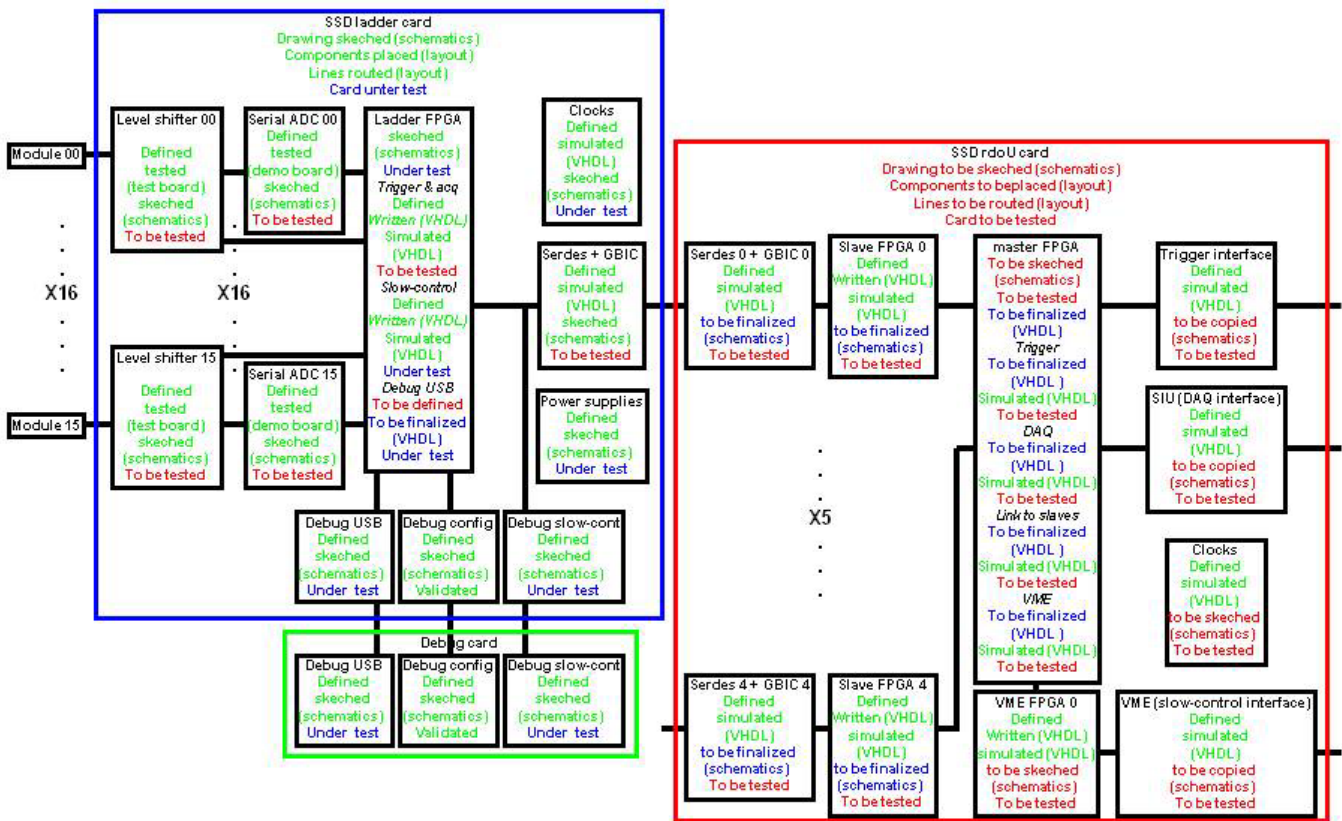


Figure 1: SSD-Upgrade to do list overview

3.1 SSD-Upgrade overview

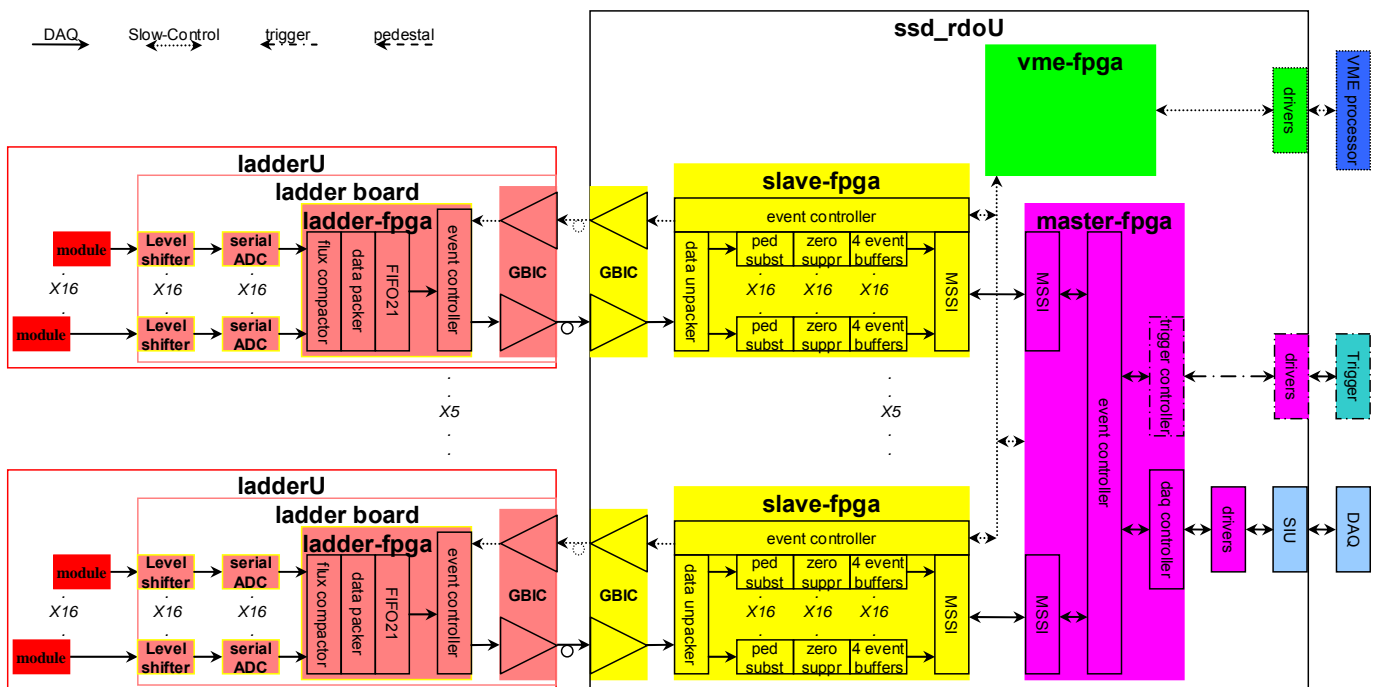


Figure 2: SSD-Upgrade overview

3.2 SSD-Upgrade power supplies

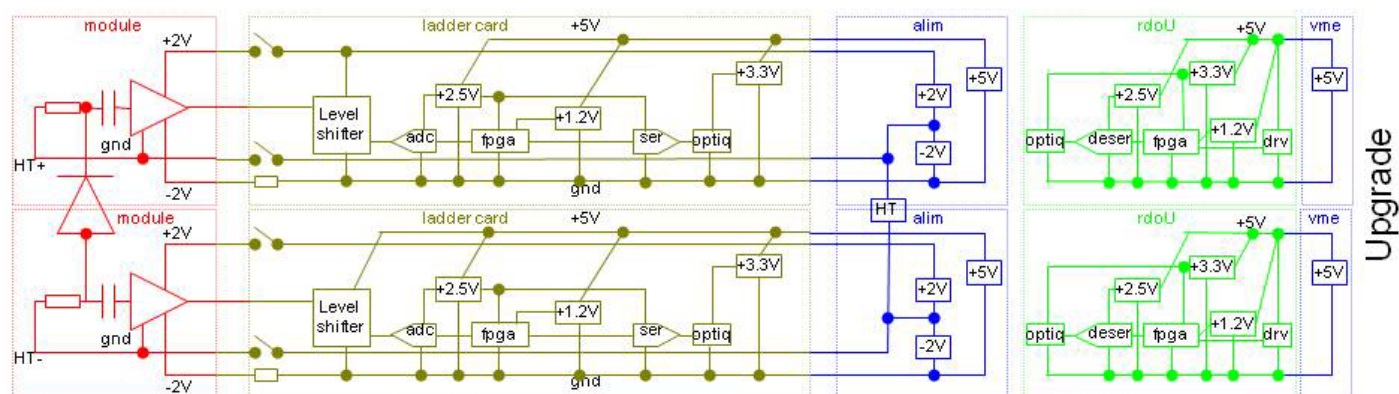


Figure 3: SSD-Upgrade power supply chain

3.3 SSD-Upgrade cables

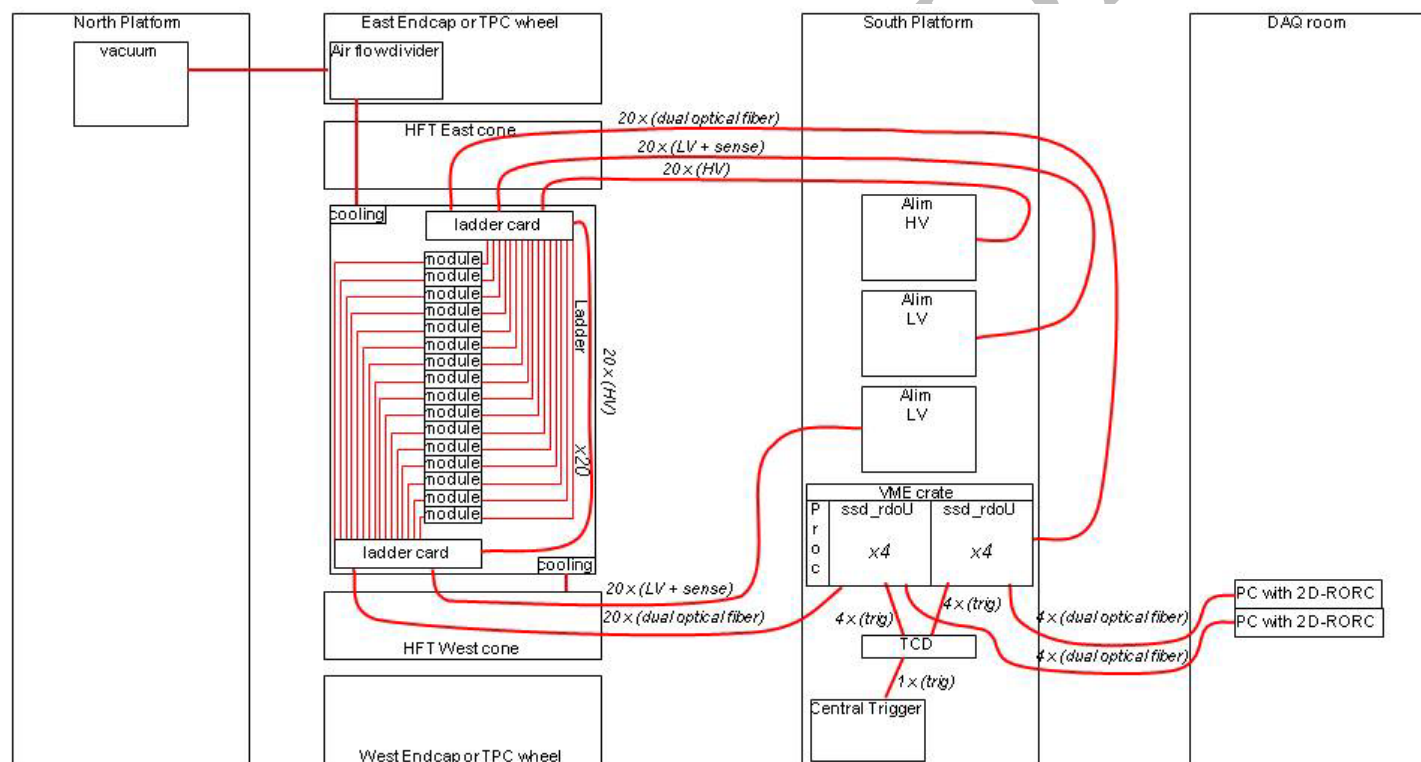


Figure 4: SSD-Upgrade cables

3.4 SSD-Upgrade Slow-Control

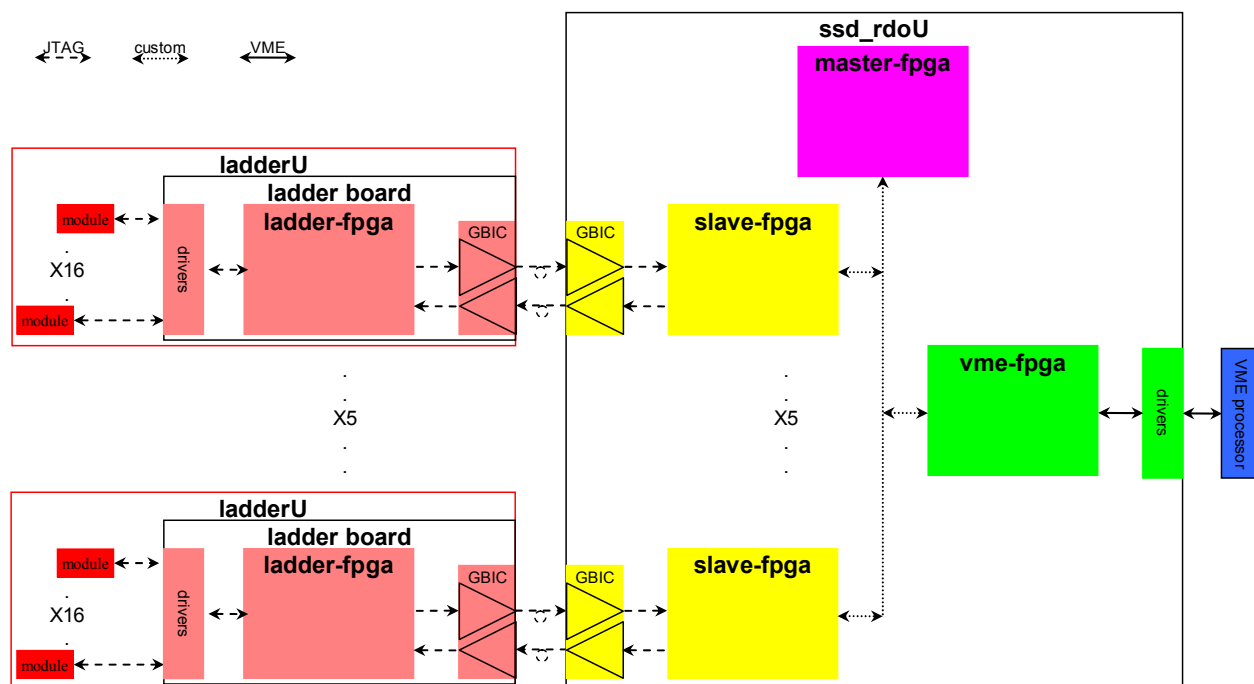
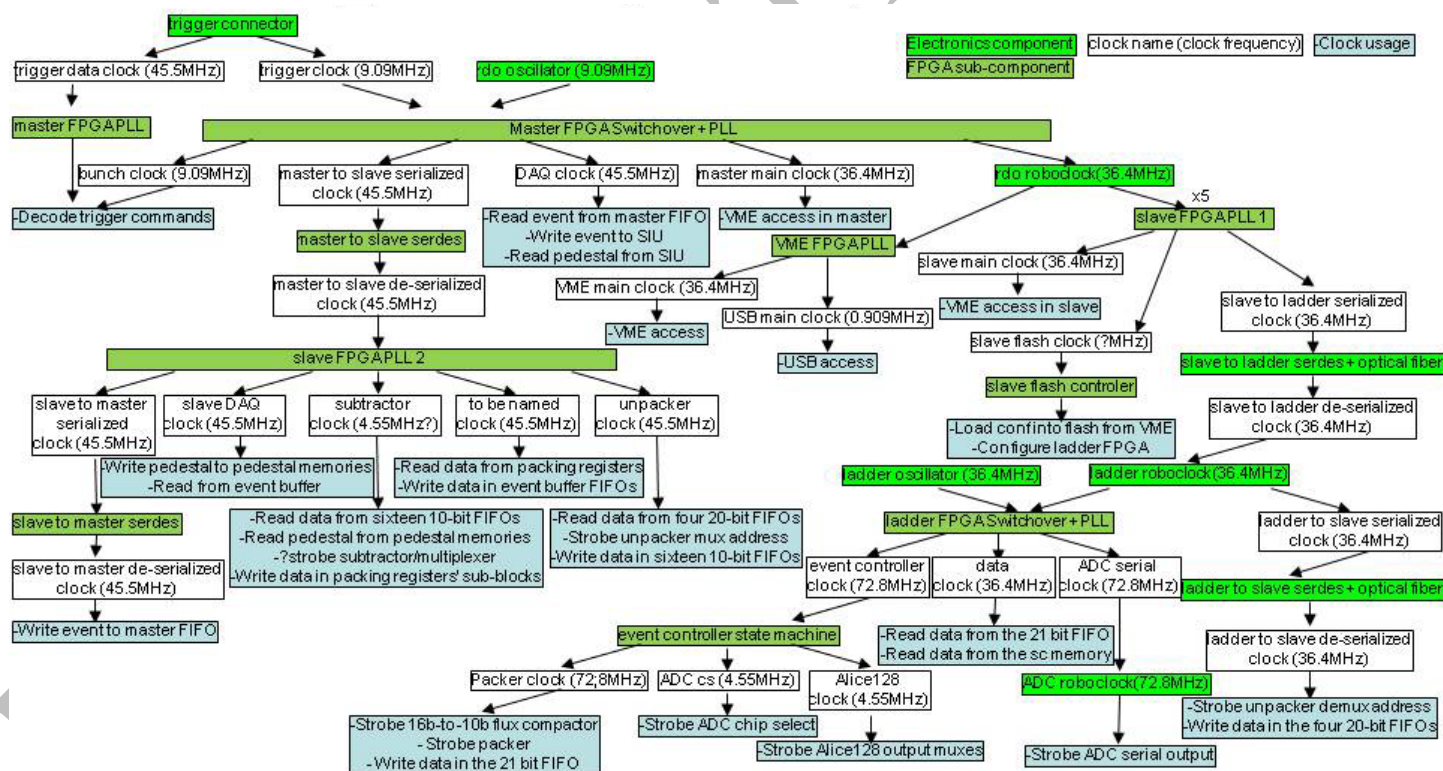


Figure 5: SSD-Upgrade Slow-Control

3.5 SSD-Upgrade clocks



rd0 roboclock^[1]: output pair 1 to VME and "reboucle" (feedback); output pairs 2, 3 & 4 to slave FPGAs.

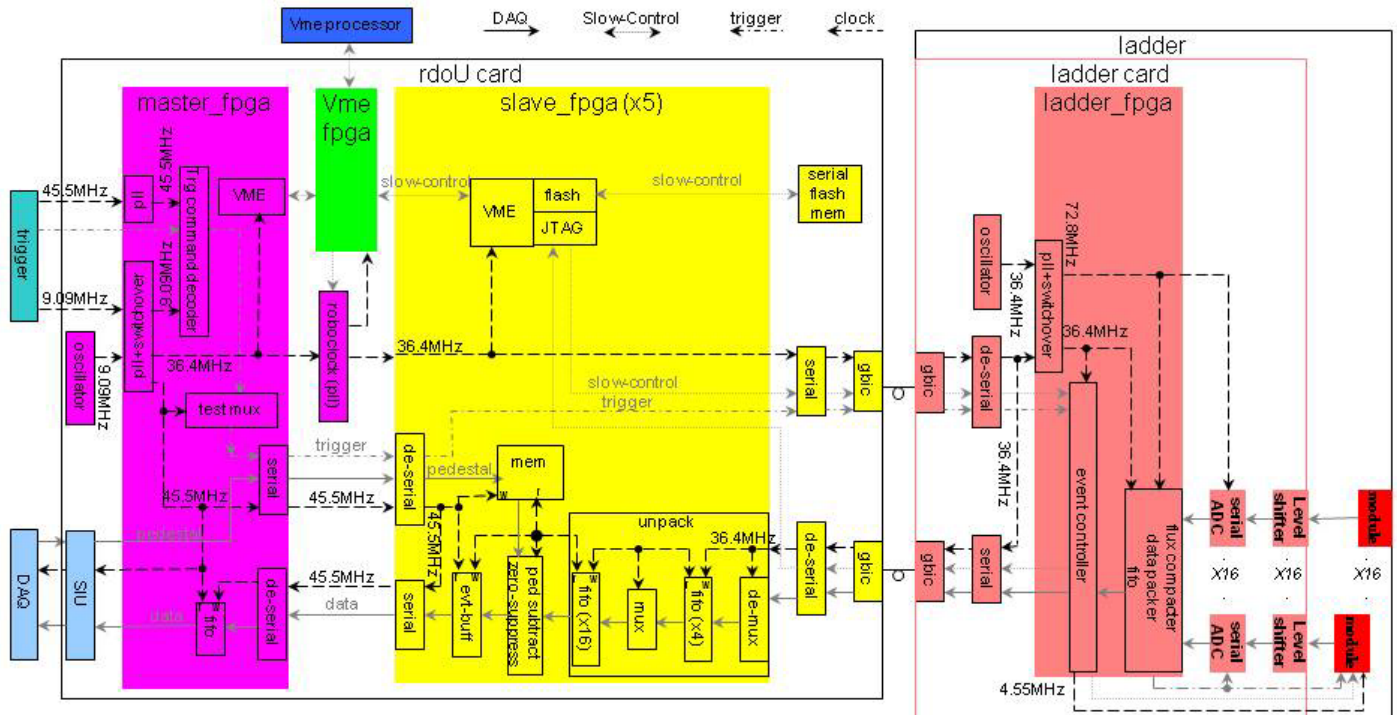


Figure 7: SSD-Upgrade Clock chain

A switchover in the Master-FPGA selects between the trigger clock (9.09MHz) and the local oscillator (also 9.09MHz). Three outputs of this switchover will give the "bunch clock" (9.09MHz), the "main clock" (36.4MHz) and the "daq clock" (45.5MHz).

A switchover in the ladder-FPGA selects between the "slave to ladder de-serialized clock" (36.4MHz) and the local oscillator (also 36.4MHz). Two outputs of this switchover will give the "ladder main clock" (36.4MHz) and the "ADC serial clock" (72.8MHz).

Here is my old (07 April 2009) opinion on the usage of the different clocks:

- "main clock" (36.4MHz)
 - is distributed on the readoutU card, using a roboclock^[1]
 - is used in all FPGAs for the VME interface
 - becomes "slave to ladder serialized clock" (36.4MHz) in the slave-FPGA
- "daq clock" (45.5MHz)
 - reads events from master FIFO
 - writes events to SIU
 - reads pedestal from SIU
 - becomes "master to slave serialized clock" (45.5MHz) in the Master-FPGA
- "master to slave de-serialized clock" (45.5MHz)
 - writes pedestals to the pedestal memories
 - reads from event buffers
 - becomes "slave to master serialized clock" (45.5MHz) in the slave-FPGA
- "slave to master de-serialized clock" (45.5MHz)
 - writes in the master FIFO (1 full hybrid event capacity for DAQ XonXoff)
- "slave to ladder de-serialized clock" (36.4MHz)
 - becomes "ladder main clock" (36.4MHz) in the ladder-FPGA
 - becomes "ADC serial clock" (72.8MHz) in the ladder-FPGA
- "ladder main clock" (36.4MHz)
 - becomes "ladder to slave serialized clock" (36.4MHz) in the ladder-FPGA
- "ladder to slave de-serialized clock" (36.4MHz)
 - strokes the unpacker demux addresses
 - writes data in the four 20bit FIFOs

- "unpacker clock" (MHz)
 - reads data from the four 20bit FIFOs
 - strokes the unpacker mux addresses
 - writes data in the sixteen 10bit FIFOS
- "subtractor clock" (4.55MHz?)
 - reads data from the sixteen 10bit FIFOS
 - reads pedestals from the pedestal memories
 - ? strokes the subtractor/multiplexer
 - writes data in the packing registers' sub-blocks
- "to be named clock" (MHz)
 - reads data from the packing registers
 - writes data in the event buffer FIFOs

"unpacker clock", "subtractor clock" and "to be named clock" can derive either from "main clock" (36.4MHz) or from "master to slave de-serialized clock" (45.5MHz).

3.6 SSD-Upgrade readout time

It takes $768 \times 200\text{ns} = 153.600\mu\text{s}$ to read the 768 channels of a complete hybrid at 5MHz (busy duration).

It takes $768/3 \times 16 \times 5 \times 20\text{ns} = 409.6\mu\text{s}$ to send the 768 x 16 x 5 channels of a complete readoutU board to the DAQ1000 system without zero suppression.

It takes $768 \times 3\% \times 16 \times 5 \times 20\text{ns} = 36.864\mu\text{s}$ to send the non zero channels of a complete readoutU board to the DAQ1000 system using zero suppression with the typical 3% occupancy.

4 The upgraded Readout card (rdoU)

4.1 rdoU Overview

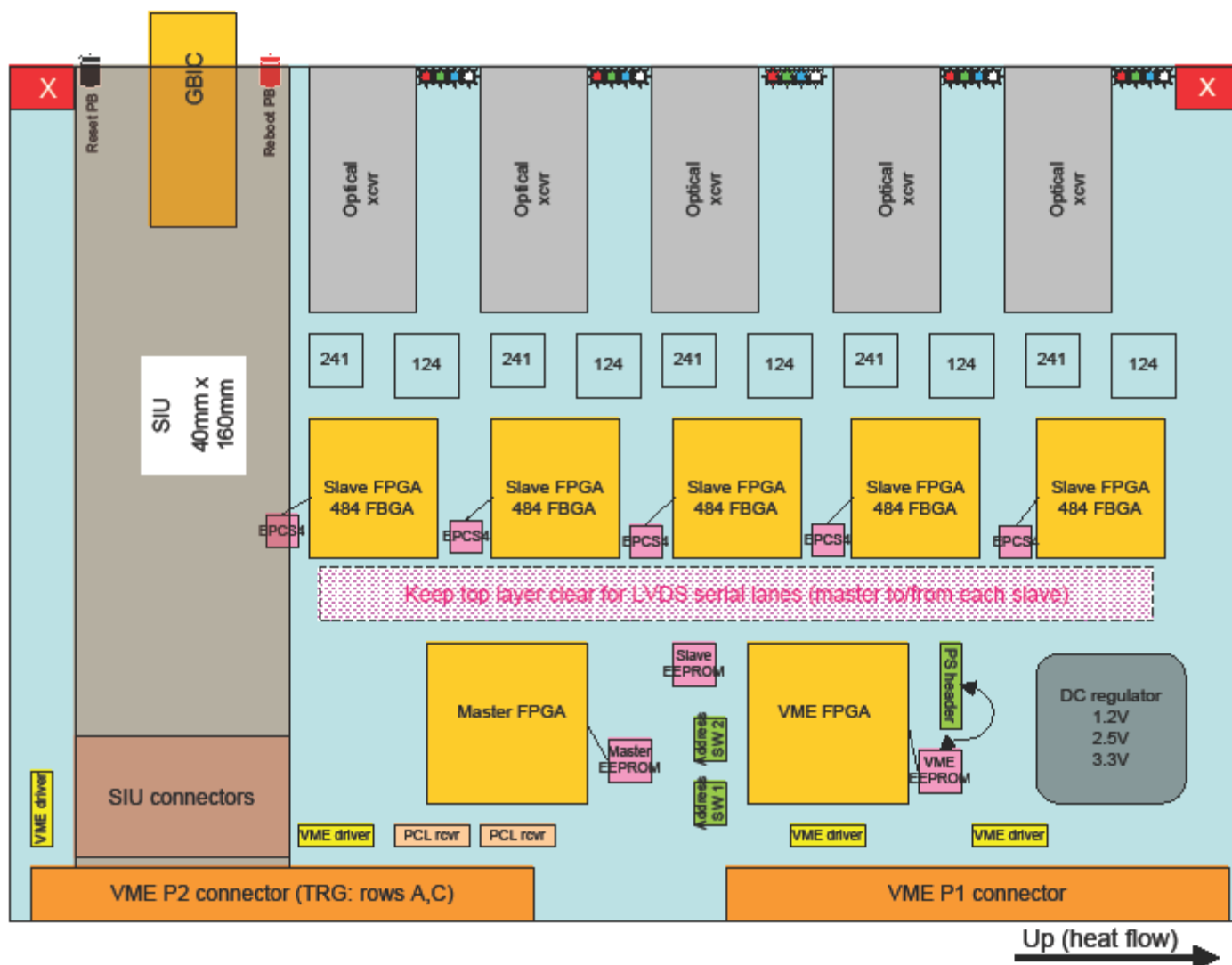


Figure 8: rdoU trial layout (by BNL October 19, 2010)

4.2 rdoU Power supply management

4.3 rdoU Clock management

A switchover in the Master-FPGA selects between the trigger clock (9.09MHz) and the local oscillator (also 9.09MHz). Three outputs of this switchover will give the "bunch clock" (9.09MHz), the "main clock" (36.4MHz) and the "daq clock" (45.5MHz).

A switchover in the ladder-FPGA selects between the "slave to ladder de-serialized clock" (36.4MHz) and the local oscillator (also 36.4MHz). Two outputs of this switchover will give the "ladder main clock" (36.4MHz) and the "ADC serial clock" (72.8MHz).

Here is my old (07 April 2009) opinion on the usage of the different clocks:

- "main clock" (36.4MHz)

Christophe-M.-A. Renard, Subatech, IN2P3/CNRS-L'UNAM, Nantes, F44307, France

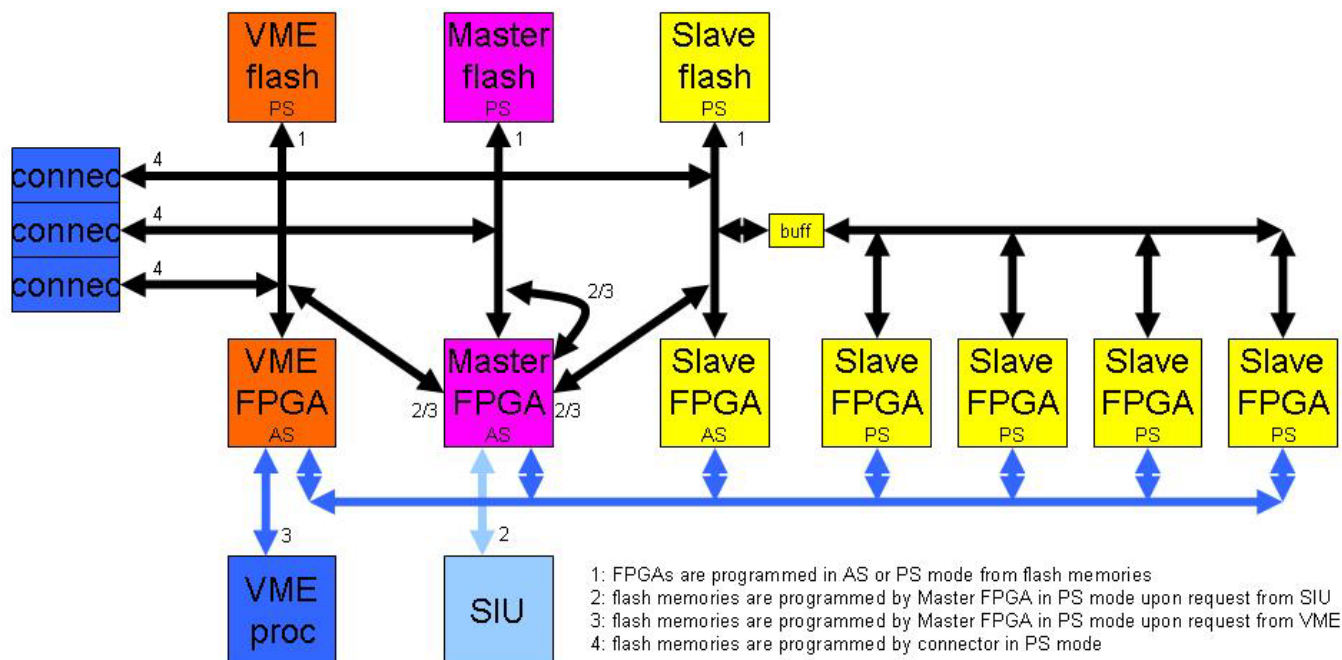
Micheal-J. LeVine, BNL, Upton, NY, USA

Stéphane Bouvier, Subatech, IN2P3/CNRS-L'UNAM, Nantes, F44307, France

- is distributed on the readoutU card, using a roboclock^[1]
- is used in all FPGAs for the VME interface
- becomes "slave to ladder serialized clock" (36.4MHz) in the slave-FPGA
- "daq clock" (45.5MHz)
 - reads events from master FIFO
 - writes events to SIU
 - reads pedestal from SIU
 - becomes "master to slave serialized clock" (45.5MHz) in the Master-FPGA
- "master to slave de-serialized clock" (45.5MHz)
 - writes pedestals to the pedestal memories
 - reads from event buffers
 - becomes "slave to master serialized clock" (45.5MHz) in the slave-FPGA
- "slave to master de-serialized clock" (45.5MHz)
 - writes in the master FIFO (1 full hybrid event capacity for DAQ XonXoff)
- "slave to ladder de-serialized clock" (36.4MHz)
 - becomes "ladder main clock" (36.4MHz) in the ladder-FPGA
 - becomes "ADC serial clock" (72.8MHz) in the ladder-FPGA
- "ladder main clock" (36.4MHz)
 - becomes "ladder to slave serialized clock" (36.4MHz) in the ladder-FPGA
- "ladder to slave de-serialized clock" (36.4MHz)
 - strokes the unpacker demux addresses
 - writes data in the four 20bit FIFOs
- "unpacker clock" (MHz)
 - reads data from the four 20bit FIFOs
 - strokes the unpacker mux addresses
 - writes data in the sixteen 10bit FIFOS
- "subtractor clock" (4.55MHz?)
 - reads data from the sixteen 10bit FIFOS
 - reads pedestals from the pedestal memories
 - ? strokes the subtractor/multiplexer
 - writes data in the packing registers' sub-blocks
- "to be named clock" (MHz)
 - reads data from the packing registers
 - writes data in the event buffer FIFOs

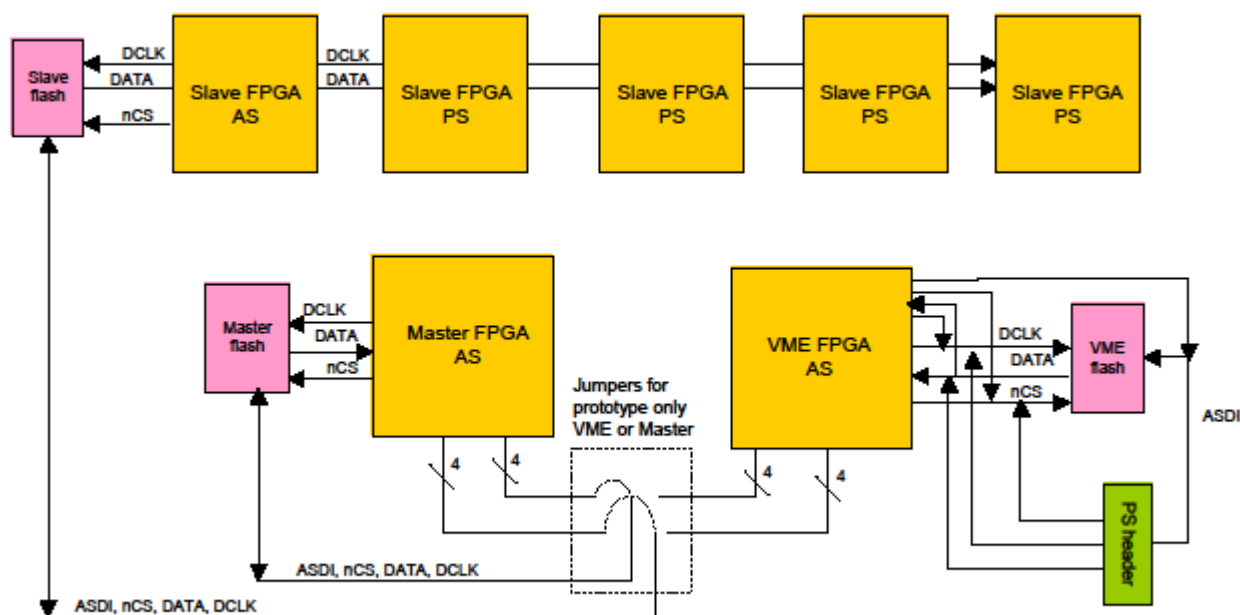
"unpacker clock", "subtractor clock" and "to be named clock" can derive either from "main clock" (36.4MHz) or from "master to slave de-serialized clock" (45.5MHz).

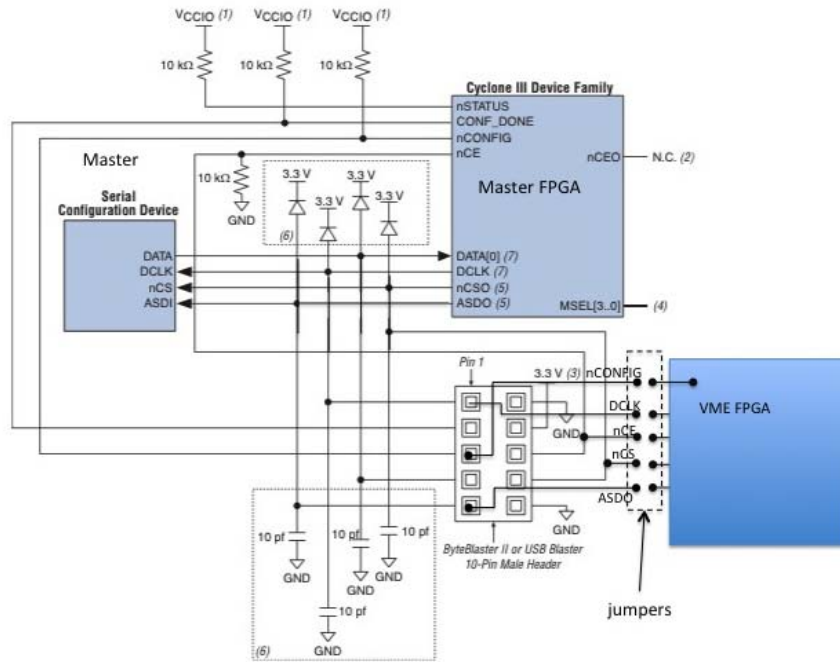
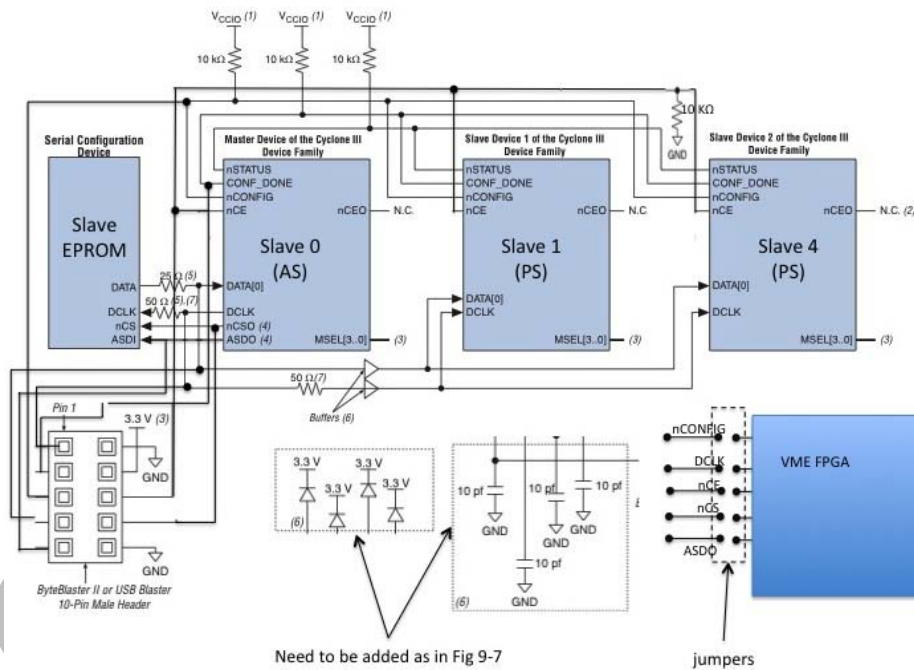
4.4 rdoU FPGA configuration

Figure 9: *rdoU FPGA configuration scheme (SUBATECH proposition)*

Connections for slaves: Cyclone III Handbook, Figure 10-6.

Multi-Device AS Configuration in which Devices Receive the Same Data with a Single SRAM Object File

Header connections as in Fig 10-7, Cyclone III Handbook
Active Serial Configuration (Serial Configuration Devices)Figure 10: *rdoU FPGA configuration scheme (BNL proposition)*

Figure 11: *rdoU* FPGA configuration scheme for master FPGAFigure 12: *rdoU* FPGA configuration scheme for slave FPGAs

4.5 *rdoU* Trigger interface^[2]

4.6 Overview

9.09MHz(110ns) or 9.383MHz(106.57ns) ?

4.6.1 *rdoU* Trigger connector

2x10 Male, right angle, front panel, latch (type HE10)

Electrical type	Signal name	Pin number	Signal direction	Signal direction	Pin number	Signal name	Electrical type
	GND	20	out	out	19	TRG_STATUS_BACK_P	PECL
Opto-isolated transistor	TRG_BUSY_BACK_emitter	18	out	out	17	TRG_BUSY_BACK_collector	Opto-isolated transistor
PECL Diff	TRG_DETEC_CLOCK2_N	16	in	in	15	TRG_DETEC_CLOCK2_P	PECL Diff
PECL Diff	TRG_DETEC_CLOCK1_N	14	in	in	13	TRG_DETEC_CLOCK1_P	PECL Diff
PECL Diff	TRG_DATA_CLOCK_N	12	in	in	11	TRG_DATA_CLOCK_P	PECL Diff
PECL Diff	TRG_DATA_N(3)	10	in	in	9	TRG_DATA_P(3)	PECL Diff
PECL Diff	TRG_DATA_N(2)	8	in	in	7	TRG_DATA_P(2)	PECL Diff
PECL Diff	TRG_DATA_N(1)	6	in	in	5	TRG_DATA_P(1)	PECL Diff
PECL Diff	TRG_DATA_N(0)	4	in	in	3	TRG_DATA_P(0)	PECL Diff
PECL Diff	TRG_RHIC_STROBE_N	2	in	in	1	TRG_RHIC_STROBE_P	PECL Diff

Table 2: trigger connector

Master FPGA asserts its "busy_back" signal to the central trigger (see "Table 2: trigger connector" and "Figure 13: BNL recommendation for Trigger interface") when bit "Slave and ladder OK" (see "Table 41: RDO-Slave to RDO-Master interface") is not '1' while the bit corresponding to this slave in "ladder_used" (see config register in "Table 37: rdoU master-FPGA TRG part VME registers") is '1'.

The electrical type of the Busy_Back signal was PECL for the SSD. It has changed for upgraded detectors. BNL requirements for the new implementation are in [3] and "Figure 13: BNL recommendation for Trigger interface".

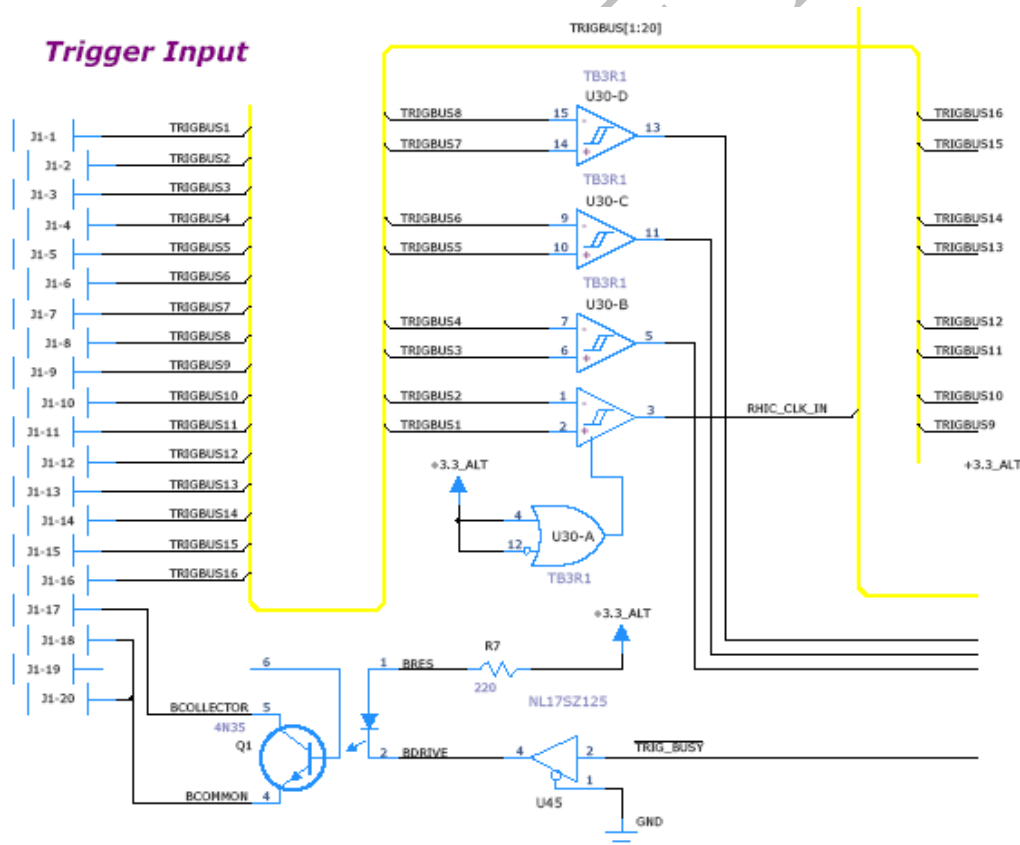


Figure 13: BNL recommendation for Trigger interface

Master FPGA asserts its "busy_back" signal to the central trigger (see "Table 2: trigger connector" and "Figure 13: BNL recommendation for Trigger interface") when bit "Slave and ladder OK" (see "Table 41: RDO-Slave to RDO-Master interface") is not '1' while the bit corresponding to this slave in "ladder_used" (see config register in "Table 37: rdoU master-FPGA TRG part VME registers") is '1'.

4.6.2 rdoU Trigger messages

A trigger message consists of 5 words, each containing 4 bits.

These 5 words consist of a trigger command, a daq command and a token

pos	Name	Number of words
0	Trigger command	1
1	DAQ command	1
2-4	token	3

Table 3: trigger message

#	Requirements from STAR	Implemented in SSD	Implemented in SSD Upgrade
0	idle (followed by a null Token)	idle	idle
1	clear -> resets the RHIC strobe counter.	resets the RHIC_strobe counter.	resets the RHIC_strobe counter.
2	master reset	resets the acquisition state machines inside the readout board FPGAs.	resets the acquisition state machines inside the readout board FPGAs.
3	reserved	executed as idle.	executed as idle.
4	Trigger0 -> physics readout	physics readout	physics readout (pedestal subtracted & zero suppression mode)
5	Trigger1 -> reserved (same as Trigger0)	pedestal readout	physics readout (pedestal subtracted NO zero suppression mode)
6	Trigger2 -> reserved (same as Trigger0)	Executed exactly as Trigger0	pedestal readout (ADC values readout mode)
7	Trigger3 -> same as Trigger0 + copy into local buffer	Executed exactly as Trigger0 (no local buffer).	pedestal debug (pedestal memory readout). (no local buffer).
8	Pulser0 -> ladder test	calibration pulse, (pedestal subtracted & zero suppression mode)	calibration pulse, (pedestal subtracted & zero suppression mode)
9	Pulser1 -> ladder test	transparent mode readout: only reads the strip on each hybrid which address is defined in the transparent mode Slow-Control's register	calibration pulse, (pedestal subtracted NO zero suppression mode)
A	Pulser2 -> electrical calibration pulse	Executed exactly as Pulser0	calibration pulse, pedestal readout (ADC values readout mode)
B	Pulser3 -> read buffer stored by Trigger3	Executed exactly as Pulser0 (no local buffer).	transparent mode readout: only reads the strip on each hybrid which address is defined in the transparent mode Slow-Control's register (ADC values readout mode). (no local buffer).
C	Config -> house keeping trigger	Executed exactly as Trigger0	executed as idle (not yet implemented in Readout).
D	Abort -> aborts and clears the event identified by the Token	aborts and clears the corresponding event.	aborts and clears the corresponding event.
E	L1accept -> the event identified by the Token is accepted by level 1 trigger (L1)	executed as idle (not implemented in Readout).	executed as idle (not implemented in Readout).
F	L2accept -> the event identified by the Token is accepted by level 2 trigger (L2)	executed as idle (not implemented in Readout).	sends the corresponding event to the DAQ

Table 4: trigger message commands and associated action

pos	Name/value	Number of words
0	Token high	1
1	Token mid	1
2	Token low	1

Table 5: trigger message token

Request from BNL (Tonko, October 2011): Every message with non zero trigger command received between L0 and L2 should be registered and sent to DAQ with event data. A limit of 128 registered trigger messages per event is foreseen.

4.6.3 rdoU Trigger Protocol

Non idle message decoded by master						
clear	reset	trigger	pulser	config	L2abort	L2acc
Master asserts busy back signal						
Master resets RHIC_strobe counter	Master resets slaves' decoder	Master sets readout mode			Master sends abort command to slave	Master sends header to DAQ
↓	Slaves send 0xFEEDBEEF trailer to master	↓	Master asks for TEST edge	↓	↓	Master sends read command to slave 0
↓	Slaves reset their decoders	↓	<ul style="list-style-type: none"> Slaves assert TEST signal to ladders Ladders transmit TEST signal to hybrids 	↓	↓	<ul style="list-style-type: none"> Slave 0 sends data to master master transmits data to DAQ
↓	↓	Master sets delay before HOLD			↓	Slave 0 sends trailer to master
↓	↓	Slaves assert "Busy fetching data from				Master sends read command to slave 1
↓	↓	Slaves wait for specified delay			↓	<ul style="list-style-type: none"> Slave 1 sends data to master master transmits data to DAQ
↓	<ul style="list-style-type: none"> Slaves assert HOLD Ladders transmit HOLD signal to hybrids 			↓	↓	Slave 1 sends trailer to master
↓	Ladders assert busy signal			↓	↓	Master sends read command to slave 2
↓	Slaves de-assert HOLD	slaves send TOKEN pulse to ladders		↓	Slaves de-assert HOLD signal	<ul style="list-style-type: none"> Slave 2 sends data to master master transmits data to DAQ
↓	↓	<ul style="list-style-type: none"> Ladders read data in hybrids Ladders convert data Ladders pack data Ladders send data to slaves Slaves unpack data Slaves compute data according to mode Slaves store data according to mode and write address 		Slaves store config according to write address	Ladders read data in hybrids at double speed	Slave 2 sends trailer to master
↓	Ladders de-assert busy signal			↓	Ladders de-assert busy signal	Master sends read command to slave 3
↓	Slaves send idle to master	↓	<ul style="list-style-type: none"> Slaves de-assert TEST Ladders transmit TEST signal to hybrids 	↓	↓	<ul style="list-style-type: none"> Slave 3 sends data to master master transmits data to DAQ
↓	↓	Slaves de-assert HOLD signal			↓	Slave 3 sends trailer to master
↓	↓	Ladders transmit HOLD signal to hybrids			↓	Master sends read command to slave 4
↓	Master resets its counters and state machines	Slaves de-assert "Busy fetching data from ladder"				<ul style="list-style-type: none"> Slave 4 sends data to master master transmits data to DAQ
↓	Master resets its counters and state machines	Master marks corresponding event buffer as used			Master marks corresponding event buffer as free	Slave 4 sends trailer to master
Master de-asserts busy back signal						Master sends trailer to DAQ

Table 6: *trigger protocol*

- SSD Upgrade busy time corresponds to the time needed to fill one event buffer.
- Signal TEST edge is used for calibration purpose. The physics signal being a negative pulse on P-side of the detector and a positive pulse on N-side of the detector, a different edge (rising or falling) must be

used for the different sides of the module (side N or side P) to generate a physics-like pulse. TEST signal is complemented in ladder FPGA when bit "hv side" in "Table 73: *ladder-FPGA slow-control register (config)*" is asserted (=1'). The signal should be de-asserted (useless edge) only after the complete event is read-out to avoid crosstalk with stored data in the Alice128^{[4][5]} chips.

- TOKEN pulse duration should last, at least, one 4.55MHz clock tick

4.7 rdoU DAQ interface (DDL using SIU daughter card)

4.7.1 rdoU DAQ connector

4.7.2 rdoU DAQ Raw data

Raw data for one event consist of a master header, 5 slave blocks and an end of event. Each slave block corresponds to one ladder side.

name	Number of words		
	min	typ	max
Master header	4	4	4
Slave 0 block	49	417	12321
Slave 1 block	49	417	12321
Slave 2 block	49	417	12321
Slave 3 block	49	417	12321
Slave 4 block	49	417	12321
End of event	1	1	1
total	250	2090	61610

Table 7: Raw Data organization

Typ corresponds to typical 3% occupancy

4.7.2.1 rdoU DAQ Raw data Master header

pos	Name/value	Number of words
0	Master total words	1
1	Master ID	1
2	Trigger message	1
3	spacer (0xDEADFACE)	1

Table 8: Raw Data organization RDO master header

19 free bits	13 bits	Number of words
Master free	Master total words (including this word)	1

Table 9: Raw Data organization RDO master total words

xFFFFFFFF means that master total words was not calculated

1 free bits	1 bit	1 bit	5 bits	8 bits	4 bits	4 bits	4 free bits	4 bits	Number of words
Master free	Trg pres	DAQ pres	Lad used	VHDL version	RDO #	RDO serial	daq #	card type	1

Table 10: Raw Data organization RDO master ID

ladder_used vector is set by slow-control in trigger part of master FPGA. Master FPGA generates fake data for slave(s)/ladder(s) that are not used (corresponding bit in "ladder_used" equal to '0').

12 counter bits	4 trigger bits	4 DAQ bits	12 token bits	Number of words
RHIC strobe counter	trigger	DAQ	token	1

Table 11: Raw Data organization RDO trigger message

4.7.2.2 rdoU DAQ Raw data Slave block

pos	Name/value	Number of words		
		min	typ	max
0-16	Slave header	17	17	17
17-a	Hybrid 0 block	2	25	769
a-b	Hybrid 1 block	2	25	769
b-c	Hybrid 2 block	2	25	769
c-d	Hybrid 3 block	2	25	769
d-e	Hybrid 4 block	2	25	769
e-f	Hybrid 5 block	2	25	769
f-g	Hybrid 6 block	2	25	769
g-h	Hybrid 7 block	2	25	769
h-i	Hybrid 8 block	2	25	769
i-j	Hybrid 9 block	2	25	769
j-k	Hybrid 10 block	2	25	769
k-l	Hybrid 11 block	2	25	769
l-m	Hybrid 12 block	2	25	769
m-n	Hybrid 13 block	2	25	769
n-o	Hybrid 14 block	2	25	769
o-p	Hybrid 15 block	2	25	769

Table 12: Raw Data organization RDO-slave block

Typ corresponds to typical 3% occupancy

4.7.2.2.1 rdoU DAQ Raw data Slave header

pos	Name/value	Number of words
0	Slave total words	1
1	Spacer (0xFACEBEEF)	1
2-9	Ladder status	8
10-15	Hybrid total words	6
16	Spacer (0xBEEFFACE)	1

Table 13: Raw Data organization RDO-slave header

7 free bits	3 slave # bits	3 ladder # bits	1 bit	2 slave mode bits	16 slave total words bits	Number of words
slave free	slave #	ladder #	ladder ok	slave mode	Slave block total words (including this word)	1

Table 14: Raw Data organization RDO-slave total words

Ladder_ok tells that the ladder-FPGA is programmed and the ladder serdes is locked.

This is an error if ladder # is different from slave #.

2 slave mode bits	Associated mode
00	ADC data pedestal subtracted and zero suppressed (physics)
01	ADC data pedestal subtracted
10	Raw ADC data
11	Pedestal memory

Table 15: Raw Data organization RDO-Slave mode

pos	11 free bits	3 Status # bits	2 latchup bits	1 busy bit	1 bit	1 bit	1 bit	12 status bits	Number of words
0	slave free	000	latchup 0-1	ladder_busy	configured	ok	serdes clock used	temperature 0	1
1	slave free	001	latchup 2-3	ladder_busy	des lock	des bist pass	crc error	Temperature 1	1
2	slave free	010	latchup 4-5	ladder_busy	fib tx fault	fib mod absent	fib rx loss	Temperature 2	1
3	slave free	011	latchup 6-7	ladder_busy	usb present	usb rdy_n	debug present	Temperature 3	1
4	slave free	100	latchup 8-9	ladder_busy	test	hold	hv side	nbr_hold	1
5	slave free	101	latchup 10-11	ladder_busy	serial(5)	serial(4)	serial(3)	nbr_test	1
6	slave free	110	latchup 12-13	ladder_busy	serial(2)	serial(1)	serial(0)	nbr_token	1
7	slave free	111	latchup 14-15	ladder_busy	ladder #			nbr_abort	1

Table 16: Raw Data organization ladder status

Serial(5 downto 0) is the serial number of the ladder card. It is defined on the printed circuit board.

Nbr_hold is the number of times that the HOLD signal was asserted since last reset of the ladder card.

Nbr_test is the number of times that the TEST signal was asserted since last reset of the ladder card.

Nbr_token is the number of times that the TOKEN signal was asserted since last reset of the ladder card.

Nbr_abort is the number of times that the HOLD signal was de-asserted during event read since last reset of the ladder card.

“serdes clock used” means that the ladder FPGA switchover is locked on the incoming serdes clock and not on the local crystal clock.

Ladder_busy signal will be asserted in case of an abort: dataready=0 (no more data to send) but event not finished (busy getting the token out of the Alice128^{[4][5]} chips). Slave-FPGA should wait until both ladder_busy and dataready signals are off before de-asserting the “Busy fetching data from ladder” signal to the Master-FPGA.

Debug_present signal indicates that the debug daughter card is plugged in.

Usb_present signal indicates that USB daughter card is plugged in.

“hv side” bit indicates, when asserted, that TEST signal and each bit of ADC data are complemented.

pos	2 free bits	10 hybrid total words bits	10 hybrid total words bits	10 hybrid total words bits	Number of words
0	slave free	Hybrid 2 block total words	Hybrid 1 block total words	Hybrid 0 block total words	1
1	slave free	Hybrid 5 block total words	Hybrid 4 block total words	Hybrid 3 block total words	1
2	slave free	Hybrid 8 block total words	Hybrid 7 block total words	Hybrid 6 block total words	1
3	slave free	Hybrid 11 block total words	Hybrid 10 block total words	Hybrid 9 block total words	1
4	slave free	Hybrid 14 block total words	Hybrid 13 block total words	Hybrid 12 block total words	1
5	slave free	slave free	slave free	Hybrid 15 block total words	1

Table 17: Raw Data organization hybrid total words

4.7.2.2.2 rdoU DAQ Raw data Hybrid block

4.7.2.2.2.1 rdoU DAQ Raw data Hybrid block (ADC values readout mode)

pos	2 free bits	10 data bits	10 data bits	10 data bits	Number of words
0-255	slave free	ADC data	ADC data	ADC data	256
256	0xCAFEFADE				1

Table 18: Raw Data organization hybrid block ADC data

4.7.2.2.2.2 rdoU DAQ Raw data Hybrid block (pedestal memory readout mode)

pos	2 free bits	10 data bits	10 data bits	10 data bits	Number of words
0-255	slave free	Pedestal memory	Pedestal memory	Pedestal memory	256
256	0xCAFEFADE				1

Table 19: Raw Data organization hybrid block pedestal memory

4.7.2.2.2.3 rdoU DAQ Raw data Hybrid block (pedestal subtracted data readout mode)

pos	2 free bits	10 data bits	10 data bits	10 data bits	Number of words
0-255	slave free	Pedestal subtracted	Pedestal subtracted	Pedestal subtracted	256
256	0xCAFEFADE				1

Table 20: Raw Data organization hybrid block pedestal subtracted

4.7.2.2.2.4 rdoU DAQ Raw data Hybrid block (pedestal subtracted and zero suppressed readout mode)

pos	5 free bits	3 slave bits	4 module bits	10 strip addr bits	10 data bits	Number of words		
						min	typ	max
0-n	slave free	Slave #	Module #	Strip #	Pedestal subtracted, zero suppressed	1	24	768
n+1	0xCAFEFADE					1	1	1

Table 21: Raw Data organization hybrid block zero suppressed

Typ corresponds to typical 3% occupancy

4.7.2.3 rdoU DAQ Raw data Slave block when slave is not used

<i>pos</i>	<i>Word name</i>	<i>Value in hex</i>	<i>signification</i>	<i>Number of words</i>
1	Slave total words		Slave #; ladder #; ladder KO; mode 0; 0x31=49 words	1
2	Spacer	FACEBEEF		1
3	Ladder status #0	00000000	Lad status #0; no latchup; not busy; not configured; Nok; temp#0=0°K	1
4	Ladder status #1	00040000	Lad status #1; no latchup; not busy; temp#1=0°K	1
5	Ladder status #2	00087000	Lad status #2; no latchup; not busy; tx fault; fib absent; fib loss; temp#2=0°K	1
6	Ladder status #3	000C2000	Lad status #3; no latchup; not busy; temp#3=0°K	1
7	Ladder status #4	00100000	Lad status #4; no latchup; not busy;	1
8	Ladder status #5	00140000	Lad status #5; no latchup; not busy;	1
9	Ladder status #6	00180000	Lad status #6; no latchup; not busy;	1
10	Ladder status #7		Lad status #7; no latchup; not busy; ladder#, 0 abort	1
11	Hybrid total words #0	00200802		1
12	Hybrid total words #1	00200802		1
13	Hybrid total words #2	00200802		1
14	Hybrid total words #3	00200802		1
15	Hybrid total words #4	00200802		1
16	Hybrid total words #5	00000002		1
17	Spacer	BEEFFACE		1
18	Hybrid #0		Slave #; module#=0; strip#=0; data=0	1
19	Spacer	CAFEFAD		1
20	Hybrid #1		Slave #; module#=0; strip#=0; data=0	1
21	Spacer	CAFEFAD		1
22	Hybrid #2		Slave #; module#=0; strip#=0; data=0	1
23	Spacer	CAFEFAD		1
24	Hybrid #3		Slave #; module#=0; strip#=0; data=0	1
25	Spacer	CAFEFAD		1
26	Hybrid #4		Slave #; module#=0; strip#=0; data=0	1
27	Spacer	CAFEFAD		1
28	Hybrid #5		Slave #; module#=0; strip#=0; data=0	1
29	Spacer	CAFEFAD		1
30	Hybrid #6		Slave #; module#=0; strip#=0; data=0	1
31	Spacer	CAFEFAD		1
32	Hybrid #7		Slave #; module#=0; strip#=0; data=0	1
33	Spacer	CAFEFAD		1
34	Hybrid #8		Slave #; module#=0; strip#=0; data=0	1
35	Spacer	CAFEFAD		1
36	Hybrid #9		Slave #; module#=0; strip#=0; data=0	1
37	Spacer	CAFEFAD		1
38	Hybrid #10		Slave #; module#=0; strip#=0; data=0	1
39	Spacer	CAFEFAD		1
40	Hybrid #11		Slave #; module#=0; strip#=0; data=0	1
41	Spacer	CAFEFAD		1
42	Hybrid #12		Slave #; module#=0; strip#=0; data=0	1
43	Spacer	CAFEFAD		1
44	Hybrid #13		Slave #; module#=0; strip#=0; data=0	1
45	Spacer	CAFEFAD		1
46	Hybrid #14		Slave #; module#=0; strip#=0; data=0	1
47	Spacer	CAFEFAD		1
48	Hybrid #15		Slave #; module#=0; strip#=0; data=0	1
49	Spacer	CAFEFAD		1

Table 22: Raw Data organization RDO-slave block when slave is not used**4.7.3 rdoU SIU daughter card^{[6][7]}**

Christophe-M.-A. Renard, Subatech, IN2P3/CNRS-L'UNAM, Nantes, F44307, France

Micheal-J. LeVine, BNL, Upton, NY, USA

Stéphane Bouvier, Subatech, IN2P3/CNRS-L'UNAM, Nantes, F44307, France

4.7.3.1 rdoU SIU connector

4.7.3.2 rdoU SIU protocol

It is copied from the one used for Alice-Dimuon-Trigger

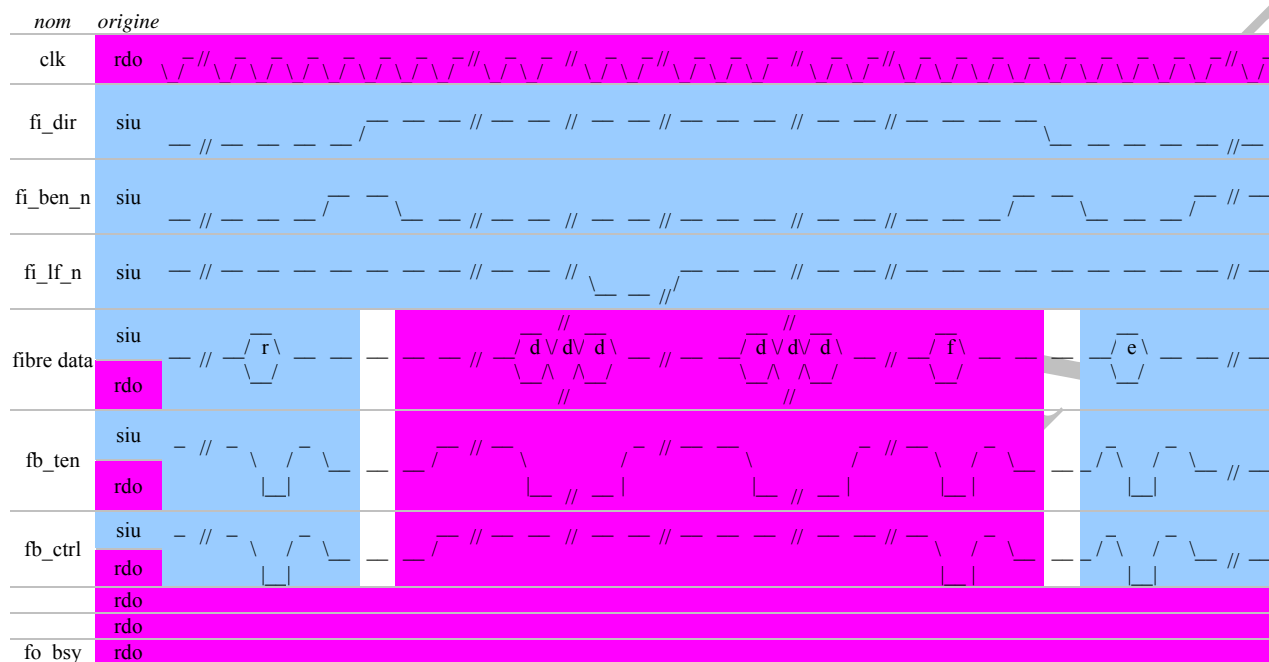


Table 23: SIU event transaction

r = RDYRX; d = data; f = FEFINW; e = EOBTR

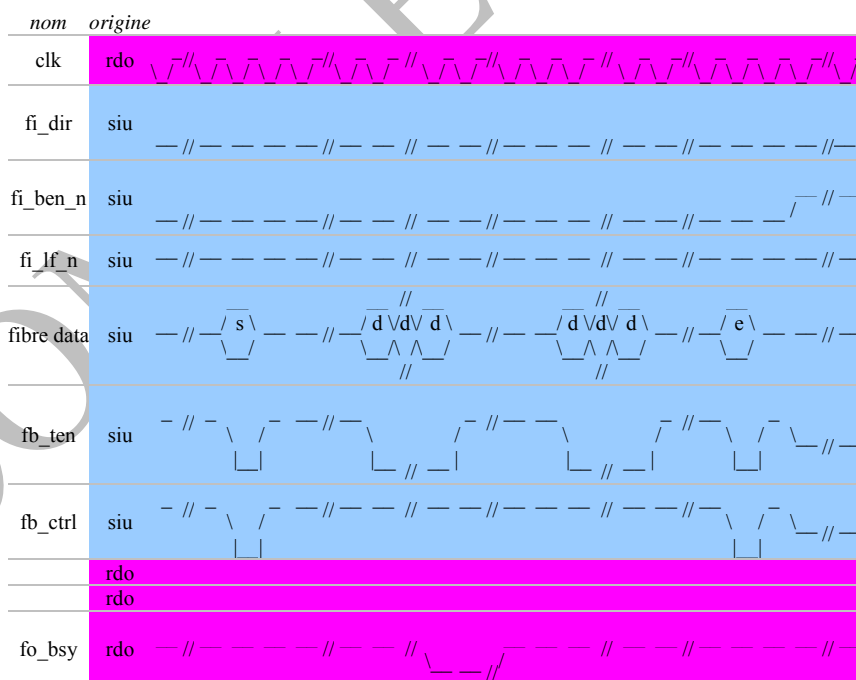


Table 24: SIU write pedestal transaction

s = STBWR; d = data; e = EOBTR

1 Do not use bit	19 parameter bits	4 identifier bits	4 code bits				4 destination bits				Command (DAQ to RDO)
X	FEE address	Transaction ID	write=1 read=0	User=1 star=0	Close=1 open=0	Block=1 single=0	JTAG	FEE	SIU	DIU	
X	Not used	Transaction ID	0	0	0	1	0	1	0	0	RDYRX (SIU ready for events)
X	Not used	Transaction ID	1	0	1	1	0	1	0	0	EOBTR (SIU refuses any new data event)
X	Address in FEE	Transaction ID	0	1	0	0	0	1	0	0	FESTRD (FEE status read)
X	Address in FEE (slave # + start addr)	Transaction ID	1	1	0	1	0	1	0	0	STBWR (Start Block WRite)

Table 25: SIU control words (commands from SIU to RDO)

2 steering bits	3 slave # bits	14 address bits
Steering bits	Slave #	Start address

Table 26: SIU control words (parameter field in STBWR command)

20 free bits	1 last data bit	1 increment bit	10 pedestal bits
DAQ free	last data	auto increment	pedestal

Table 27: SIU data word (set pedestal value)

1 error bit	19 status parameter bits	4 identifier bits	4 code bits				4 source bits				Answer (RDO to DAQ)
Error=1 ok=0	Status parameter	Transaction ID	write=1 read=0	User=1 star=0	Close=1 open=0	Block=1 single=0	JTAG	FEE	SIU	DIU	
0	Address in FEE copied from FESTRD	Transaction ID copied from FESTRD	0	1	0	0	0	1	0	0	FESTW (RDO status word)
0	Address in FEE copied from RDYRX	Transaction ID copied from RDYRX	0	1	1	0	0	1	0	0	FEFINW (event end)

Table 28: SIU control words (status from RDO to SIU)

4.8 rdoU Slow-Control interface

4.8.1 Overview

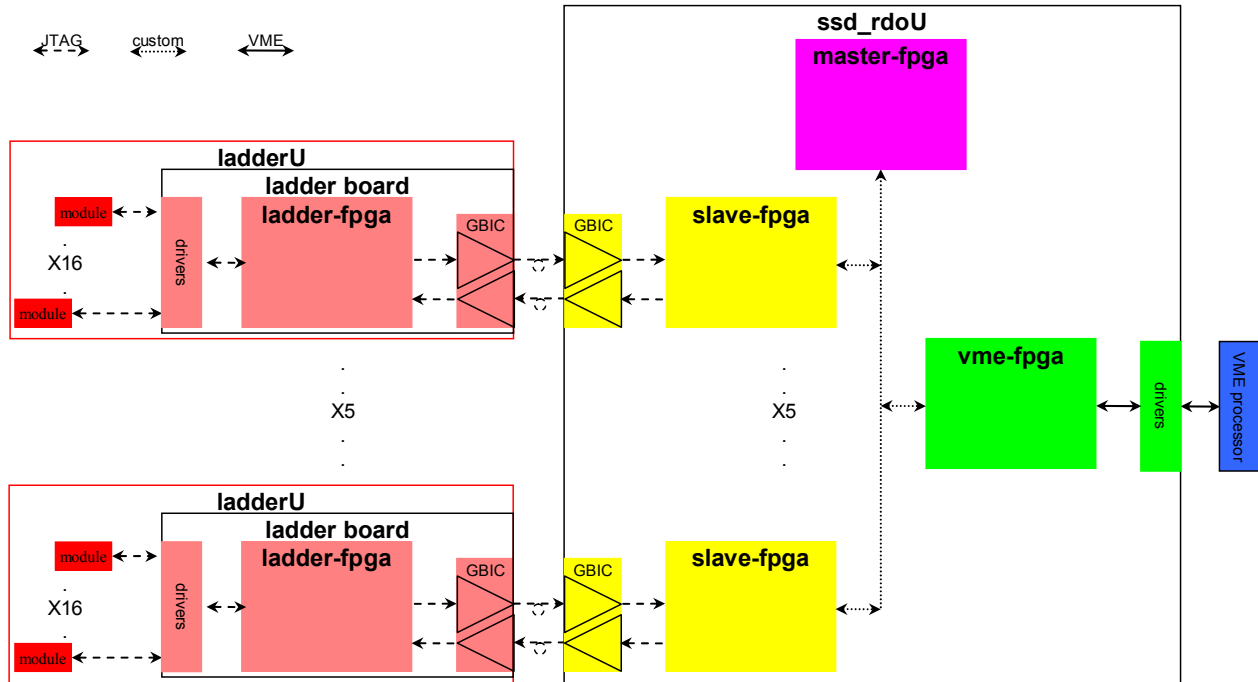


Figure 14: rdoU Slow-Control

4.8.2 VME

VME registers and addresses in A32/D32 mode

FPGA		nbr word	used	avail	addr	31 downto 12	11	10	9	8	7	6	5	4	3	2	1	0	comment
vme	internal registers	5	5	16	card address	0	0	0	0	0	0	0			register #		0	0	
vme	special registers	0	0	16	card address	0	0	0	0	0	0	1			register #		0	0	
trg	internal registers	14	4+10	16	card address	0	0	1	0	0	0	0			register #		0	0	
trg	special registers	1025	2	16	card address	0	0	1	0	0	0	1			register #		0	0	
trg	debug registers	?	32	32	r- card address	0	0	1	0	1					register #		0	0	
acq	internal registers	14	4+10	16	card address	0	1	0	0	0	0	0			register #		0	0	
acq	special registers	7	7	16	card address	0	1	0	0	0	0	1			register #		0	0	
slave 0	internal registers	14	4+10	16	card address	0	1	1	0	0	0	0			register #		0	0	
slave 0	special registers	4	4	16	card address	0	1	1	0	0	0	1			register #		0	0	
slave 0	debug registers	4+12288+68+16384+5	12	16	card address	0	1	1	0	1	0				register #		0	0	
slave 1	internal registers	14	4+10	16	card address	1	0	0	0	0	0	0			register #		0	0	
slave 1	special registers	4	4	16	card address	1	0	0	0	0	0	1			register #		0	0	
slave 1	debug registers	4+12288+68+16384+5	12	16	card address	1	0	0	0	1	0				register #		0	0	
slave 2	internal registers	14	4+10	16	card address	1	0	1	0	0	0	0			register #		0	0	
slave 2	special registers	4	4	16	card address	1	0	1	0	0	0	1			register #		0	0	
slave 2	debug registers	4+12288+68+16384+5	12	16	card address	1	0	1	0	1	0				register #		0	0	
slave 3	internal registers	14	4+10	16	card address	1	1	0	0	0	0	0			register #		0	0	
slave 3	special registers	4	4	16	card address	1	1	0	0	0	0	1			register #		0	0	
slave 3	debug registers	4+12288+68+16384+5	12	16	card address	1	1	0	0	1	0				register #		0	0	
slave 4	internal registers	14	4+10	16	card address	1	1	1	0	0	0	0			register #		0	0	
slave 4	special registers	4	4	16	card address	1	1	1	0	0	0	1			register #		0	0	
slave 4	debug registers	4+12288+68+16384+5	12	16	card address	1	1	1	0	1	0				register #		0	0	

Table 29: rdoU VME addresses

VME registers and addresses in A32/D32 mode																	
		addr				###	512	256	128	64	32	16	8	4	2		
FPGA		nbr word	used	avail	31 downto 12	11	10	9	8	7	6	5	4	3	2	1 0	comment
internal registers in all FPGAs																	
fpga	ID	1	1	1	r-	card address	fpga addr	0	0	0	0	0	0	0	0	0	
fpga	Date	1	1	1	r-	card address	fpga addr	0	0	0	0	0	0	0	1	0	
fpga	Status	1	1	1	r-	card address	fpga addr	0	0	0	0	0	0	1	0	0	
fpga	Config	1	1	1	rw	card address	fpga addr	0	0	0	0	0	0	1	1	0	
more internal registers in VME FPGA																	
vme	roboclock phase	1	1	1	rw	card address	fpga addr	0	0	0	0	0	1	0	0	0	
more internal registers in all FPGAs except VME																	
fpga	nbr trg phy	1	1	1	r-	card address	fpga addr	0	0	0	0	0	1	0	0	0	
fpga	nbr trg cal	1	1	1	r-	card address	fpga addr	0	0	0	0	0	1	0	1	0	
fpga	nbr l2 acq	1	1	1	r-	card address	fpga addr	0	0	0	0	0	1	1	0	0	
fpga	nbr l2 rej	1	1	1	r-	card address	fpga addr	0	0	0	0	0	1	1	1	0	
fpga	nbr trg mode 0	1	1	1	r-	card address	fpga addr	0	0	0	0	1	0	0	0	0	
fpga	nbr trg mode 1	1	1	1	r-	card address	fpga addr	0	0	0	0	1	0	0	1	0	
fpga	nbr trg mode 2	1	1	1	r-	card address	fpga addr	0	0	0	0	1	0	1	0	0	
fpga	nbr trg mode 3	1	1	1	r-	card address	fpga addr	0	0	0	0	1	0	1	1	0	
fpga	nbr rhic clk	1	1	1	r-	card address	fpga addr	0	0	0	0	1	1	0	0	0	
fpga	dead time	1	1	1	r-	card address	fpga addr	0	0	0	0	1	1	0	1	0	nbr rhic clk while busy
special registers in slave FPGAs																	
slave	jtag_ladderSC_cmd_reg	1	1	1	rw	card address	slave addr	0	0	1	0	0	0	0	0	0	
slave	jtag_ladderSC_clk	1	1	1	r-	card address	slave addr	0	0	1	0	0	0	0	1	0	
slave	FPGA_config_cmd	1	1	1	-w	card address	slave addr	0	0	1	0	0	0	1	0	0	
slave	FPGA_config_status	1	1	1	-w	card address	slave addr	0	0	1	0	0	0	1	1	0	
special registers in master FPGA (trig part)																	
trg	trg state machine	1	1	1	r-	card address	0	0	1	0	0	0	1	0	0	0	
trg	delay_before_hold	1	1	1	rw	card address	0	0	1	0	0	0	1	0	0	0	calib_hold(15.0) trig_hold(15.0)
trg	common threshold	1	1	1	rw	card address	0	0	1	0	0	0	1	0	0	0	x0000 threshold(15.0)
trg	trigger input mem	1024	1024	1	r-	card address	0	0	1	0	0	1	0	0	1	1	0xC+28b RhicClk or 0x100+20b trg_in
trg	status_event_buffer0	1	1	1	r-	card address	0	0	1	0	0	1	0	0	0	0	trg_status(1.0) + slave_status(4.0)(1.0)+20b L0mess
trg	status_event_buffer1	1	1	1	r-	card address	0	0	1	0	0	1	0	0	1	0	trg_status(1.0) + slave_status(4.0)(1.0)+20b L0mess
trg	status_event_buffer2	1	1	1	r-	card address	0	0	1	0	0	1	0	0	1	0	trg_status(1.0) + slave_status(4.0)(1.0)+20b L0mess
trg	status_event_buffer3	1	1	1	r-	card address	0	0	1	0	0	1	0	0	1	1	trg_status(1.0) + slave_status(4.0)(1.0)+20b L0mess
trg	status_master_mssi0	1	1	1	r-	card address	0	0	1	0	0	1	1	0	0	0	x00+slav_stat+mssi_stat+L2stat+L2com+L0stat+L0com
trg	status_master_mssi1	1	1	1	r-	card address	0	0	1	0	0	1	1	0	0	1	x00+slav_stat+mssi_stat+L2stat+L2com+L0stat+L0com
trg	status_master_mssi2	1	1	1	r-	card address	0	0	1	0	0	1	1	0	0	1	x00+slav_stat+mssi_stat+L2stat+L2com+L0stat+L0com
trg	status_master_mssi3	1	1	1	r-	card address	0	0	1	0	0	1	1	0	0	1	x00+slav_stat+mssi_stat+L2stat+L2com+L0stat+L0com
trg	status_master_mssi4	1	1	1	r-	card address	0	0	1	0	0	1	1	1	0	0	x00+slav_stat+mssi_stat+L2stat+L2com+L0stat+L0com
special registers in master FPGA (acq part)																	
acq	nbr rdyrx	1	1	1	r-	card address	0	1	0	0	0	0	0	0	0	0	
acq	nbr eobtr	1	1	1	r-	card address	0	1	0	0	0	0	1	0	0	0	
acq	nbr l2acc no daq	1	1	1	r-	card address	0	1	0	0	0	0	1	0	0	0	
acq	nbr festrd	1	1	1	r-	card address	0	1	0	0	0	0	1	0	0	0	
acq	nbr link full	1	1	1	r-	card address	0	1	0	0	0	0	1	0	0	0	
acq	duration link full	1	1	1	r-	card address	0	1	0	0	0	0	1	0	1	0	nbr rhic clk while link full on
acq	acq state machine	1	1	1	r-	card address	0	1	0	0	0	0	1	0	0	0	
debug registers in slave FPGAs																	
slave	rd_buff_no_reg	1	1	1	rw	card address	slave addr	0	1	0	0	0	0	0	0	0	
slave	wr_buff_no_reg	1	1	1	rw	card address	slave addr	0	1	0	0	0	0	0	1	0	
slave	mem_start_addr_reg	1	1	1	rw	card address	slave addr	0	1	0	0	0	0	0	1	0	
slave	adc_mask_reg	1	1	1	rw	card address	slave addr	0	1	0	0	0	0	0	1	1	
slave	pedestal_reg	12288	16*1024	1	rw	card address	slave addr	0	1	0	0	0	1	0	0	0	16 hybrids x 768 words (12288)
slave	hdr_buffer_reg	68	64	1	r-	card address	slave addr	0	1	0	0	0	1	0	1	0	4 header buffers x 17 words (68)
slave	event_buffer_reg	16384	16*1024	1	rw	card address	slave addr	0	1	0	0	0	1	1	0	0	4 event buffers x 16 hybrides x 256 words (16384)
slave	test_pulse_reg	1	1	1	rw	card address	slave addr	0	1	0	0	0	1	1	1	0	
slave	set_hold_reg	1	1	1	rw	card address	slave addr	0	1	0	0	1	0	0	0	0	
slave	abort_event_reg	1	1	1	rw	card address	slave addr	0	1	0	0	1	0	0	1	0	
slave	reset_fpga_reg	1	1	1	rw	card address	slave addr	0	1	0	0	1	0	0	1	0	
slave	echo_usb_reg	1	1	1	rw	card address	slave addr	0	1	0	0	1	0	0	1	1	only in usb mode

Table 30: rdoU VME registers

4.9 rdoU-VME-FPGA

4.9.1 Overview

The firmware for this ALTERA^[8] CycloneIII EP3C16F484C6^[9] FPGA was developed in VHDL using Synplify_pro^[10] as a synthesizer, Quartus II^[11] as place & route software and ModelSim-Altera^[12] as a simulation software. Complete project can be found in [0]

VME-FPGA provides a 6 bit wide local address bus, corresponding to VME address bits 2 to 8.

<i>name</i>		<i>addr</i>	4 bits	4 bits	4 bits	4 bits	4 bits	4 bits	4 bits	4 bits
<i>ID</i>	<i>r-</i>	0	application	FPGA ID (vme=1)	FPGA version		card #	card serial	card version	card type (rdoUv0=7)
<i>date</i>	<i>r-</i>	4	day		month		year			
<i>Status</i>	<i>r-</i>	8	free	free	free	free	free	free	free	free
<i>config</i>	<i>rw</i>	C	free	free	free	free	free	free	free	free free free soft reset
<i>Roboclock phase</i> ^[1]	<i>rw</i>	10	free	free	free	free	RDO roboclock 36.4MHz phase			

Table 31: rdoU VME-FPGA VME registers

Clock36MHz-slv4 and clock36MHz-slv3 phase	rdo_sc_roboclock_phase(3 downto 2)			
	11	10	01	00
rdo_sc_roboclock_phase(1 downto 0)	/2	+1,7ns	+1,7ns	-3,4ns
	+5,2ns	0	0	-5,2ns
	+5,2ns	0	0	-5,2ns
	+3,4ns	-1,7ns	-1,7ns	inv

Table 32: ladder-FPGA slow-control register (roboclock_phase)^[1] effect on Clock36MHz-slv4 and clock36MHz-slv3

Clock36MHz-slv2 and clock36MHz-slv1 phase	rdo_sc_roboclock_phase(7 downto 6)			
	11	10	01	00
rdo_sc_roboclock_phase(5 downto 4)	/2	+1,7ns	+1,7ns	-3,4ns
	+5,2ns	0	0	-5,2ns
	+5,2ns	0	0	-5,2ns
	+3,4ns	-1,7ns	-1,7ns	/4

Table 33: ladder-FPGA slow-control register (roboclock_phase)^[1] effect on Clock36MHz-slv2 and clock36MHz-slv1

Feed-back and clock36MHz-slv0 phase	rdo_sc_roboclock_phase(11 downto 10)			
	11	10	01	00
rdo_sc_roboclock_phase(9 downto 8)	+3,4ns	+0,9ns	+0,9ns	-1,7ns
	+2,6ns	0	0	-2,6ns
	+2,6ns	0	0	-2,6ns
	+1,7ns	-0,9ns	-0,9ns	-3,4ns

Table 34: ladder-FPGA slow-control register (roboclock_phase)^[1] effect on feed-back and clock36MHz-slv0
feed-back input (FB) connected to output 2Q1

Clock36MHz-vme FPGA phase	rdo_sc_roboclock_phase(15 downto 14)			
	11	10	01	00
rdo_sc_roboclock_phase(13 downto 12)	+3,4ns	+0,9ns	+0,9ns	-1,7ns
	+2,6ns	0	0	-2,6ns
	+2,6ns	0	0	-2,6ns
	+1,7ns	-0,9ns	-0,9ns	-3,4ns

Table 35: ladder-FPGA slow-control register (roboclock_phase)^[1] effect on Clock36MHz-vme FPGA

Copyright (C) 1991-2011 Altera Corporation
Your use of Altera Corporation's design tools logic functions
and other software and tools and its AMPP partner logic
functions and any output files from any of the foregoing
(including device programming or simulation files) and any
associated documentation or information are expressly subject
to the terms and conditions of the Altera Program License
Subscription Agreement Altera MegaCore Function License

# Agreement or other applicable license agreement including							
# without limitation that your use is for the sole purpose of							
# programming logic devices manufactured by Altera and sold by							
# Altera or its authorized distributors. Please refer to the							
# applicable agreement for further details.							
# Quartus II Version 10.1 Build 197 01/19/2011 Service Pack 1 SJ Full Version							
# CHIP "rdoU_fpga_vme" ASSIGNED TO AN: EP3C16F486C6							
# File: C:/SSD upgrade/New VME FPGA/rev_1/pr_1/rdoU_fpga_vme.csv							
# Generated on: Thu Mar 17 14:47:45 2011							
Annotations as to the function of each signal group added -- MJL/ChR							
# Note: The column header names should not be changed if you wish to import this .csv file into the Quartus II software.							
To	Direction	Location	I/O Bank	VREF Group	I/O Standard	Reserved	comment
card_addr<31>	Input	PIN_L16	6B6_N1	2.5 V			from on-board address switches
card_addr<30>	Input	PIN_U22	5B5_N0	2.5 V			from on-board address switches
card_addr<29>	Input	PIN_U21	5B5_N0	2.5 V			from on-board address switches
card_addr<28>	Input	PIN_C22	6B6_N0	2.5 V			from on-board address switches
card_addr<27>	Input	PIN_AB19	4B4_N0	2.5 V			from on-board address switches
card_addr<26>	Input	PIN_N1	2B2_N0	2.5 V			from on-board address switches
card_addr<25>	Input	PIN_B17	7B7_N1	2.5 V			from on-board address switches
card_addr<24>	Input	PIN_A15	7B7_N1	2.5 V			from on-board address switches
card_addr<23>	Input	PIN_R20	5B5_N1	2.5 V			from on-board address switches
card_addr<22>	Input	PIN_E16	7B7_N0	2.5 V			from on-board address switches
card_addr<21>	Input	PIN_M2	2B2_N0	2.5 V			from on-board address switches
card_addr<20>	Input	PIN_B16	7B7_N1	2.5 V			from on-board address switches
card_addr<19>	Input	PIN_M22	5B5_N0	2.5 V			from on-board address switches
card_addr<18>	Input	PIN_U13	4B4_N0	2.5 V			from on-board address switches
card_addr<17>	Input	PIN_Y13	4B4_N1	2.5 V			from on-board address switches
card_addr<16>	Input	PIN_A16	7B7_N1	2.5 V			from on-board address switches
card_addr<15>	Input	PIN_J1	1B1_N1	2.5 V			from on-board address switches
card_addr<14>	Input	PIN_G12	7B7_N1	2.5 V			from on-board address switches
card_addr<13>	Input	PIN_E22	6B6_N0	2.5 V			from on-board address switches
card_addr<12>	Input	PIN_F11	7B7_N1	2.5 V			from on-board address switches
card_clk_vme	Input	PIN_G2	1B1_N1	2.5 V			from Roboclock output 2Q1 (2Q0 used for Feed-Back)
card_daq_num<3>	Input	PIN_P1	2B2_N0	2.5 V			RDO number: from master FPGA
card_daq_num<2>	Input	PIN_AB13	4B4_N1	2.5 V			RDO number: from master FPGA
card_daq_num<1>	Input	PIN_E9	8B8_N0	2.5 V			RDO number: from master FPGA
card_daq_num<0>	Input	PIN_P4	2B2_N0	2.5 V			RDO number: from master FPGA
card_data_ack_n	Output	PIN_L7	2B2_N0	2.5 V			to VME bus
card_ser_num<3>	Input	PIN_D2	1B1_N0	2.5 V			serial # from on-board jumpers
card_ser_num<2>	Input	PIN_T12	4B4_N1	2.5 V			serial # from on-board jumpers
card_ser_num<1>	Input	PIN_N6	2B2_N0	2.5 V			serial # from on-board jumpers
card_ser_num<0>	Input	PIN_L6	2B2_N0	2.5 V			serial # from on-board jumpers
card_ver<3>	Input	PIN_L21	6B6_N1	2.5 V			version # from on-board jumpers
card_ver<2>	Input	PIN_V8	3B3_N0	2.5 V			version # from on-board jumpers
card_ver<1>	Input	PIN_V14	4B4_N0	2.5 V			version # from on-board jumpers
card_ver<0>	Input	PIN_D10	8B8_N0	2.5 V			version # from on-board jumpers
led_vme_card_acces_n	Output	PIN_P3	2B2_N0	2.5 V			FP LED: card addressed via VME
reset_n	Input	PIN_T1	2B2_N0	2.5 V			from power supervisor that includes inputs from "FP reset switch", "BP sysreset" and "soft reset"
roboclock_rdo_1F<1>	Output	PIN_M6	2B2_N0	2.5 V			Roboclock control input
roboclock_rdo_1F<0>	Output	PIN_J4	1B1_N1	2.5 V			Roboclock control input
roboclock_rdo_2F<1>	Output	PIN_AA18	4B4_N0	2.5 V			Roboclock control input
roboclock_rdo_2F<0>	Output	PIN_C21	6B6_N0	2.5 V			Roboclock control input
roboclock_rdo_3F<1>	Output	PIN_AB10	3B3_N0	2.5 V			Roboclock control input
roboclock_rdo_3F<0>	Output	PIN_P15	5B5_N1	2.5 V			Roboclock control input
roboclock_rdo_4F<1>	Output	PIN_E15	7B7_N0	2.5 V			Roboclock control input
roboclock_rdo_4F<0>	Output	PIN_AA3	3B3_N1	2.5 V			Roboclock control input
usb_clk	Input	PIN_G1	1B1_N1	2.5 V			Is this OSCO on 245R block diagram? should be debug output (to FP LED ? or test pin ?)
usb_data<7>	Bidir	PIN_F20	6B6_N0	2.5 V			from FTDI 245R
usb_data<6>	Bidir	PIN_J17	6B6_N0	2.5 V			from FTDI 245R
usb_data<5>	Bidir	PIN_G17	6B6_N0	2.5 V			from FTDI 245R
usb_data<4>	Bidir	PIN_F21	6B6_N0	2.5 V			from FTDI 245R
usb_data<3>	Bidir	PIN_G15	7B7_N0	2.5 V			from FTDI 245R
usb_data<2>	Bidir	PIN_D20	6B6_N0	2.5 V			from FTDI 245R
usb_data<1>	Bidir	PIN_H16	6B6_N0	2.5 V			from FTDI 245R
usb_data<0>	Bidir	PIN_F22	6B6_N1	2.5 V			from FTDI 245R
usb_read_n	Output	PIN_K18	6B6_N1	2.5 V			from FTDI 245R

usb_ready_n	Input	PIN_M19	5B5_NO	2.5 V	from FTDI 245R (PWREN#)
usb_reset_n	Input	PIN_V15	4B4_NO	2.5 V	from FTDI 245R
usb_rx_empty	Input	PIN_H21	6B6_N1	2.5 V	from FTDI 245R
usb_tx_full	Input	PIN_F2	1B1_NO	2.5 V	from FTDI 245R
usb_write_n	Output	PIN_D22	6B6_NO	2.5 V	from FTDI 245R
vme_acq_access_n	Output	PIN_R18	5B5_NO	2.5 V	to master FPGA
vme_addr_01	Input	PIN_A12	7B7_N1	2.5 V	from VME bus address mod buffer
vme_addr<31>	Input	PIN_B12	7B7_N1	2.5 V	from VME bus address buffer
vme_addr<30>	Input	PIN_G3	1B1_NO	2.5 V	from VME bus address buffer
vme_addr<29>	Input	PIN_N22	5B5_NO	2.5 V	from VME bus address buffer
vme_addr<28>	Input	PIN_G22	6B6_N1	2.5 V	from VME bus address buffer
vme_addr<27>	Input	PIN_G21	6B6_N1	2.5 V	from VME bus address buffer
vme_addr<26>	Input	PIN_AB17	4B4_N1	2.5 V	from VME bus address buffer
vme_addr<25>	Input	PIN_N2	2B2_NO	2.5 V	from VME bus address buffer
vme_addr<24>	Input	PIN_J15	6B6_N1	2.5 V	from VME bus address buffer
vme_addr<23>	Input	PIN_U15	4B4_NO	2.5 V	from VME bus address buffer
vme_addr<22>	Input	PIN_K21	6B6_N1	2.5 V	from VME bus address buffer
vme_addr<21>	Input	PIN_F12	7B7_N1	2.5 V	from VME bus address buffer
vme_addr<20>	Input	PIN_E21	6B6_NO	2.5 V	from VME bus address buffer
vme_addr<19>	Input	PIN_N21	5B5_NO	2.5 V	from VME bus address buffer
vme_addr<18>	Input	PIN_J7	1B1_N1	2.5 V	from VME bus address buffer
vme_addr<17>	Input	PIN_A17	7B7_N1	2.5 V	from VME bus address buffer
vme_addr<16>	Input	PIN_F17	6B6_NO	2.5 V	from VME bus address buffer
vme_addr<15>	Input	PIN_N14	5B5_N1	2.5 V	from VME bus address buffer
vme_addr<14>	Input	PIN_A5	8B8_NO	2.5 V	from VME bus address buffer
vme_addr<13>	Input	PIN_AB15	4B4_N1	2.5 V	from VME bus address buffer
vme_addr<12>	Input	PIN_AA7	3B3_NO	2.5 V	from VME bus address buffer
vme_addr<11>	Input	PIN_N20	5B5_NO	2.5 V	from VME bus address buffer
vme_addr<10>	Input	PIN_N17	5B5_NO	2.5 V	from VME bus address buffer
vme_addr<9>	Input	PIN_W2	2B2_N1	2.5 V	from VME bus address buffer
vme_addr<8>	Input	PIN_F7	8B8_N1	2.5 V	from VME bus address buffer
vme_addr<7>	Input	PIN_V4	2B2_N1	2.5 V	from VME bus address buffer
vme_addr<6>	Input	PIN_C1	1B1_NO	2.5 V	from VME bus address buffer
vme_addr<5>	Input	PIN_N5	2B2_NO	2.5 V	from VME bus address buffer
vme_addr<4>	Input	PIN_AA20	4B4_NO	2.5 V	from VME bus address buffer
vme_addr<3>	Input	PIN_Y21	5B5_N1	2.5 V	from VME bus address buffer
vme_addr<2>	Input	PIN_P5	2B2_NO	2.5 V	from VME bus address buffer
vme_addr_modif<5>	Input	PIN_E14	7B7_N1	2.5 V	from VME bus address mod buffer
vme_addr_modif<4>	Input	PIN_AA16	4B4_N1	2.5 V	from VME bus address mod buffer
vme_addr_modif<3>	Input	PIN_H17	6B6_NO	2.5 V	from VME bus address mod buffer
vme_addr_modif<2>	Input	PIN_B4	8B8_N1	2.5 V	from VME bus address mod buffer
vme_addr_modif<1>	Input	PIN_P16	5B5_N1	2.5 V	from VME bus address mod buffer
vme_addr_modif<0>	Input	PIN_T16	4B4_NO	2.5 V	from VME bus address mod buffer
vme_addr_strobe_n	Input	PIN_M16	5B5_NO	2.5 V	from VME bus address mod buffer
vme_card_access_n	Output	PIN_U2	2B2_NO	2.5 V	Redundant with each vme_xxx_access_n and led_vme_card_access_n. May not be needed in FPGAs
vme_card_addr<14>	Output	PIN_H20	6B6_NO	2.5 V	not used
vme_card_addr<13>	Output	PIN_V22	5B5_N1	2.5 V	not used
vme_card_addr<12>	Output	PIN_H9	8B8_NO	2.5 V	not used
vme_card_addr<11>	Output	PIN_AA15	4B4_N1	2.5 V	not used
vme_card_addr<10>	Output	PIN_AB8	3B3_NO	2.5 V	not used
vme_card_addr<9>	Output	PIN_P22	5B5_NO	2.5 V	not used
vme_card_addr<8>	Output	PIN_P21	5B5_NO	2.5 V	not used
vme_card_addr<7>	Output	PIN_V1	2B2_NO	2.5 V	not used
vme_card_addr<6>	Output	PIN_E5	8B8_N1	2.5 V	to all slaves, master FPGA
vme_card_addr<5>	Output	PIN_AA2	2B2_N1	2.5 V	to all slaves, master FPGA
vme_card_addr<4>	Output	PIN_E4	1B1_NO	2.5 V	to all slaves, master FPGA
vme_card_addr<3>	Output	PIN_U1	2B2_NO	2.5 V	to all slaves, master FPGA
vme_card_addr<2>	Output	PIN_W17	4B4_NO	2.5 V	to all slaves, master FPGA
vme_card_addr<1>	Output	PIN_W21	5B5_N1	2.5 V	to all slaves, master FPGA
vme_card_addr<0>	Output	PIN_M7	2B2_NO	2.5 V	to all slaves, master FPGA
vme_card_data<31>	Bidir	PIN_C20	6B6_NO	2.5 V	to all slaves, master FPGA
vme_card_data<30>	Bidir	PIN_B18	7B7_NO	2.5 V	to all slaves, master FPGA
vme_card_data<29>	Bidir	PIN_E11	7B7_N1	2.5 V	to all slaves, master FPGA
vme_card_data<28>	Bidir	PIN_K16	6B6_N1	2.5 V	to all slaves, master FPGA
vme_card_data<27>	Bidir	PIN_B6	8B8_NO	2.5 V	to all slaves, master FPGA
vme_card_data<26>	Bidir	PIN_B7	8B8_NO	2.5 V	to all slaves, master FPGA
vme_card_data<25>	Bidir	PIN_AB18	4B4_NO	2.5 V	to all slaves, master FPGA
vme_card_data<24>	Bidir	PIN_H15	7B7_NO	2.5 V	to all slaves, master FPGA

vme_card_data<23>	Bidir	PIN_C17	7B7_NO	2.5 V	to all slaves, master FPGA
vme_card_data<22>	Bidir	PIN_J16	6B6_N1	2.5 V	to all slaves, master FPGA
vme_card_data<21>	Bidir	PIN_B20	7B7_NO	2.5 V	to all slaves, master FPGA
vme_card_data<20>	Bidir	PIN_F19	6B6_NO	2.5 V	to all slaves, master FPGA
vme_card_data<19>	Bidir	PIN_W15	4B4_NO	2.5 V	to all slaves, master FPGA
vme_card_data<18>	Bidir	PIN_N16	5B5_NO	2.5 V	to all slaves, master FPGA
vme_card_data<17>	Bidir	PIN_F1	1B1_NO	2.5 V	to all slaves, master FPGA
vme_card_data<16>	Bidir	PIN_W14	4B4_N1	2.5 V	to all slaves, master FPGA
vme_card_data<15>	Bidir	PIN_F9	8B8_N1	2.5 V	to all slaves, master FPGA
vme_card_data<14>	Bidir	PIN_B14	7B7_N1	2.5 V	to all slaves, master FPGA
vme_card_data<13>	Bidir	PIN_A8	8B8_NO	2.5 V	to all slaves, master FPGA
vme_card_data<12>	Bidir	PIN_B9	8B8_NO	2.5 V	to all slaves, master FPGA
vme_card_data<11>	Bidir	PIN_C13	7B7_N1	2.5 V	to all slaves, master FPGA
vme_card_data<10>	Bidir	PIN_G13	7B7_N1	2.5 V	to all slaves, master FPGA
vme_card_data<9>	Bidir	PIN_F13	7B7_N1	2.5 V	to all slaves, master FPGA
vme_card_data<8>	Bidir	PIN_V11	3B3_NO	2.5 V	to all slaves, master FPGA
vme_card_data<7>	Bidir	PIN_W1	2B2_N1	2.5 V	to all slaves, master FPGA
vme_card_data<6>	Bidir	PIN_AA5	3B3_N1	2.5 V	to all slaves, master FPGA
vme_card_data<5>	Bidir	PIN_J22	6B6_N1	2.5 V	to all slaves, master FPGA
vme_card_data<4>	Bidir	PIN_V12	4B4_N1	2.5 V	to all slaves, master FPGA
vme_card_data<3>	Bidir	PIN_AB7	3B3_NO	2.5 V	to all slaves, master FPGA
vme_card_data<2>	Bidir	PIN_V16	4B4_NO	2.5 V	to all slaves, master FPGA
vme_card_data<1>	Bidir	PIN_E12	7B7_N1	2.5 V	to all slaves, master FPGA
vme_card_data<0>	Bidir	PIN_J2	1B1_N1	2.5 V	to all slaves, master FPGA
vme_card_rd_wr_n	Output	PIN_H10	8B8_NO	2.5 V	
vme_card_read_n	Output	PIN_M15	5B5_N1	2.5 V	redundant. One could go to FPGAs and the other to FP LED.
vme_data<31>	Bidir	PIN_B22	6B6_NO	2.5 V	from VME bus
vme_data<30>	Bidir	PIN_A18	7B7_NO	2.5 V	from VME bus
vme_data<29>	Bidir	PIN_A7	8B8_NO	2.5 V	from VME bus
vme_data<28>	Bidir	PIN_G10	8B8_NO	2.5 V	from VME bus
vme_data<27>	Bidir	PIN_H11	8B8_NO	2.5 V	from VME bus
vme_data<26>	Bidir	PIN_E13	7B7_N1	2.5 V	from VME bus
vme_data<25>	Bidir	PIN_J18	6B6_N1	2.5 V	from VME bus
vme_data<24>	Bidir	PIN_H14	7B7_NO	2.5 V	from VME bus
vme_data<23>	Bidir	PIN_D17	7B7_NO	2.5 V	from VME bus
vme_data<22>	Bidir	PIN_A20	7B7_NO	2.5 V	from VME bus
vme_data<21>	Bidir	PIN_C19	7B7_NO	2.5 V	from VME bus
vme_data<20>	Bidir	PIN_D21	6B6_NO	2.5 V	from VME bus
vme_data<19>	Bidir	PIN_T14	4B4_NO	2.5 V	from VME bus
vme_data<18>	Bidir	PIN_V21	5B5_N1	2.5 V	from VME bus
vme_data<17>	Bidir	PIN_K7	1B1_N1	2.5 V	from VME bus
vme_data<16>	Bidir	PIN_V13	4B4_N1	2.5 V	from VME bus
vme_data<15>	Bidir	PIN_A13	7B7_N1	2.5 V	from VME bus
vme_data<14>	Bidir	PIN_A9	8B8_NO	2.5 V	from VME bus
vme_data<13>	Bidir	PIN_A10	8B8_NO	2.5 V	from VME bus
vme_data<12>	Bidir	PIN_E10	8B8_NO	2.5 V	from VME bus
vme_data<11>	Bidir	PIN_A6	8B8_NO	2.5 V	from VME bus
vme_data<10>	Bidir	PIN_C15	7B7_N1	2.5 V	from VME bus
vme_data<9>	Bidir	PIN_A14	7B7_N1	2.5 V	from VME bus
vme_data<8>	Bidir	PIN_U11	3B3_NO	2.5 V	from VME bus
vme_data<7>	Bidir	PIN_Y2	2B2_N1	2.5 V	from VME bus
vme_data<6>	Bidir	PIN_U10	3B3_NO	2.5 V	from VME bus
vme_data<5>	Bidir	PIN_J21	6B6_N1	2.5 V	from VME bus
vme_data<4>	Bidir	PIN_V9	3B3_NO	2.5 V	from VME bus
vme_data<3>	Bidir	PIN_Y10	3B3_NO	2.5 V	from VME bus
vme_data<2>	Bidir	PIN_L8	1B1_N1	2.5 V	from VME bus
vme_data<1>	Bidir	PIN_B13	7B7_N1	2.5 V	from VME bus
vme_data<0>	Bidir	PIN_H2	1B1_N1	2.5 V	from VME bus
vme_data_strobe_0_n	Input	PIN_V10	3B3_NO	2.5 V	from VME bus
vme_data_strobe_1_n	Input	PIN_K15	6B6_N1	2.5 V	from VME bus
vme_data_strobe_n	Output	PIN_Y17	4B4_NO	2.5 V	to all slaves, master FPGA
vme_fpga_test<31>	Output	PIN_P6	2B2_N1	2.5 V	for diagnostics, to test header
vme_fpga_test<30>	Output	PIN_AB5	3B3_N1	2.5 V	for diagnostics, to test header
vme_fpga_test<29>	Output	PIN_M1	2B2_NO	2.5 V	for diagnostics, to test header
vme_fpga_test<28>	Output	PIN_D15	7B7_NO	2.5 V	for diagnostics, to test header
vme_fpga_test<27>	Output	PIN_P2	2B2_NO	2.5 V	for diagnostics, to test header
vme_fpga_test<26>	Output	PIN_M4	2B2_NO	2.5 V	for diagnostics, to test header
vme_fpga_test<25>	Output	PIN_L15	6B6_N1	2.5 V	for diagnostics, to test header
vme_fpga_test<24>	Output	PIN_V2	2B2_NO	2.5 V	for diagnostics, to test header

vme_fpga_test<23>	Output	PIN_D19	7B7_N0	2.5 V	for diagnostics, to test header
vme_fpga_test<22>	Output	PIN_AA14	4B4_N1	2.5 V	for diagnostics, to test header
vme_fpga_test<21>	Output	PIN_D13	7B7_N1	2.5 V	for diagnostics, to test header
vme_fpga_test<20>	Output	PIN_H12	7B7_N1	2.5 V	for diagnostics, to test header
vme_fpga_test<19>	Output	PIN_U12	4B4_N1	2.5 V	for diagnostics, to test header
vme_fpga_test<18>	Output	PIN_AB16	4B4_N1	2.5 V	for diagnostics, to test header
vme_fpga_test<17>	Output	PIN_AA17	4B4_N1	2.5 V	for diagnostics, to test header
vme_fpga_test<16>	Output	PIN_P17	5B5_N1	2.5 V	for diagnostics, to test header
vme_fpga_test<15>	Output	PIN_U19	5B5_N1	2.5 V	for diagnostics, to test header
vme_fpga_test<14>	Output	PIN_R1	2B2_N0	2.5 V	for diagnostics, to test header
vme_fpga_test<13>	Output	PIN_M20	5B5_N0	2.5 V	for diagnostics, to test header
vme_fpga_test<12>	Output	PIN_L22	6B6_N1	2.5 V	for diagnostics, to test header
vme_fpga_test<11>	Output	PIN_P20	5B5_N0	2.5 V	for diagnostics, to test header
vme_fpga_test<10>	Output	PIN_R2	2B2_N0	2.5 V	for diagnostics, to test header
vme_fpga_test<9>	Output	PIN_R19	5B5_N0	2.5 V	for diagnostics, to test header
vme_fpga_test<8>	Output	PIN_AA19	4B4_N0	2.5 V	for diagnostics, to test header
vme_fpga_test<7>	Output	PIN_M21	5B5_N0	2.5 V	for diagnostics, to test header
vme_fpga_test<6>	Output	PIN_F14	7B7_N0	2.5 V	for diagnostics, to test header
vme_fpga_test<5>	Output	PIN_H19	6B6_N0	2.5 V	for diagnostics, to test header
vme_fpga_test<4>	Output	PIN_R13	4B4_N1	2.5 V	for diagnostics, to test header
vme_fpga_test<3>	Output	PIN_AB14	4B4_N1	2.5 V	for diagnostics, to test header
vme_fpga_test<2>	Output	PIN_K19	6B6_N1	2.5 V	for diagnostics, to test header
vme_fpga_test<1>	Output	PIN_W13	4B4_N1	2.5 V	for diagnostics, to test header
vme_fpga_test<0>	Output	PIN_T15	4B4_N0	2.5 V	for diagnostics, to test header
vme_int_ack_n	Input	PIN_N18	5B5_N0	2.5 V	from VME bus
vme_long_word	Input	PIN_M5	2B2_N0	2.5 V	from VME bus address mod buffer
vme_read_write_n	Input	PIN_H22	6B6_N1	2.5 V	from VME bus
vme_reset_card_n	Output	PIN_G9	8B8_N0	2.5 V	soft reset: to power supervisor
vme_slav0_acces_n	Output	PIN_AA9	3B3_N0	2.5 V	to slave FPGA 0
vme_slav1_acces_n	Output	PIN_R17	5B5_N1	2.5 V	to slave FPGA 1
vme_slav2_acces_n	Output	PIN_R22	5B5_N0	2.5 V	to slave FPGA 2
vme_slav3_acces_n	Output	PIN_N19	5B5_N0	2.5 V	to slave FPGA 3
vme_slav4_acces_n	Output	PIN_C10	8B8_N0	2.5 V	to slave FPGA 4
vme_sysclk	Input	PIN_T2	2B2_N0	2.5 V	from VME bus
vme_sysclk_present	Output	PIN_G4	1B1_N0	2.5 V	debug to (FP LED ? or Test pin ?)
vme_trg_acces_n	Output	PIN_R21	5B5_N0	2.5 V	to master FPGA
~ALTERA_ASDO_DATA1~	Input	PIN_D1	1B1_N0		
~ALTERA_FLASH_nCE_nCSO~	Input	PIN_E2	1B1_N0		
~ALTERA_DCLK~	Output	PIN_K2	1B1_N1		AS configuration: to serial memory
~ALTERA_DATA0~	Input	PIN_K1	1B1_N1		AS configuration: from serial memory
~ALTERA_nCEO~	Output	PIN_K22	6B6_N1		probably used only if more than one FPGA is configured by the same serial memory

Table 36: rdoU VME-FPGA pinout

4.10 rdoU-Master-FPGA

4.10.1 Overview

The firmware for this ALTERA^[8] CycloneIII EP3C40F484C6^[9] FPGA was developed in VHDL using Synplify_pro^[10] as a synthesizer, Quartus II^[11] as place & route software and ModelSim-Altera^[12] as a simulation software. Complete project can be found in [0]

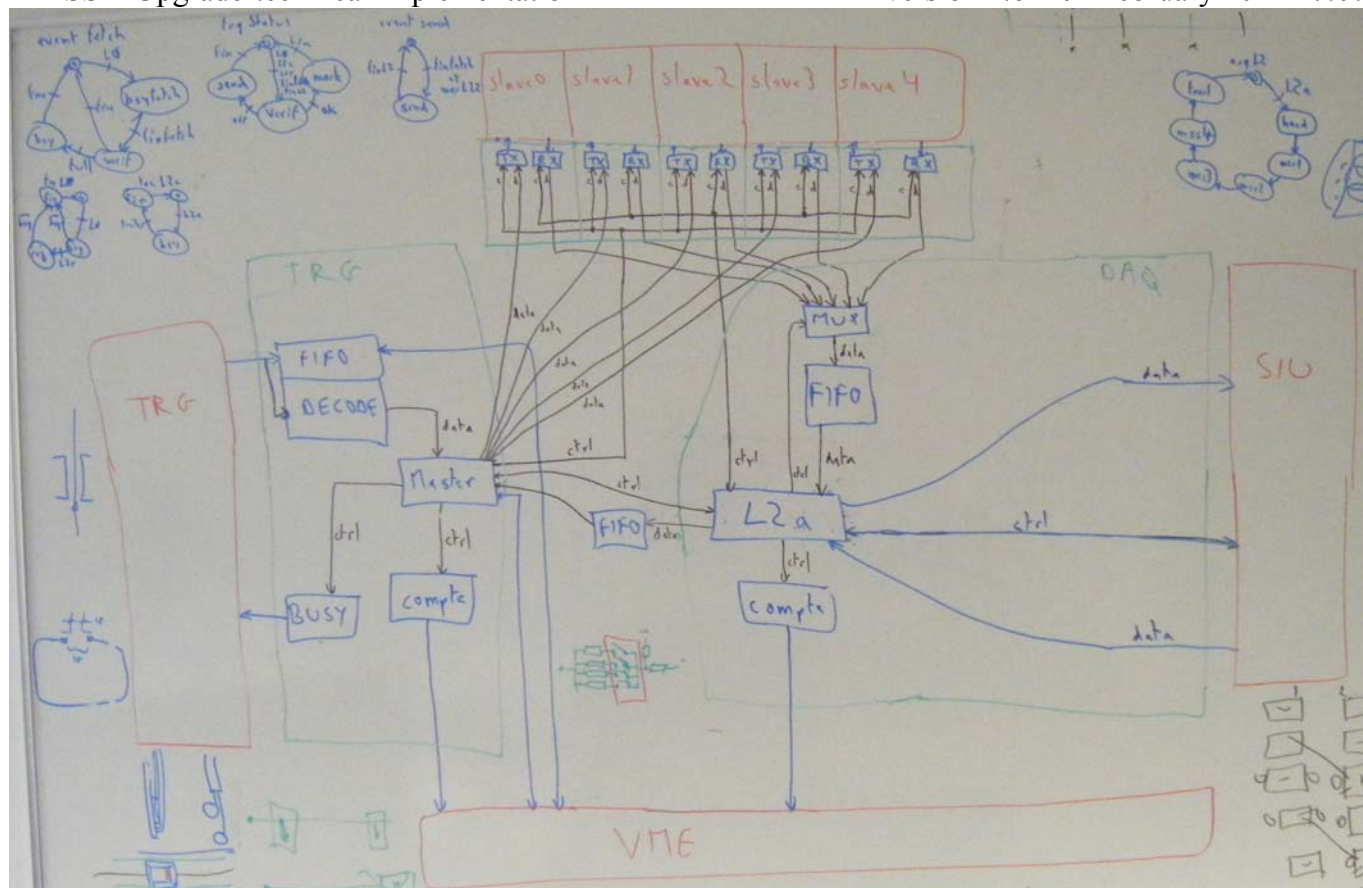


Figure 15: rdoU master FPGA block diagram

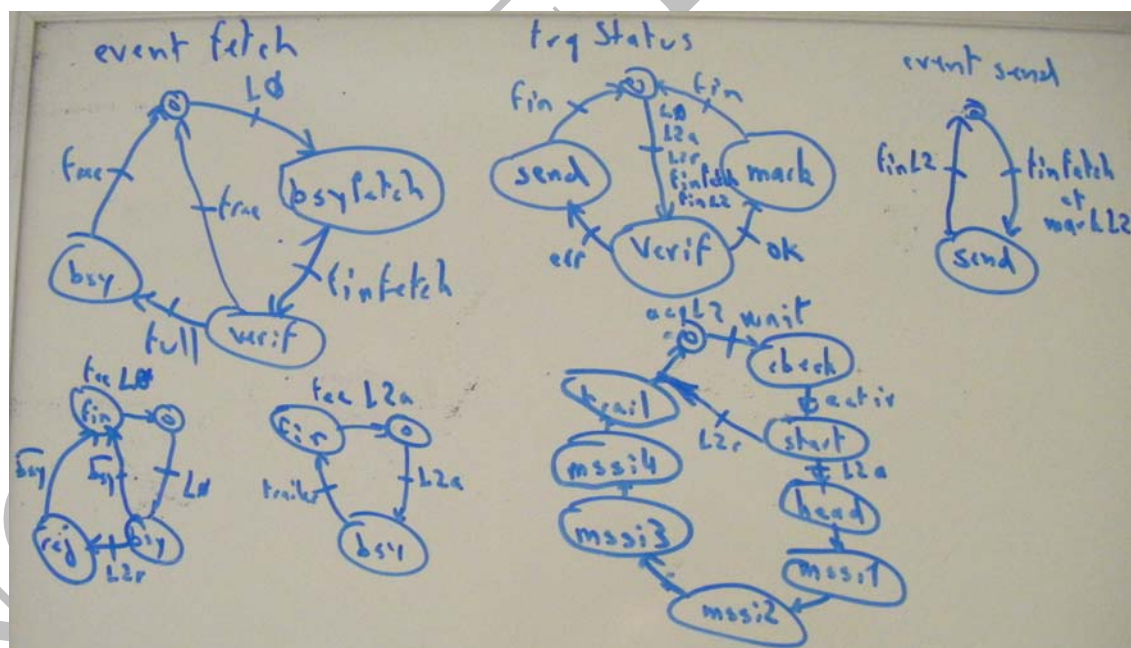


Figure 16: rdoU master FPGA event state machines

L0_fetch has a higher priority at MSSI interface than L2x_send:

- not busy while sending L2 event to DAQ
- L2_send postponed if L0_fetch arrives

4.10.2 rdoU-Master-FPGA TRG part

name		addr	4 bits	4 bits	4 bits	4 bits	4 bits	4 bits	4 bits	4 bits	
ID	r-	0	application	FPGA ID (trg=2)	FPGA version		card #	card serial	daq #	card type (rdoUv0=7)	
date	r-	4	day		month		year				
Status	r-	8	free	free	free	free	free	free	free	free	
config	rw	C	free	free	free	3 free bits ladder_used(4..0)		free trg pos(2..0)	status back select card clk trg vme(1..0) vme action(3..0)		
nbr trg phy	r-	10	used				received				
nbr trg cal	r-	14	used				received				
nbr L2 acc	r-	18	used				received				
nbr L2 rej	r-	1C	used				received				
nbr trg mode 0	r-	20	used				received				
nbr trg mode 1	r-	24	used				received				
nbr trg mode 2	r-	28	used				received				
nbr trg mode 3	r-	2C	used				received				
nbr rhic clock	r-	30	Free running rhic clock counter								
dead time	r-	34	nbr rhic clock while busy								
trg state machines	r-	40	free	free	free	free	free	free	free	free	
delay before hold	rw	44	delay before calib hold				delay before trig hold				
threshold	rw	48	free	free	free	free	threshold				
trg input mem	r-	4C	message number 0		time stamp (rhic clock counter)						
			message number 1		free	non idle trigger message					
L0 event buffer 0	r-	50	time_stamp				L0 message				
L0 event buffer 1	r-	54	time_stamp				L0 message				
L0 event buffer 2	r-	58	time_stamp				L0 message				
L0 event buffer 3	r-	5C	time_stamp				L0 message				
L2 event buffer 0	r-	60	time_stamp				L2 message				
L2 event buffer 1	r-	64	time_stamp				L2 message				
L2 event buffer 2	r-	68	time_stamp				L2 message				
L2 event buffer 3	r-	6C	time_stamp				L2 message				
status event buffer 0	r-	70	mssi_busy				mssi4 status	mssi3 status	mssi2 status	mssi1 status	mssi0 status
status event buffer 1	r-	74	free	free	free	mssi4 status	mssi3 status	mssi2 status	mssi1 status	mssi0 status	
status event buffer 2	r-	78	free	free	free	mssi4 status	mssi3 status	mssi2 status	mssi1 status	mssi0 status	
status event buffer 3	r-	7C	free	free	free	mssi4 status	mssi3 status	mssi2 status	mssi1 status	mssi0 status	
mssi slv2mstr one	r-	80	free	free	free	mssi_slv4_2mstr_one(5..0) mssi_slv3_2mstr_one(5..0) mssi_slv2_2mstr_one(5..0) mssi_slv1_2mstr_one(5..0) mssi_slv0_2mstr_one(5..0)					
mssi_tx_delay	r-	78	free	free	free	mssi4 tx delay	mssi3 tx delay	mssi2 tx delay	mssi1 tx delay	mssi0 tx delay	
mssi_rx_delay	r-	78	free	free	free	mssi4 rx delay	mssi3 rx delay	mssi2 rx delay	mssi1 rx delay	mssi0 rx delay	

Table 37: rdoU master-FPGA TRG part VME registers

Trigger input memory contains 128 words including 28bit timestamp based on RHIC clock tick counter (~29s width) and 20bit “non idle” trigger messages. Two VME read are necessary to get one full word: First read gives timestamp, second read gives trigger message. Most significant digit (4bits) tells if it is a time stamp (0xC) or a trigger message (0x1). **Commands to be implemented: start free running, stop, start running until full, start running until error**

mssi_status (1..0): 00=free-idle; 11=full-idle; 01=busy fetching data from slave; 10=busy sending data to DAQ.

mssi_busy (4..0) (1..0): L2_busy; L0_busy

Master FPGA asserts its "busy_back" signal to the central trigger (see "Table 2: *trigger connector*" and "Figure 13: *BNL recommendation for Trigger interface*") when bit "Slave and ladder OK" (see "Table 41: *RDO-Slave to RDO-Master interface*") is not '1' while the bit corresponding to this slave in "ladder_used" (see config register in "Table 37: *rdoU master-FPGA TRG part VME registers*") is '1'.

4.10.3 rdoU-Master-FPGA ACQ part

name		addr	4 bits	4 bits	4 bits	4 bits	4 bits	4 bits	4 bits	4 bits
ID	r-	0	application	FPGA ID (acq=3)	FPGA version		card #	card serial	daq #	card type (rdoUv0=7)
date	r-	4	day		month		year			
Status	r-	8	free	free	free	free	free	free	free	free
config	rw	C	free	free	free	free	free	free	free	free
nbr trg phy	r-	10	used				received			
nbr trg cal	r-	14	used				received			
nbr L2 acc	r-	18	used				received			
nbr L2 rej	r-	1C	used				received			
nbr trg mode 0	r-	20	used				received			
nbr trg mode 1	r-	24	used				received			
nbr trg mode 2	r-	28	used				received			
nbr trg mode 3	r-	2C	used				received			
nbr rhic clock	r-	30	Free running rhic clock counter							
dead time	r-	34	nbr rhic clock while busy							
free	rw	38	free	free	free	free	free	free	free	free
free	rw	3C	free	free	free	free	free	free	free	free
acq state machines	r-	40	free	free	free	free	free	free	free	free
nbr RDYRX	r-	44	Free running RDYRX counter							
nbr EOBTR	r-	48	Free running EOBTR counter							
nbr L2 acc no daq	r-	4C	nbr L2 acc while DDL closed							
nbr FESTRD	r-	50	Free running FESTRD counter							
nbr link_full	r-	54	Free running link_full counter							
duration link_full	r-	58	nbr rhic clock while link_full							
free	r-	5C	free	free	free	free	free	free	free	free

Table 38: rdoU master-FPGA ACQ part VME registers

4.11 Master<->slave serial Interface (MSSI)

4.11.1 overview

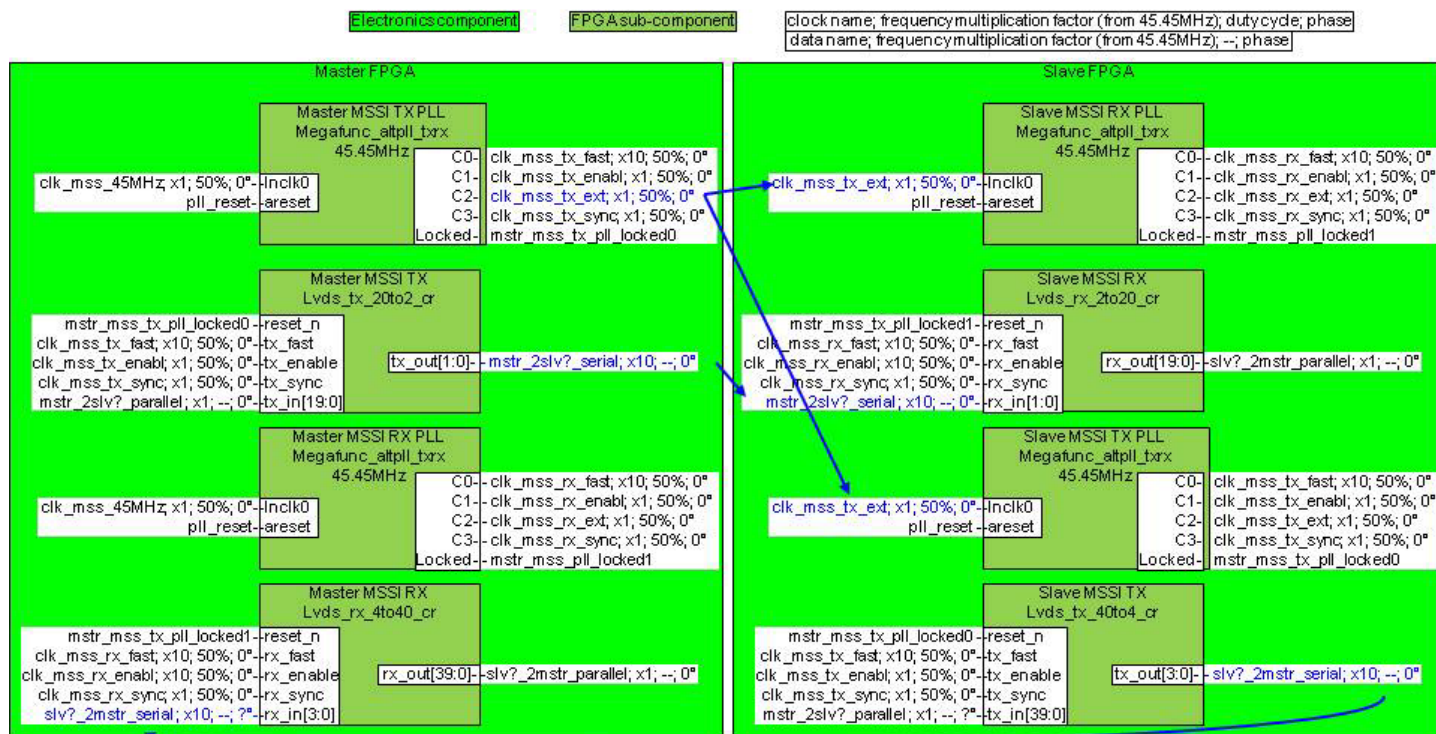


Figure 17: *rdoU* master FPGA MSSI cabling

4.11.2 MSSl rdoU-Master to rdoU-Slave direction

4 Steering bits	Associated action	16 payload bits		
0000	idle	16 free bits		
0001	Slave configure	12 free bits	2 mode bits	2 Write buffer # bits
0010	Pedestal write start address	2 free bits	14 Pedestal write address bits	
0011	Pedestal write data	5 free bits	1 auto increment addr bit	10 Pedestal write data bits
0100	Test pulse	16 free bits		
0101	Send event data	14 free bits		
0110	Set hold after delay	Delay before hold (36.4MHz clock ticks)		
0111	free (idle)	16 free bits		
1000	Abort event fetching	16 free bits		
1001	threshold	threshold		
1010	Slave decoder reset	16 free bits		
1011	free (idle)	16 free bits		
1100	free (idle)	16 free bits		
1101	free (idle)	16 free bits		
1110	free (idle)	16 free bits		
1111	free (idle)	16 free bits		

Table 39: *RDO-Master to RDO-Slave interface steering bits and associated action and payload*

2 slave mode bits	Associated mode
00	ADC data pedestal subtracted and zero suppressed (physics)
01	ADC data pedestal subtracted
10	Raw ADC data
11	Pedestal memory

Table 40: RDO-Master to RDO-Slave interface RDO-Slave mode

4.11.3 MSSl rdoU-Slave to rdoU-Master direction

4 free bits	1 bit	1 bit	2 steering bits	32 payload bits
free	Slave and ladder OK	Busy fetching data from ladder	steering bits	payload

Table 41: RDO-Slave to RDO-Master interface

Master FPGA asserts its "busy_back" signal to the central trigger (see "Table 2: trigger connector" and "Figure 13: BNL recommendation for Trigger interface") when bit "Slave and ladder OK" (see "Table 41: RDO-Slave to RDO-Master interface") is not '1' while the bit corresponding to this slave in "ladder_used" (see config register in "Table 37: rdoU master-FPGA TRG part VME registers") is '1'.

2 steering bits	Associated payload
00	idle
01	Event data
10	header
11	trailer

Table 42: RDO-Slave to RDO-Master interface steering bits and associated payload

4.11.3.1 MSSl rdoU-Slave to rdoU-Master direction payload

pos	2 steering bits	Name/value	Number of words		
			min	typ	max
0-16	10	Slave header	17	17	17
17-a	01	Hybrid 0 block	1	24	769
a-b	01	Hybrid 1 block	1	24	769
b-c	01	Hybrid 2 block	1	24	769
c-d	01	Hybrid 3 block	1	24	769
d-e	01	Hybrid 4 block	1	24	769
e-f	01	Hybrid 5 block	1	24	769
f-g	01	Hybrid 6 block	1	24	769
g-h	01	Hybrid 7 block	1	24	769
h-i	01	Hybrid 8 block	1	24	769
i-j	01	Hybrid 9 block	1	24	769
j-k	01	Hybrid 10 block	1	24	769
k-l	01	Hybrid 11 block	1	24	769
l-m	01	Hybrid 12 block	1	24	769
m-n	01	Hybrid 13 block	1	24	769
n-o	01	Hybrid 14 block	1	24	769
o-p	01	Hybrid 15 block	1	24	769

Table 43: RDO-Slave to RDO-Master interface payload

4.11.3.2 MSSI rdoU-Slave to rdoU-Master direction payload “Slave header”

pos	2 steering bits	Name/value	Number of words
0	10	Slave total words	1
1	10	Spacer (0xFACEBEEF)	1
2-9	10	Ladder status	8
10-15	10	Hybrid total words	6
16	10	Spacer (0xBEEFFACE)	1

Table 44: RDO-Slave to RDO-Master interface payload “RDO-slave header”

2 steering bits	7 free bits	3 slave # bits	3 ladder # bits	1 bit	2 slave mode bits	16 slave total words bits	Number of words
10	slave free	slave #	ladder #	ladder ok	slave mode	slave total words (including this word)	1

Table 45: RDO-Slave to RDO-Master interface payload “RDO-slave total words”

Ladder_ok tells that the ladder-FPGA is programmed and the GBIC serdes is locked.

pos	2 steering bits	11 free bits	3 Status # bits	2 latchup bits	1 busy bit	1 bit	1 bit	1 bit	12 status bits	Number of words
0	10	slave free	000	latchup 0-1	ladder_busy	configured	ok	serdes clock used	temperature 0	1
1	10	slave free	001	latchup 2-3	ladder_busy	des lock	des bist pass	crc error	Temperature 1	1
2	10	slave free	010	latchup 4-5	ladder_busy	fib tx fault	fib mod absent	fib rx loss	Temperature 2	1
3	10	slave free	011	latchup 6-7	ladder_busy	usb present	usb rdy_n	debug present	Temperature 3	1
4	10	slave free	100	latchup 8-9	ladder_busy	test	hold	hv side	nbr_hold	1
5	10	slave free	101	latchup 10-11	ladder_busy	serial(5)	serial(4)	serial(3)	nbr_test	1
6	10	slave free	110	latchup 12-13	ladder_busy	serial(2)	serial(1)	serial(0)	nbr_token	1
7	10	slave free	111	latchup 14-15	ladder_busy	ladder #			nbr_abort	1

Table 46: rdoU-Slave to rdoU-Master interface payload “ladder status”

Serial(5 down to 0) is the serial number of the ladder card. It is defined on the ladder card printed circuit board.

Nbr_hold is the number of times that the HOLD signal was asserted since last reset of the ladder card.

Nbr_test is the number of times that the TEST signal was asserted since last reset of the ladder card.

Nbr_token is the number of times that the TOKEN signal was asserted since last reset of the ladder card.

Nbr_abort is the number of times that the HOLD signal was de-asserted during event read since last reset of the ladder card.

“serdes clock used” means that the ladder FPGA switchover is locked on the incoming serdes clock and not on the local crystal clock.

“hv side” bit indicates, when asserted, that TEST signal and each bit of ADC data are complemented.

pos	2 steering bits	2 free bits	10 hybrid total words bits	10 hybrid total words bits	10 hybrid total words bits	Number of words
0	10	slave free	Hybrid 2 block total words	Hybrid 1 block total words	Hybrid 0 block total words	1
1	10	slave free	Hybrid 5 block total words	Hybrid 4 block total words	Hybrid 3 block total words	1
2	10	slave free	Hybrid 8 block total words	Hybrid 7 block total words	Hybrid 6 block total words	1
3	10	slave free	Hybrid 11 block total words	Hybrid 10 block total words	Hybrid 9 block total words	1
4	10	slave free	Hybrid 14 block total words	Hybrid 13 block total words	Hybrid 12 block total words	1
5	10	slave free	slave free	slave free	Hybrid 15 block total words	1

Table 47: rdoU-Slave to rdoU-Master interface payload “hybrid total words”

4.11.3.3 MSSI rdoU-Slave to rdoU-Master direction payload “Hybrid block”

4.11.3.3.1 MSSI rdoU-Slave to rdoU-Master direction payload “Hybrid block (zero suppressed, pedestal subtracted readout mode)”

pos	2 steering bits	5 free bits	3 slave bits	4 module bits	10 strip addr bits	10 data bits	Number of words		
							min	typ	max
0-n	01	slave free	Slave #	Module #	Strip #	Pedestal subtracted, zero suppressed	0	24	768
n+1	01	0xCAFEFADE					1	1	1

Table 48: rdoU-Slave to rdoU-Master interface payload “hybrid block zero suppressed (mode=0)”

4.11.3.3.2 MSSI rdoU-Slave to rdoU-Master direction payload “Hybrid block (pedestal subtracted data readout mode)”

pos	2 steering bits	2 free bits	10 data bits	10 data bits	10 data bits	Number of words
0-255	01	slave free	Pedestal subtracted	Pedestal subtracted	Pedestal subtracted	256
256	01	0xCAFEFADE				1

Table 49: rdoU-Slave to rdoU-Master interface payload “hybrid block pedestal subtracted (mode=1)”

4.11.3.3.3 MSSI rdoU-Slave to rdoU-Master direction payload “Hybrid block (ADC values readout mode)”

pos	2 steering bits	2 free bits	10 data bits	10 data bits	10 data bits	Number of words
0-255	01	slave free	ADC data	ADC data	ADC data	256
256	01	0xCAFEFADE				1

Table 50: rdoU-Slave to rdoU-Master interface payload “hybrid block raw ADC data (mode=2)”

4.11.3.3.4 MSSl rdoU-Slave to rdoU-Master direction payload “Hybrid block (pedestal memory readout mode)”

pos	2 steering bits	2 free bits	10 data bits	10 data bits	10 data bits	Number of words
0-255	01	slave free	Pedestal memory	Pedestal memory	Pedestal memory	256
256	01	0xCAFEFADE				1

Table 51: rdoU-Slave to rdoU-Master interface payload “hybrid block pedestal memory (mode=3)”

4.12 rdoU-Slave-FPGA

4.12.1 Overview

name		addr	4 bits	4 bits	4 bits	4 bits	4 bits	4 bits	4 bits	4 bits	
ID	r-	0	•application	•FPGA ID (slaveX=8+X)	•FPGA version		•card #	•card serial	•free	•card type (rdoUv1=7)	
version date	r-	4	•day			•month		•year			
Status	r-	8	•free	•free	•free	•free	•free	•free	•free	•free	
config	rw	C	•vme address master •reset counters •free •free	•free	•free	•free	•free	•free	•free	•free	
nbr trg phy	r-	10	•used				•received				
nbr trg cal	r-	14	•used				•received				
nbr L2 acc	r-	18	•used				•received				
nbr L2 rej	r-	1C	•used				•received				
nbr trg mode 0	r-	20	•used				•received				
nbr trg mode 1	r-	24	•used				•received				
nbr trg mode 2	r-	28	•used				•received				
nbr trg mode 3	r-	2C	•used				•received				
nbr rhic clock	r-	30	•Free running rhic clock counter								
dead time	r-	34	• nbr rhic clock while busy								
version time	r-	38	hour			minute		second		•free	•free
jtag ladderSC cmd	rw	40	•free	•free	•free	•free	•free	•free	•free	•tdi •tms •tdo •tdo valid	
jtag ladderSC clk	rw	44	•free •free •tck delay(9..8)	•tck delay(7..0)		•delay for tdo(9..2)		•delay for tdo(1..0) •clk high duration(9..8)	•clk high duration(7..0)		
FPGA config cmd	-w	48	•cmd(2..0) •erase	•start address					•number of bytes -1		
FPGA config status	r-		•free	•free	•free	•free	•free	•free	•free	•free •ladder config •cmd busy •epcs4 busy	
FPGA config data	-w	4C	•first byte			•second byte		•third byte		•fourth byte	
rd buff no	rw	80	•free	•free	•free	•free	•free	•free	•free	•free	
wr buff no	rw	84	•free	•free	•free	•free	•free	•free	•free	•free	
mem start addr	rw	88	•free	•free	•free	•free	•free	•free	•free	•free	
adc mask	rw	8C	•free	•free	•free	•free	•free	•free	•free	•free	
Pedestal	rw	90	•free	•free	•free	•free	•free	•free	•free	•free	
hdr buffer	r-	94	•free	•free	•free	•free	•free	•free	•free	•free	
event buffer	rw	98	•free	•free	•free	•free	•free	•free	•free	•free	

Table 52: rdoU slave-FPGA VME registers

When vme_address_master='1' (config register), VME processor has access to the internal memory of the slave-FPGA. vme_address_master should be '0' for normal operation

3 cmd bits	Associated command
000	load configuration data into EPCS4
001	transfer configuration data from EPCS4 to ladder FPGA
other	reserved

Table 53: rdoU slave-FPGA VME FPGA_config_cmd register cmd bits and associated commands

4.12.2 EPSC4

The ladder FPGA configuration data are stored in a small serial device (EPCS4) associated with each slave FPGA on the RDO.

The VME processor can use the VME interface of the slave FPGA to load new data into the serial PROM. Writing the 4 Mbits of configuration data to the EPCS4 is a slow process (approximately 20 sec for bulk erase followed by multiple writes). But this process needs to be repeated only when the configuration data change (new code version).

The VME processor initiates the configuration by sending a “transfer configuration data to ladder FPGA” command to the slave FPGA. Transferring the data stored in the EPCS4 to the ladder FPGA requires approximately 0.5 sec.

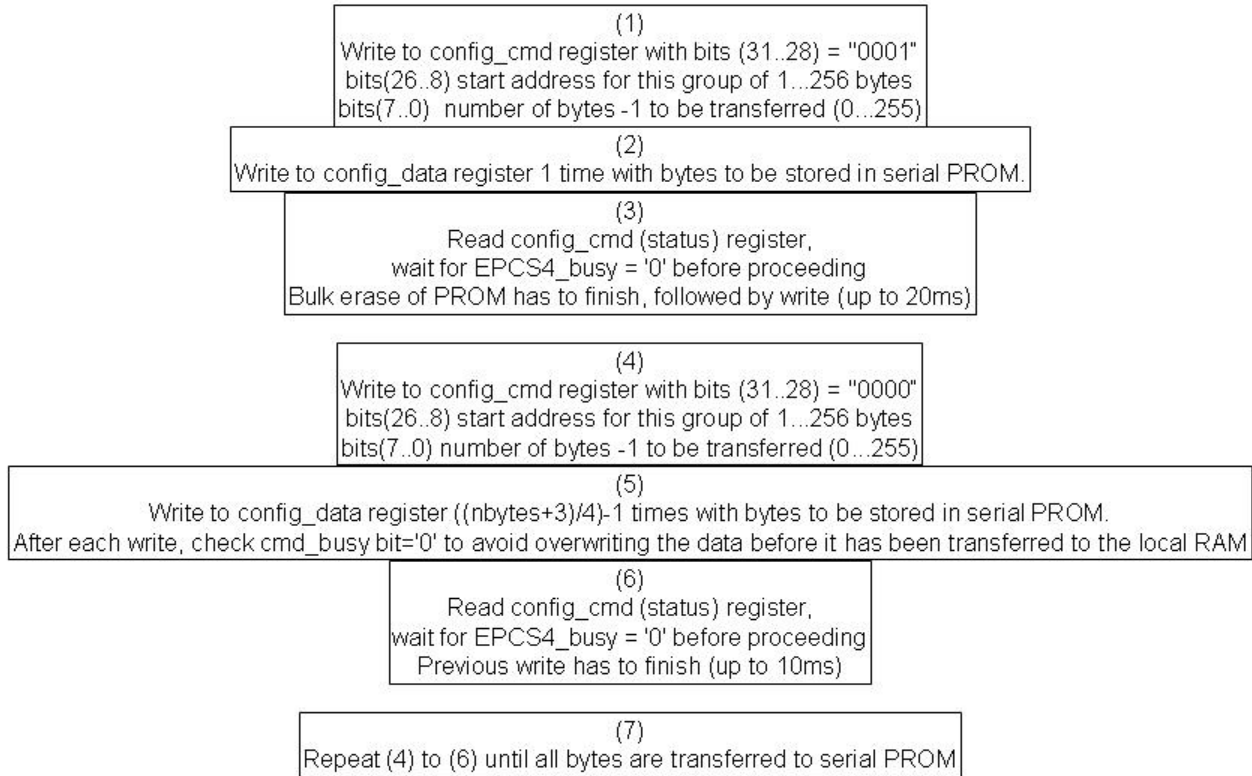


Figure 18: RDO slave-FPGA VME protocol to load data into EPCS4

Step (1) is time consuming (5 or more seconds), but the slave RDO will not attempt to transfer the first batch of data from its internal RAM to the EPCS4 until the bulk erase is complete. Effectively, the polling in step (3) for the first iteration will have a duration which includes the bulk erase.

The command containing the start address with initial address = 1 (step 1) differs from the command with initial address = 0 (step 4) only in that a bulk erase of the EPCS4 memory is performed when initial address=1. Each of the iterations of steps (4) to (7) will require approx. 5 msec. As many as 2048 iterations are required, depending on the program represented by the configuration data contained in the programming file (.rpd or .hex).

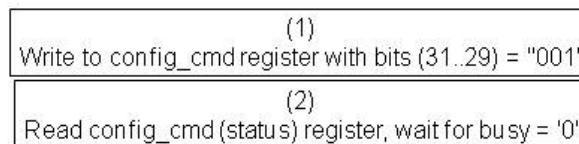


Figure 19: RDO slave-FPGA VME protocol to configure ladder FPGA with data stored in EPCS4

Estimated time required: 0.5 sec.

4.12.3 JTAG

Each slave FPGA emulates an ACT8990^[13] (TI) that was in Corelis CVME-1149.1 card that was used in the old SSD.

4.13.1 Overview

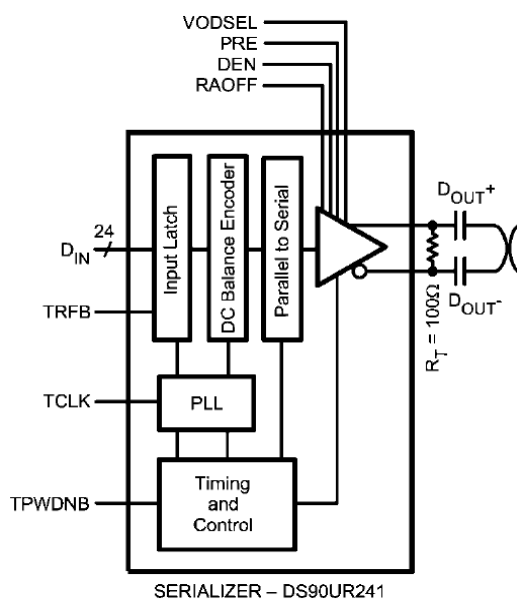
DS90UR241^[14] 5-43 MHz DC-Balanced 24-Bit LVDS serializer

Figure 20: DS90UR241^[14] 5-43 MHz DC-Balanced 24-Bit LVDS serializer block diagram

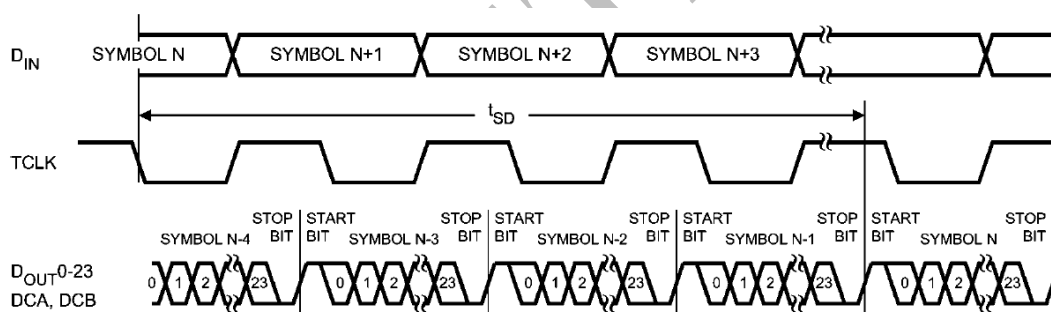


Figure 21: DS90UR241^[14] 5-43 MHz DC-Balanced 24-Bit LVDS serializer timing

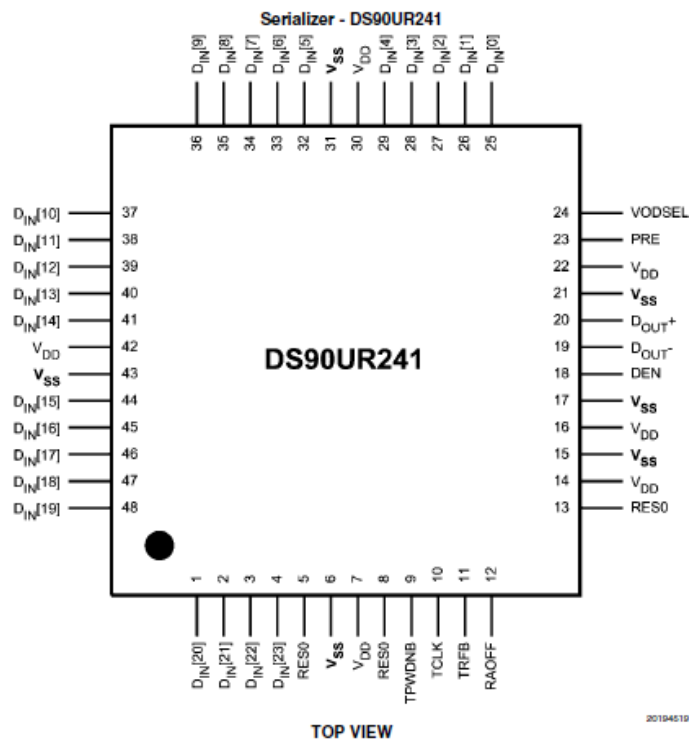


Figure 22: DS90UR241^[14] 5-43 MHz DC-Balanced 24-Bit LVDS serializer pinout (top view)

AFBR-57M5APZ^[15] 850 nm, SFP (Small Form Pluggable), RoHS Compliant, Low Voltage (3.3 V) Digital Diagnostic Optical Transceiver

DS90UR124^[14] 5-43 MHz DC-Balanced 24-Bit LVDS deserializer

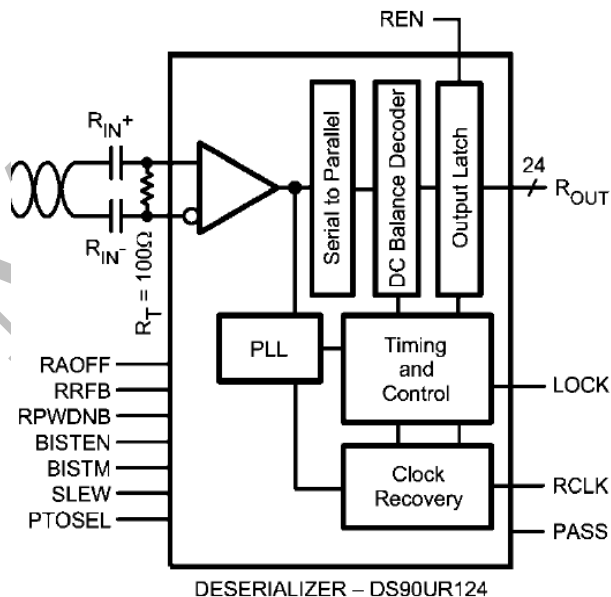


Figure 23: DS90UR124^[14] 5-43 MHz DC-Balanced 24-Bit LVDS deserializer block diagram

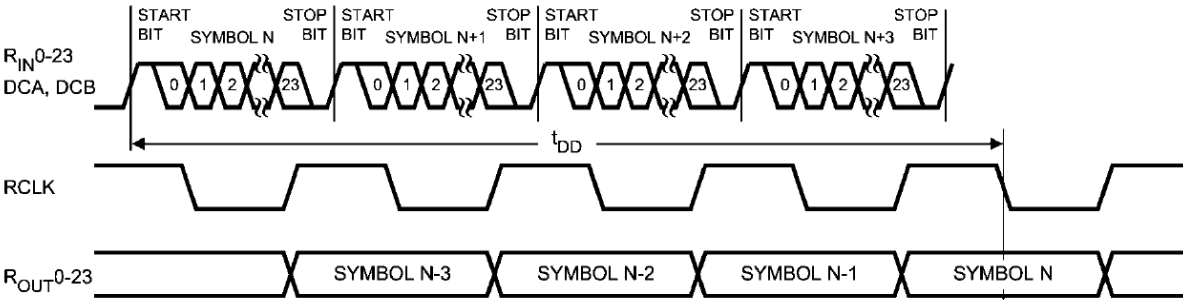


Figure 24: DS90UR124^[14] 5-43 MHz DC-Balanced 24-Bit LVDS deserializer timing

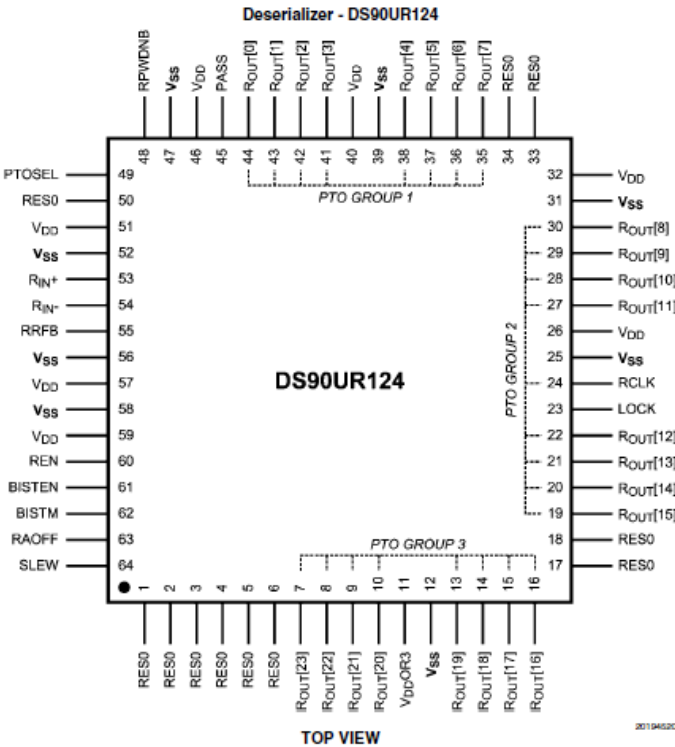


Figure 25: DS90UR124^[14] 5-43 MHz DC-Balanced 24-Bit LVDS deserializer pinout (top view)

4.13.2 GBIC: RDO-slave to Ladder direction

4 fpga bits				10 free bits	3 addr bits	3 trigger bits			4 slow-control bits			
fpga dclk	fpga nconfig	fpga data0	Init switch	slave free	ladder address	token	test	hold	ladder sc tck	ladder sc tms	ladder sc trst	ladder sc tdi

Table 54: GBIC: RDO-slave to ladder interface

4.13.3 GBIC: Ladder to RDO-slave direction

1 bit	1 bit dependant on Init switch	22 payload bits
FPGA NSTATUS	return from FPGA	payload

Table 55: GBIC: ladder to RDO-slave interface

When “Init switch” is 0, “return from FPGA” is FPGA_CONF_DONE from ladder_FPGA. When “Init switch” is 1, “return from FPGA” is JTAG_SC_TDO from ladder_FPGA.

When the ladder-FPGA is not configured, all bits of the payload are 0 due to line termination resistors.

<i>pos</i>	<i>data ready</i>	<i>first word</i>	<i>20 packed data bits</i>					<i>Number of words</i>
0	1	1	bit8(3..0)	bit9(15..12)	bit9(11..8)	bit9(7..4)	bit9(3..0)	1
1	1	0	bit7(7..4)	bit7(3..0)	bit8(15..12)	bit8(11..8)	bit8(7..4)	1
2	1	0	bit6(11..8)	bit6(7..4)	bit6(3..0)	bit7(15..12)	bit7(11..8)	1
3	1	0	bit5(15..12)	bit5(11..8)	bit5(7..4)	bit5(3..0)	bit6(15..12)	1
4	1	1	bit3(3..0)	bit4(15..12)	bit4(11..8)	bit4(7..4)	bit4(3..0)	1
5	1	0	bit2(7..4)	bit2(3..0)	bit3(15..12)	bit3(11..8)	bit3(7..4)	1
6	1	0	bit1(11..8)	bit1(7..4)	bit1(3..0)	bit2(15..12)	bit2(11..8)	1
7	1	0	bit0(15..12)	bit0(11..8)	bit0(7..4)	bit0(3..0)	bit1(15..12)	1

Table 56: GBIC: ladder to RDO-slave interface payload “acquisition”

<i>pos</i>	<i>data ready</i>	<i>3 position bits</i>	<i>2 power_up bits</i>	<i>1 busy bit</i>	<i>1 bit</i>	<i>1 bit</i>	<i>1 bit</i>	<i>12 status bits</i>	<i>Number of words</i>
0	0	000	hybrid_pow(1..0)	ladder_busy	configured	ok	serdes clock used	temp_sensor0	1
1	0	001	hybrid_pow(3..2)	ladder_busy	des lock	des bist pass	crc error	temp_sensor1	1
2	0	010	hybrid_pow(5..4)	ladder_busy	fib tx fault	fib mod absent	fib rx loss	temp_sensor2	1
3	0	011	hybrid_pow(7..6)	ladder_busy	usb present	usb rdy_n	debug present	temp_sensor3	1
4	0	100	hybrid_pow(9..8)	ladder_busy	test	hold	hv side	nbr_hold	1
5	0	101	hybrid_pow(11..10)	ladder_busy	serial(5)	serial(4)	serial(3)	nbr_test	1
6	0	110	hybrid_pow(13..12)	ladder_busy	serial(2)	serial(1)	serial(0)	nbr_token	1
7	0	111	hybrid_pow(15..14)	ladder_busy	Ladder #			nbr_abort	1

Table 57: GBIC: ladder to RDO-slave interface payload “status”

Serial(5 downto 0) is the serial number of the ladder card. It is defined on the printed circuit board.

Nbr_hold is the number of times that the HOLD signal was asserted since last reset of the ladder card.

Nbr_test is the number of times that the TEST signal was asserted since last reset of the ladder card.

Nbr_token is the number of times that the TOKEN signal was asserted since last reset of the ladder card.

Nbr_abort is the number of times that the HOLD signal was de-asserted during event read since last reset of the ladder card.

“serdes clock used” means that the ladder FPGA switchover is locked on the incoming serdes clock and not on the local crystal clock.

Ladder_busy signal will be asserted in case of an abort: dataready=0 (no more data to send) but event not finished (busy getting the token out of the Alice128^{[4][5]} chips). Slave-FPGA should wait until both ladder_busy and dataready signals are off before de-asserting the “Busy fetching data from ladder” signal to the Master-FPGA.

Debug_present signal indicates that the debug daughter card is plugged in.

Usb_present signal indicates that USB daughter card is plugged in.

“hv side” bit indicates, when asserted, that TEST signal and each bit of ADC data are complemented.

5 The ladder card

5.1 Overview

The ladder card is the upgrade for the CONNEXION card (sometimes called C2D2 card) and for the ADC card.

5.2 Power management

typical	Qty	5V	analog	3.3V	3.3V	2.5V	2.5V	1.2V	
74LV125APW	2				0.20				quadruple bus buffer
74LV126APW	2				0.20				quadruple bus buffer
90LV031	12			23.00					quad cmos differential line drivers
AD7356	8					14.00			dual 12bits serial output ADC
AD8040	16		5.20						quad rail-to-rail op-amp
AFBR57M5APZ	1				210.00				optical fiber transceiver
Altera blaster	1						0.00		DEBUG altera cable
CMP04	4			0.80					quad comparator
CY7B991	1			28.40					roboclock
CY7B991	1				30.56				roboclock
DS90UR124	1				85.00				1 to 24 deserializer
DS90UR241	1				66.00				24 to 1 serializer
EP3C16FC484-6	1					11.34	21.98	42.99	FPGA
FAN2558S12X	1			43.14					low-dropout (1.7V) 1.2V regulator
LM1117MPX-2.5	2	77.66							low-dropout (1.2V) 2.5V regulator
LM1117MPX-3.3	2	376.85							low-dropout (1.2V) 3.3V regulator
LTC1662	1		0.01						dual 10bits serial input DAQ
MAX6575L	2			0.15					temperature sensor
MAX6575L	2				0.15				temperature sensor
IQD CFPS-32IB 40MHz	1					0.00			DEBUG local oscillator
UM245R	1				0.00				DEBUG usb interface
TOTAL		909.0	83.2	351.0	392.7	123.3	22.0	43.0	mA
	4955	1670	416	1158	1296	308	55	52	mW

Table 58: ladder card typical power requirement (normal operation)^{[9][14][15][16][17][18][19][20][21][22]}

Typical current values from the referenced data sheets were used in this table.

CycloneIII FPGA require two separate 2.5V power planes (1 digital and 1 analog)

Analog voltage assumed to be 5V (as worst case) for power calculation: In old SSD, one had to adjust -2V around -2.2V and to adjust +2V around +2.5V to get -2V and +2V on all the hybrids. This was due to voltage loss along the flex cables on the ladder and to the sense cable being connected to the electronics at the end of the ladder.

<i>maximum</i>	Qty	5V	analog	3.3V	3.3V	2.5V	2.5V	1.2V	
74LV125APW	2				0.20				quadruple bus buffer
74LV126APW	2				0.20				quadruple bus buffer
90LV031	12			30.00					quad cmos differential line drivers
AD7356	8					20.00			dual 12bits serial output ADC
AD8040	16		6.00						quad rail-to-rail op-amp
AFBR57M5APZ	1				210.00				optical fiber transceiver
Altera blaster	1						0.10		DEBUG altera cable
CMP04	4			2.00					quad comparator
CY7B991	1			146.00					roboclock
CY7B991	1				146.00				roboclock
DS90UR124	1				105.00				1 to 24 deserializer
DS90UR241	1				90.00				24 to 1 serializer
EP3C16FC484-6	1					11.34	21.98	42.99	FPGA
FAN2558S12X	1			43.14					low-dropout (1.7V) 1.2V regulator
LM1117MPX-2.5	2	110.71							low-dropout (1.2V) 2.5V regulator
LM1117MPX-3.3	2	567.77							low-dropout (1.2V) 3.3V regulator
LTC1662	1		0.80						dual 10bits serial input DAQ
MAX6575L	2			0.40					temperature sensor
MAX6575L	2				0.40				temperature sensor
IQD CFPS-32IB 40MHz	1					18.00			DEBUG local oscillator
UM245R	1				15.00				DEBUG usb interface
TOTAL		1357.0	96.8	557.9	567.6	189.3	22.1	43.0	mA
	7262	2484	484	1841	1873	473	55	52	mW

Table 59: ladder card maximum power requirement (normal operation) ^{[9][14][15][16][17][18][19][20][21][22]}

Maximum current values from the referenced data sheets were used in this table.

Analog voltage assumed to be 5V (as worst case) for power calculation: In old SSD, one had to adjust -2V around -2.2V and to adjust +2V around +2.5V to get -2V and +2V on all the hybrids. This was due to voltage loss along the flex cables on the ladder and to the sense cable being connected to the electronics at the end of the ladder.

<i>typical</i>	Qty	5V	analog	3.3V	3.3V	2.5V	2.5V	1.2V	
74LV125APW	2				0.20				quadruple bus buffer
74LV126APW	2				0.20				quadruple bus buffer
90LV031	12			8.00					quad cmos differential line drivers
AD7356	8					14.00			dual 12bits serial output ADC
AD8040	16		1.30						quad rail-to-rail op-amp
AFBR57M5APZ	1				0.00				optical fiber transceiver
Altera blaster	1						0.02		DEBUG altera USB-blaster cable
CMP04	4			0.80					quad comparator
CY7B991	1			28.40					roboclock
CY7B991	1				30.56				roboclock
DS90UR124	1				30.00				1 to 24 deserializer
DS90UR241	1				40.00				24 to 1 serializer
EP3C16FC484-6	1					11.34	21.98	42.99	FPGA
FAN2558S12X	1			43.14					low-dropout (1.7V) 1.2V regulator
LM1117MPX-2.5	2	86.67							low-dropout (1.2V) 2.5V regulator
LM1117MPX-3.3	2	141.35							low-dropout (1.2V) 3.3V regulator
LTC1662	1		0.01						dual 10bits serial input DAQ
MAX6575L	2			0.15					temperature sensor
MAX6575L	2				0.15				temperature sensor
IQD CFPS-32IB 40MHz	1					18.00			DEBUG local oscillator
UM245R	1				0.00				DEBUG usb interface
TOTAL		456.0	20.8	171.0	101.7	141.3	22.0	43.0	mA
	2378	914	104	564	335	353	55	52	mW

Table 60: ladder card typical power requirement (debug operation without hybrid, without RDO) ^{[9][14][15][16][17][18][19][20][21][22][23][24]}

numbers in red are only guesses to match measured current on +5V power supply.

Typical current values from the referenced data sheets were used in this table.

Analog voltage assumed to be 5V (as worst case) for power calculation: In old SSD, one had to adjust -2V around -2.2V and to adjust +2V around +2.5V to get -2V and +2V on all the hybrids. This was due to voltage

loss along the flex cables on the ladder and to the sense cable being connected to the electronics at the end of the ladder.

5.3 Power connector



Figure 26: Nicomatic power connector cabling

Colors in Figure do not use general typographic convention defined in §2.1 of this document: Green is Analog Ground, Blue is Digital Ground (=−2V)

To protect the thin sense cables in case of a short circuit, all sense lines (+, − and ref) have a 0603L010 polyfuse^[25] in series.

Cables have copper-clad aluminum wires^[26]. AWG26 for sense and bias and AWG22 (or bigger) for low voltage power.

According to Table 58, Table 59 and Table 94, currents for one ladder end (each Nicomatic connector) are:

	<i>−2V power supply</i>	<i>+2V power supply</i>	<i>+5V power supply</i>
typical	870 mA	2172 mA	909 mA
maximum	883 mA	2186 mA	1357 mA

Table 61: ladder-end current requirement for low voltage power supplies

Bias current for one ladder (one Nicomatic connector per ladder) is:

	<i>bias power supply</i>
typical	16*5 μ A
maximum	16*10 μ A

Table 62: ladder current requirement for bias power supply

5.4 Grounding

The ground wire of the High voltage is available on the fixing hole of the lower part of the ladder card. It will be connected to the carbon-epoxy ladder only on the East side.

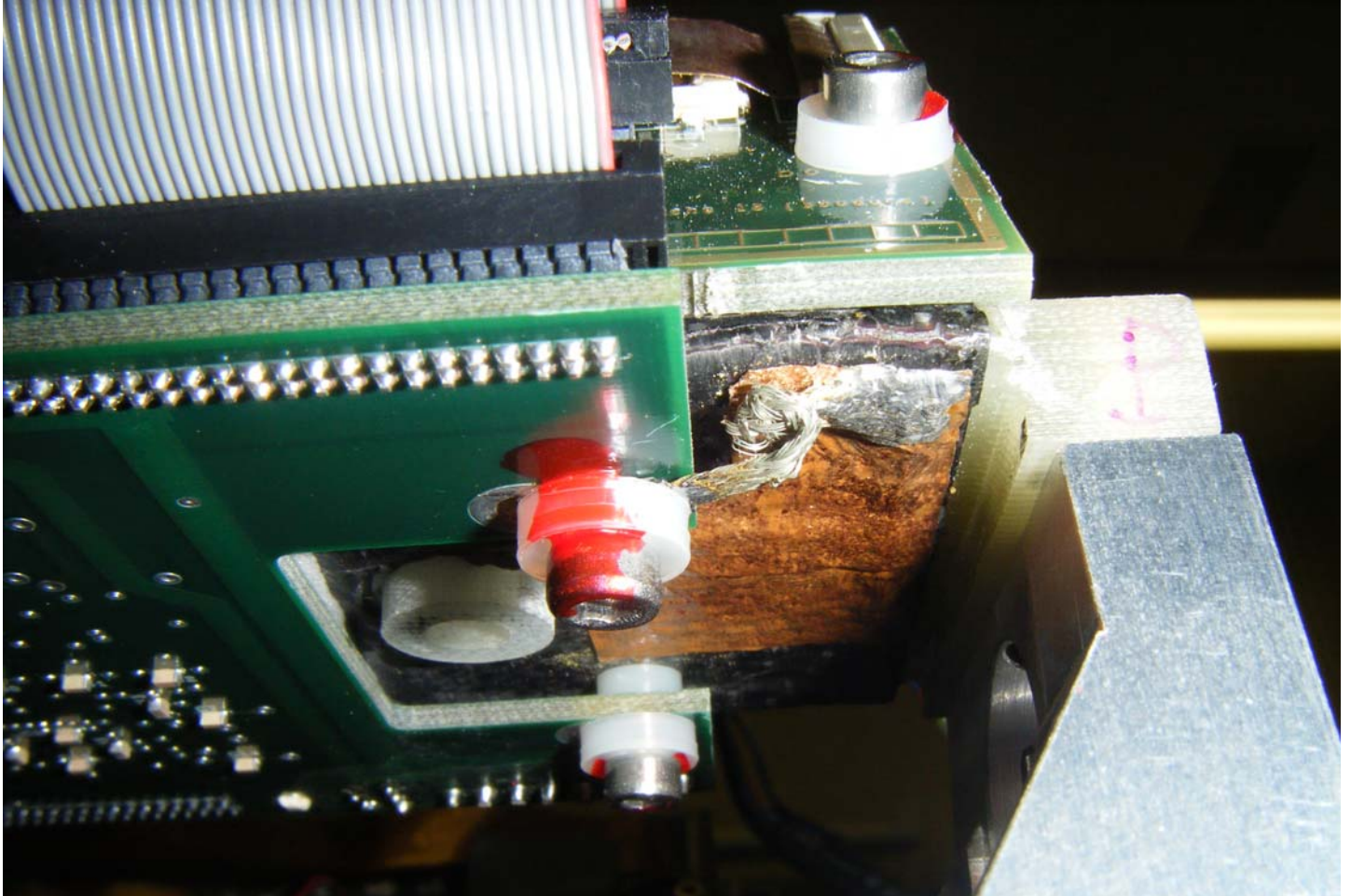


Figure 27: ground connection for the carbon-epoxy ladder (old ADC card)

5.5 Ladder card switches

There is a block of 4 micro switches on the ladder card. They mainly permit to enable/disable debug components or interfaces

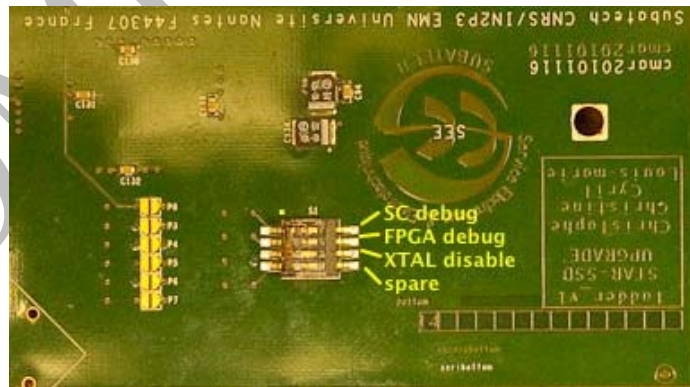


Figure 28: Ladder card switches

sc_debug selects between normal slow-control via optical fiber or slow-control from debug card.

FPGA_debug selects between normal FPGA configuring via optical fiber or FPGA configuring from debug card.

XTAL_disable enables/disables onboard xtal oscillator.

spare selects between normal acquisition via optical fiber or acquisition via USB on debug card.

5.6 Power-up reset (SMT1813 Open Drain, Active-Low with Push- Button Detect)

An Open Drain, Active-Low with Push- Button Detect STMicroelectronics is used to generate a reset signal when digital power supply (+5V) falls below 4.62V (SMT1813L).

Christophe-M.-A. Renard, Subatech, IN2P3/CNRS-L'UNAM, Nantes, F44307, France

Micheal-J. LeVine, BNL, Upton, NY, USA

Stéphane Bouvier, Subatech, IN2P3/CNRS-L'UNAM, Nantes, F44307, France

This component gives the possibility to add a second source for the reset but this is not yet implemented on the ladder card.

5.7 Clock management

There is a 40MHz CFPS32IB^[23] quartz oscillator from IQD/CMAC-Microtechnology onboard. The enable pin of this component is connected to pin 5 of Switch S1 (channel S1c). In normal operation, to lower consumption and remove a potential parasitic signal, this onboard oscillator is disabled. This oscillator must be enabled for tests when the optical fiber is not connected.

A switchover implemented in the ladder-FPGA automatically selects between the "slave to ladder de-serialized clock" (36.4MHz) and the local oscillator (40MHz). Two outputs of this switchover give the "ladder main clock" (36.4MHz) and the "ADC serial clock" ($2 \times 36.4 = 72.8$ MHz).

Here is my old (07 April 2009) opinion on the usage of the different clocks:

- "slave to ladder de-serialized clock" (36.4MHz)
becomes "ladder main clock" (36.4MHz) in the ladder-FPGA
becomes "ADC serial clock" (72.8MHz) in the ladder-FPGA
- "ladder main clock" (36.4MHz)
becomes "ladder to slave serialized clock" (36.4MHz) in the ladder-FPGA
- "ladder to slave de-serialized clock" (36.4MHz)

5.8 Ladder readout sequence

The strips in each silicon wafer are read out by 6 Alice128^{[4][5]} chips. The connections of the strips to the Alice128 chip are not in sequential order, due to routing constraints. Thus the order in which the ADC values appear does not correspond to the geographical order of the strips. The correspondence is defined by the following two lists. The first list consists of the readout order, where the entry position corresponds to the physical silicon strip number. The second list consists of silicon strip numbers, ordered by readout order. Note that the lists contain 128 entries; the pattern is repeated for each of the six Alice128 chips connected to a wafer.

<pre>//table of ALICE128 readout order ordered by silicon strip numbers int fwd_table[128] = { 65, 67, 69, 71, 73, 75, 77, 79, 81, 83, 85, 87, 89, 91, 93, 95, 97, 99, 101, 103, 105, 107, 109, 111, 113, 115, 117, 119, 121, 123, 125, 127, 128, 126, 124, 122, 120, 118, 116, 114, 112, 110, 108, 106, 104, 102, 100, 98, 96, 94, 92, 90, 88, 86, 84, 82, 80, 78, 76, 74, 72, 70, 68, 66, 64, 62, 60, 58, 56, 54, 52, 50, 48, 46, 44, 42, 40, 38, 36, 34, 32, 30, 28, 26, 24, 22, 20, 18, 16, 14, 12, 10, 8, 6, 4, 2, 1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27, 29, 31, 33, 35, 37, 39, 41, 43, 45, 47, 49, 51, 53, 55, 57, 59, 61, 63 };</pre>	<pre>//table of silicon strip numbers ordered by ALICE128 readout order // note: values run from [1,...,128] int rev_table[128] = { 97, 96, 98, 95, 99, 94, 100, 93, 101, 92, 102, 91, 103, 90, 104, 89, 105, 88, 106, 87, 107, 86, 108, 85, 109, 84, 110, 83, 111, 82, 112, 81, 113, 80, 114, 79, 115, 78, 116, 77, 117, 76, 118, 75, 119, 74, 120, 73, 121, 72, 122, 71, 123, 70, 124, 69, 125, 68, 126, 67, 127, 66, 128, 65, 1, 64, 2, 63, 3, 62, 4, 61, 5, 60, 6, 59, 7, 58, 8, 57, 9, 56, 10, 55, 11, 54, 12, 53, 13, 52, 14, 51, 15, 50, 16, 49, 17, 48, 18, 47, 19, 46, 20, 45, 21, 44, 22, 43, 23, 42, 24, 41, 25, 40, 26, 39, 27, 38, 28, 37, 29, 36, 30, 35, 31, 34, 32, 33 };</pre>
--	---

Table 63: Alice128^{[4][5]} readout order

5.9 Slow-Control

Ladder-FPGA slow-control protocol is JTAG. In normal operation, the ladder-FPGA slow-control lines transit via the GBIC optical interface. The VME processor sends requests to VME registers in the slave-FPGA. The slave FPGA that emulates an ACT8990 (TI) that was in Corelis CVME-1149.1 card, translates the VME requests into JTAG bursts and drive them into the GBIC optical link.

For debug purpose, electronics group of SUBATECH has developed a general purpose debug card. When this debug card is connected to the ladder card, it is possible to use a STAR-SSD standard cable to drive the JTAG lines from a PC (see §5.12.3). A micro-switch on the debug card permits to select between the two sources of slow-control (GBIC optical-link or debug-connector)

At startup, only ladder FPGA is included in the JTAG chain. Table 64 bellow shows names, JTAG instruction, JTAG length and operation (dir) of all ladder-FPGA JTAG registers.

<i>name</i>	<i>nb bits</i>	<i>dir.</i>	<i>JTAG instruction</i>	<i>Description</i>
ladder_fpga_sc_roboclock_phase	24	rw	00001=0x01	Roboclock ^[1] phase adjustment
ladder_fpga_sc_reg_etat	22	r	00010=0x02	Status of the ladder-FPGA
ladder_fpga_sc_config	16	rw	00011=0x03	Configuration of the ladder-FPGA
ladder_fpga_sc_level_shifter_dac_values	20	rw	00100=0x04	Values for the dual output DAC
ladder_fpga_sc_etat_alims	16	r	00111=0x07	Status of the hybrids
ladder_fpga_sc_dr_mux_ref_latchup	2	rw	01000=0x08	voltage reference select for latchup detector
ladder_fpga_sc_rallumage	16	rw	01001=0x09	Switch ON or OFF hybrids power supplies
ladder_fpga_sc_bypass_hybride	16	rw	01011=0x0B	Remove hybrid output from JTAG chain
ladder_fpga_sc_version	32	r	01100=0x0C	Ladder-FPGA version date
ladder_fpga_sc_temperature	48	r	01110=0x0E	Ladder-FPGA temperature
ladder_fpga_sc_reg_identite	8	r	11011=0x1B	Identity of the ladder-FPGA
ladder_fpga_sc_reg_bypass	1	rw	11111=0x1F	Bypass of the ladder-FPGA

Table 64: ladder-FPGA slow-control registers

All instructions that are not described in Table 64, address the bypass register.

Dir="rw" means that reading of the previous value and writing of the new value are done at the same time.

Read-only is not possible for rw registers!.

Dir="r" means that only reading of the previous value is done.

24 bits
ladder_fpga_sc_roboclock_phase

Table 65: ladder-FPGA slow-control register (roboclock_phase)^[1]

<i>clock40MHz_fpga_phase</i>	<i>ladder_fpga_sc_roboclock_phase(23 downto 22)</i>			
	<i>11</i>	<i>10</i>	<i>01</i>	<i>00</i>
<i>ladder_fpga_sc_roboclock_phase(21 downto 20)</i>	<i>11</i>	+4,2ns	+1,1ns	+1,1ns
	<i>10</i>	+3,2ns	0	0
	<i>01</i>	+3,2ns	0	0
	<i>00</i>	+2,1ns	-1,1ns	-1,1ns

Table 66: ladder-FPGA slow-control register (roboclock_phase)^[1] effect on clock40MHz_fpga

<i>clock40MHz_ser phase</i>	<i>ladder_fpga_sc_roboclock_phase(19 downto 18)</i>				
<i>ladder_fpga_sc_roboclock_phase(17 downto 16)</i>		11	10	01	00
	11	+4,2ns	+1,1ns	+1,1ns	-2,1ns
	10	+3,2ns	0	0	-3,2ns
	01	+3,2ns	0	0	-3,2ns
	00	+2,1ns	-1,1ns	-1,1ns	-4,2ns

Table 67: ladder-FPGA slow-control register (roboclock_phase)^[1] effect on clock40MHz_ser

<i>clock80MHz-adc0 and clock80MHz-adc1 phase</i>	<i>ladder_fpga_sc_roboclock_phase(15 downto 14)</i>				
<i>ladder_fpga_sc_roboclock_phase(13 downto 12)</i>		11	10	01	00
	11	+3,4ns	+0,9ns	+0,9ns	-1,7ns
	10	+2,6ns	0	0	-2,6ns
	01	+2,6ns	0	0	-2,6ns
	00	+1,7ns	-0,9ns	-0,9ns	-3,4ns

Table 68: ladder-FPGA slow-control register (roboclock_phase)^[1] effect on clock80MHz-adc0 and clock80MHz-adc1

<i>clock80MHz-adc2 and clock80MHz-adc3 phase</i>	<i>ladder_fpga_sc_roboclock_phase(11 downto 10)</i>				
<i>ladder_fpga_sc_roboclock_phase(9 downto 8)</i>		11	10	01	00
	11	+3,4ns	+0,9ns	+0,9ns	-1,7ns
	10	+2,6ns	0	0	-2,6ns
	01	+2,6ns	0	0	-2,6ns
	00	+1,7ns	-0,9ns	-0,9ns	-3,4ns

Table 69: ladder-FPGA slow-control register (roboclock_phase)^[1] effect on clock80MHz-adc2 and clock80MHz-adc3

<i>clock80MHz-adc4 and clock80MHz-adc5 phase</i>	<i>ladder_fpga_sc_roboclock_phase(7 downto 6)</i>				
<i>ladder_fpga_sc_roboclock_phase(5 downto 4)</i>		11	10	01	00
	11	/2	+1,7ns	+1,7ns	-3,4ns
	10	+5,2ns	0	0	-5,2ns
	01	+5,2ns	0	0	-5,2ns
	00	+3,4ns	-1,7ns	-1,7ns	/4

Table 70: ladder-FPGA slow-control register (roboclock_phase)^[1] effect on clock80MHz-adc4 and clock80MHz-adc5

<i>clock80MHz-adc6 and clock80MHz-adc7 phase</i>	<i>ladder_fpga_sc_roboclock_phase(3 downto 2)</i>				
<i>ladder_fpga_sc_roboclock_phase(1 downto 0)</i>		11	10	01	00
	11	/2	+1,7ns	+1,7ns	-3,4ns
	10	+5,2ns	0	0	-5,2ns
	01	+5,2ns	0	0	-5,2ns
	00	+3,4ns	-1,7ns	-1,7ns	inv

Table 71: ladder-FPGA slow-control register (roboclock_phase)^[1] effect on clock80MHz-adc6 and clock80MHz-adc7

1 bit	1 bit	1 bit	1 bit	1 bit	1 bit	1 bit	1 bit	1 bit	1 bit	1 bit	1 bit	1 bit	1 bit	1 bit	1 bit	6 serial # bits
spare switch	fpga serdes	sc serdes	xtal en	debug present	crc error	HV side	hold	test	Switchover uses 36.4 MHz from xtal	switchover locked	switchover forced to use xtal	bad 36.4 MHz from serdes	bad 36.4 MHz from xtal	? latchup ?	end of temperature conversion	ladder card serial #

Table 72: ladder-FPGA slow-control register (status)

"hv side" bit indicates, when asserted, that TEST signal and each bit of ADC data are complemented.

8 bits	1 bit	1 level-shifter bit	1 bit	1 bit	4 hybrid # bits
free	HV side	Load DAC channels	force switchover to use 36.4MHz from xtal	jtag chain with selected hybrid	# of hybrid in jtag chain

Table 73: ladder-FPGA slow-control register (config)

Load DAC channel must be cycled to perform load (rising edge is used)

When "jtag chain with selected hybrid" is asserted, The hybrid selected by "# of hybrid in jtag chain" is included in the JTAG chain.

"hv side" bit, when asserted, forces the ladder FPGA to complement TEST signal and each bit of ADC data.

10 DAC bits	10 DAC bits
Level-shifter DAC channel B	Level-shifter DAC channel A

Table 74: ladder-FPGA slow-control register (level-shifter DAC values)

Load DAC channel must be cycled to perform load (rising edge is used)

1 bit	1 bit	1 bit	1 bit	1 bit	1 bit	1 bit	1 bit	1 bit	1 bit	1 bit	1 bit	1 bit	1 bit	1 bit	1 bit
hybrid 15 is ON	hybrid 14 is ON	hybrid 13 is ON	hybrid 12 is ON	hybrid 11 is ON	hybrid 10 is ON	hybrid 9 is ON	hybrid 8 is ON	hybrid 7 is ON	hybrid 6 is ON	hybrid 5 is ON	hybrid 4 is ON	hybrid 3 is ON	hybrid 2 is ON	hybrid 1 is ON	hybrid 0 is ON

Table 75: ladder-FPGA slow-control register (etat alim)

2 select bits
00 0.84V
01 0.63V
10 0.42V
11 0.21V

Table 76: ladder-FPGA slow-control register (mux ref latchup)

1 bit	1 bit	1 bit	1 bit	1 bit	1 bit	1 bit	1 bit	1 bit	1 bit	1 bit	1 bit	1 bit	1 bit	1 bit	1 bit
switch hybrid 15 ON	switch hybrid 14 ON	switch hybrid 13 ON	switch hybrid 12 ON	switch hybrid 11 ON	switch hybrid 10 ON	switch hybrid 9 ON	switch hybrid 8 ON	switch hybrid 7 ON	switch hybrid 6 ON	switch hybrid 5 ON	switch hybrid 4 ON	switch hybrid 3 ON	switch hybrid 2 ON	switch hybrid 1 ON	switch hybrid 0 ON

Table 77: ladder-FPGA slow-control register (rallumage)

1 bit	1 bit	1 bit	1 bit	1 bit	1 bit	1 bit	1 bit	1 bit	1 bit	1 bit	1 bit	1 bit	1 bit	1 bit	1 bit
hybrid 15 bypassed	hybrid 14 bypassed	hybrid 13 bypassed	hybrid 12 bypassed	hybrid 11 bypassed	hybrid 10 bypassed	hybrid 9 bypassed	hybrid 8 bypassed	hybrid 7 bypassed	hybrid 6 bypassed	hybrid 5 bypassed	hybrid 4 bypassed	hybrid 3 bypassed	hybrid 2 bypassed	hybrid 1 bypassed	hybrid 0 bypassed

Table 78: ladder-FPGA slow-control register (bypass)

This register should not be used anymore.

8 day bits	8 month bits	16 year bits
Day (DD format)	Month (MM format)	Year (YYYY format)

Table 79: ladder-FPGA slow-control register (version)

12 temperature bits	12 temperature bits	12 temperature bits	12 temperature bits
Sensor 3 temperature (in °K/10) on debug part (external)	Sensor 2 temperature (in °K/10) on connector part (internal)	Sensor 1 temperature (in °K/10) on serdes part (internal)	Sensor 0 temperature (in °K/10) on ADC part (internal)

Table 80: ladder-FPGA slow-control register (temperature)

4 ID bits	1 bit	3 addr bits
0xB	free	ladder address

Table 81: ladder-FPGA slow-control register (identity)

1 bit
bypass

Table 82: ladder-FPGA slow-control register (bypass)

When "jtag chain with selected hybrid" (Table 73) is asserted, The hybrid selected by "# of hybrid in jtag chain" (Table 73) is included in the JTAG chain. **The corresponding hybrid must have been powered up (Table 77) before including it in the JTAG chain.**

register	Ladder-FPGA Table 64	Hybrid #X Alice128 #0	Hybrid #X Alice128 #1	Hybrid #X Alice128 #2	Hybrid #X Alice128 #3	Hybrid #X Alice128 #4	Hybrid #X Alice128 #5	Hybrid #X Costar ^[27]
----------	-------------------------	--------------------------	--------------------------	--------------------------	--------------------------	--------------------------	--------------------------	-------------------------------------

Table 83: JTAG slow-control chain when "jtag chain with selected hybrid" is asserted

pos	4 ID bits	4 bits	Number of words
0	0x?	?	1

Table 84: Alice128^{[4][5]} slow-control registers (identite)

Alice128^{[4][5]} chip has no identity register.

The documentation we have about the JTAG registers in the Alice128 chip is very poor. Information have to be extracted from the old slow-control developed by David BONNET (IRES/Strasbourg) that was used in STAR control room from 2002 to 2007 and from the old test bench for module developed by Cyril DRANCOURT (now LAPP/Annecy) that has been used at Subatech since 2000.

pos	4 ID bits	4 bits	Number of words
0	0xA	?	1

Table 85: Costar^[27] slow-control registers (identite)

name origin

TCK	slave												
TMS	slave												
TDI	slave												
TDO	ladder												
bit #	//	0	1	2	3	4	//	0	1	//	n-2	n-1	
register	//	Instruction (5 bits)					//	Data (n bits)					

Table 86: JTAG transaction timing

i = instruction; d = new data; o = old data

5.10.1 Overview

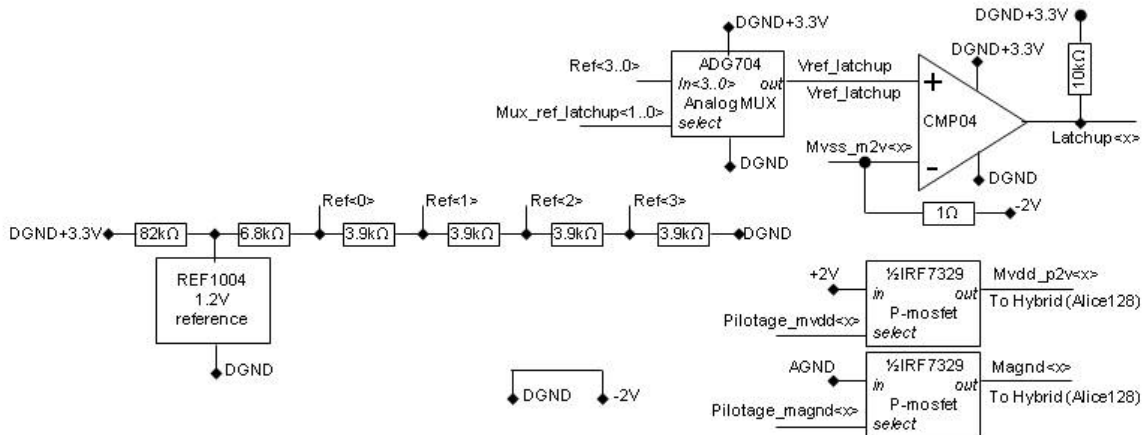


Figure 29: schematics of the latch up detection in the ladder card ^{[17][18][19]}

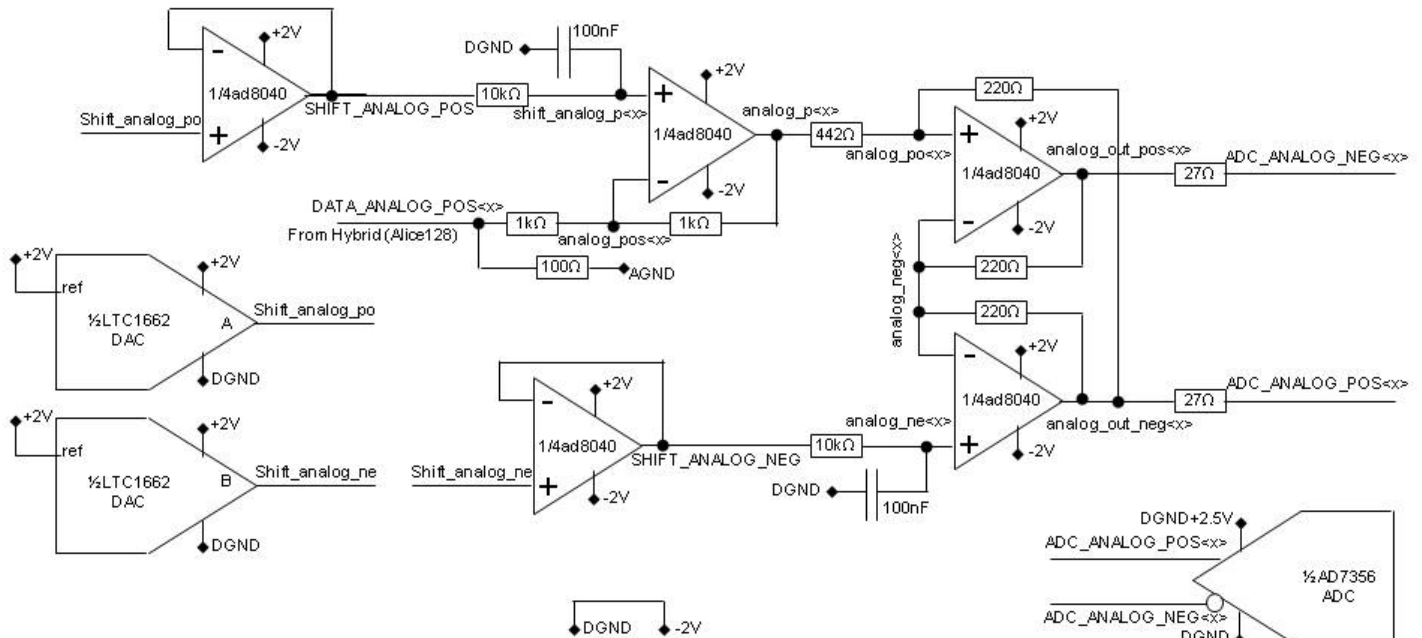


Figure 30: schematics of the analog part in the ladder card ^{[20][21][22]}

A spreadsheet to calculate ADC output value vs. DAC register input values, Alice128 analog voltage and power supply voltages can be found in [0].
 maximum output of Alice128^{[4][5]} chip: $\pm 13\text{MIP} = \pm 624\text{mV}$

5.10.2 Connectors for flex cables (30 pins)

5.10.3 Level-shifter (from Alice128^{[4][5]} to ADC)

5.10.3.1 Overview

The level shifter job consists to shift the analog signal from the Alice128 output range to the ADC input range and to convert the shifted analog signal into differential analog signal. Two DACs permit to set the base level according to the foreseen sweep of the signal.

- First stage (one Op-Amp) is an adder. It is used to shift the analog signal from the Alice128 output range (-2V; +2V) to the ADC input range (-2V; 0,5V).

The Alice128 output is connected to one input of this stage.

One channel output of the Digital to Analog Converter (DAC), controlled via one JTAG register, is connected to the other input of this stage. It permits to set the base level according to the foreseen sweep of the signal: positive pulse on one side of the ladder, negative pulse on the other side of the ladder.

- Second stage (two Op-Amps) convert the single ended signal coming out of the first stage into differential signal that is sent to the Analog to Digital Converter (ADC) and to shift the differential signal into the ADC input range (-2V; 0,5V).

The output of the first stage is connected to one input of this stage.

The output of the other channel of the Digital to Analog Converter (DAC), controlled via one JTAG register, is connected to the other input of this stage. It permits to set the base level according to the foreseen sweep of the signal: positive pulse on one side of the ladder, negative pulse on the other side of the ladder.

Node equations:

$$\bullet \frac{ana\ log_p - ana\ log_pos}{1k\Omega} + \frac{DATA_ANALOG_POS - ana\ log_pos}{1k\Omega} = 0$$

Équation 1: level shifter, first stage, negative input

$$\bullet \frac{ana\ log_out_neg - ana\ log_po}{220\Omega} + \frac{ana\ log_p - ana\ log_po}{442\Omega} = 0$$

Équation 2: level shifter, first stage to second stage

$$\bullet \frac{ana\ log_out_pos - ana\ log_neg}{220\Omega} + \frac{ana\ log_out_neg - ana\ log_neg}{220\Omega} = 0$$

Équation 3: level shifter, second stage, negative inputs

If operational amplifiers AD8040^[21] remain in linear domain (mainly inputs and outputs remain between $\pm 2V$ power supplies) and DC input currents ($< 2\mu A$) are neglected :

$$\bullet ana\ log_pos = shift_ana\ log_p = SHIFT_ANALOG_POS$$

Équation 4: level shifter, first stage, inputs

$$\bullet ana\ log_po = ana\ log_neg = ana\ log_ne$$

Équation 5: level shifter, second stage, inputs

If DC input current of AD7356^[20] ($< 5\mu A$) is neglected

$$\bullet ADC_ANALOG_POS = ana\ log_out_neg \quad \text{and} \quad ADC_ANALOG_NEG = ana\ log_out_pos$$

Équation 6: level shifter, second stage, outputs

Results:

$$\bullet ana\ log_p = 2 * SHIFT_ANALOG_POS - DATA_ANALOG_POS$$

Équation 7: level shifter, first stage, output

$$\bullet ADC_ANALOG_NEG = \frac{1}{2} * SHIFT_ANALOG_NEG + SHIFT_ANALOG_POS - \frac{DATA_ANALOG_POS}{2}$$

Équation 8: level shifter, ADC, negative input ADC_ANALOG_NEG

$$\bullet ADC_ANALOG_POS = \frac{3}{2} * SHIFT_ANALOG_NEG - SHIFT_ANALOG_POS + \frac{DATA_ANALOG_POS}{2}$$

Équation 9: level shifter, ADC, positive input ADC_ANALOG_POS

$$\bullet ADC(in+) = ADC_ANALOG_POS - VSS_M2V$$

Équation 10: ADC(in+)

$$\bullet ADC(in-) = ADC_ANALOG_NEG - VSS_M2V$$

Équation 11: ADC(in-)

5.10.3.2 Level-shifter DAC (Linear Technology LTC1662 dual 10 bit DAQ)^[22]

5.10.3.2.1 Level-shifter DAC configuration

At power-up, the default values are loaded into the slow-control register (Table 74), the DAC is woke-up, the slow-control register values are loaded into the two channels and the output of both channels are updated. At any time, the slow-control can change the values in the slow-control register (Table 74) and ask for an update of the DAC outputs by cycling bit "load DAC values" in config register (Table 73)

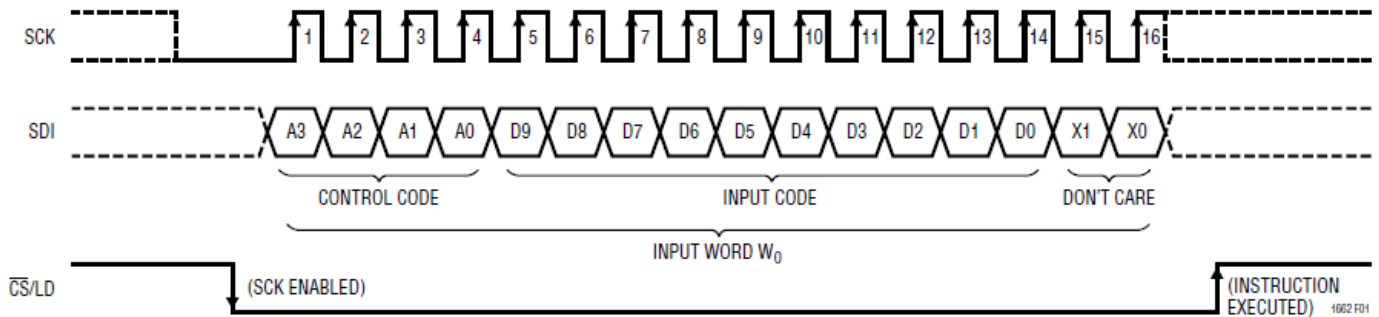


Figure 31: Diagram of operation for level-shifter DAC serial interface.^[22]

CONTROL				INPUT REGISTER STATUS	DAC REGISTER STATUS	POWER-DOWN STATUS (SLEEP/WAKE)	COMMENTS
A3	A2	A1	A0				
0	0	0	0	No Change	No Update	No Change	No Operation. Power-Down Status Unchanged (Part Stays In Wake or Sleep Mode)
0	0	0	1	Load DAC A	No Update	No Change	Load Input Register A with Data. DAC Outputs Unchanged. Power-Down Status Unchanged
0	0	1	0	Load DAC B	No Update	No Change	Load Input Register B with Data. DAC Outputs Unchanged. Power-Down Status Unchanged
1	0	0	0	No Change	Update Outputs	Wake	Load Both DAC Regs with Existing Contents of Input Regs. Outputs Update. Part Wakes Up
1	0	0	1	Load DAC A	Update Outputs	Wake	Load Input Reg A. Load DAC Regs with New Contents of Input Reg A and Existing Contents of Reg B. Outputs Update. Part Wakes Up
1	0	1	0	Load DAC B	Update Outputs	Wake	Load Input Reg B. Load DAC Regs with Existing Contents of Input Reg A and New Contents of Reg B. Outputs Update. Part Wakes Up
1	1	0	1	No Change	No Update	Wake	Part Wakes Up. Input and DAC Regs Unchanged. DAC Outputs Reflect Existing Contents of DAC Regs
1	1	1	0	No Change	No Update	Sleep	Part Goes to Sleep. Input and DAC Regs Unchanged. DAC Outputs Set to High Impedance State
1	1	1	1	Load DACs A, B with Same 10-Bit Code	Update Outputs	Wake	Load Both Input Regs. Load Both DAC Regs with New Contents of Input Regs. Outputs Update. Part Wakes Up

Note: All control codes other than those shown are undefined and not subject to test.

Figure 32: Control codes for operation of level-shifter DAC serial interface.^[22]

$$\bullet \text{ DACana log}_{\text{out}} = \frac{VDD_P2V - VSS_M2V}{1023} * \text{DACregister}$$

Équation 12: Output voltage of DAC with respect to its analog ground vs. register value^[22]

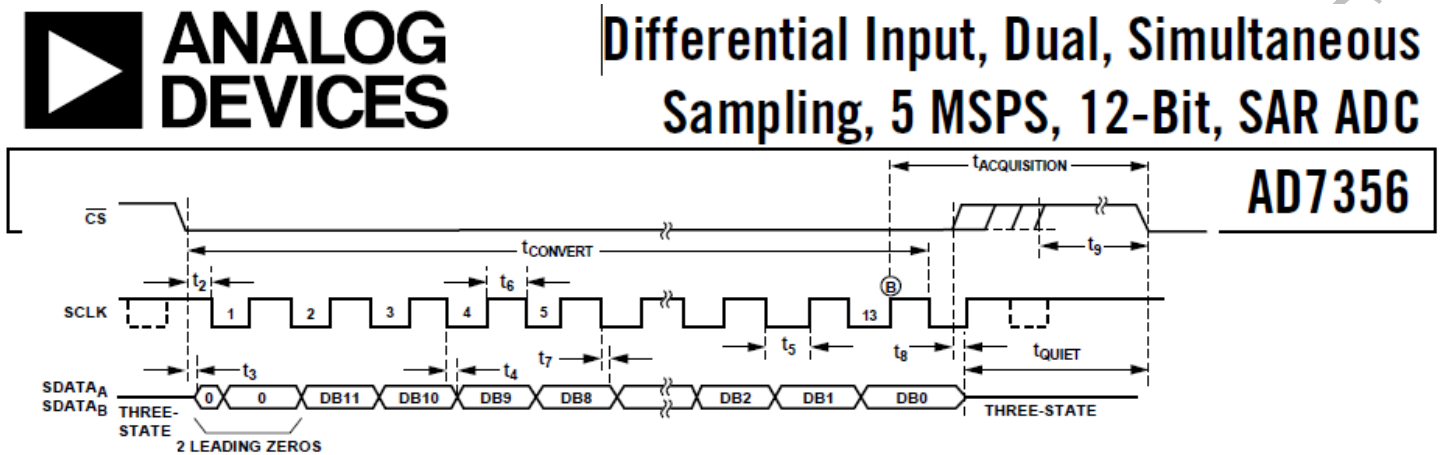
$$\bullet \text{ SHIFT_ANALOG_NEG} = VSS_M2V + \frac{VDD_P2V - VSS_M2V}{1023} * \text{DAC} - B''$$

Équation 13: Output voltage of DAC-B with respect to analog ground vs. register value

$$\bullet \text{ } SHIFT_ANALOG_POS = VSS_M2V + \frac{VDD_P2V - VSS_M2V}{1023} * "DAC - A"$$

Équation 14: Output voltage of DAC-A with respect to analog ground vs. register value

5.10.4 ADC (Analog Devices AD7356 dual 12 bit ADC with serial output)^[20]

Figure 33: Diagram of operation for serial ADC.^[20]

$V_{DD} = 2.5 \text{ V} \pm 10\%$, $V_{DRIVE} = 2.25 \text{ V}$ to 3.6 V , internal reference = 2.048 V , $T_A = T_{MAX}$ to T_{MIN}^1 , unless otherwise noted.

Table 3. **AD7356**

Parameter	Limit at T_{MIN} , T_{MAX}	Unit	Description
f_{SCLK}	50 80	kHz min MHz max	
$t_{CONVERT}$	$t_2 + 13 \times t_{SCLK}$	ns max	$t_{SCLK} = 1/f_{SCLK}$
t_{QUIET}	5	ns min	Minimum time between end of serial read and next falling edge of \overline{CS}
t_2	5	ns min	\overline{CS} to SCLK setup time
t_3^2	6	ns max	Delay from \overline{CS} until $SDATA_A$ and $SDATA_B$ are three-state disabled
$t_4^{2,3}$	12.5 11 9.5 9	ns max	Data access time after SCLK falling edge $1.8 \text{ V} \leq V_{DRIVE} < 2.25 \text{ V}$ $2.25 \text{ V} \leq V_{DRIVE} < 2.75 \text{ V}$ $2.75 \text{ V} \leq V_{DRIVE} < 3.3 \text{ V}$ $3.3 \text{ V} \leq V_{DRIVE} \leq 3.6 \text{ V}$
t_5	5	ns min	SCLK low pulse width
t_6	5	ns min	SCLK high pulse width
t_7^2	3.5	ns min	SCLK to data valid hold time
t_8^2	9.5	ns max	\overline{CS} rising edge to $SDATA_A$, $SDATA_B$ high impedance
t_9	5	ns min	\overline{CS} rising edge to falling edge pulse width
t_{10}^2	4.5 9.5	ns min ns max	SCLK falling edge to $SDATA_A$, $SDATA_B$ high impedance SCLK falling edge to $SDATA_A$, $SDATA_B$ high impedance

¹ Temperature ranges are as follows: Y Grade: -40°C to $+125^\circ\text{C}$; B Grade: -40°C to $+85^\circ\text{C}$.

² Specified with a load capacitance of 10 pF on $SDATA_A$ and $SDATA_B$.

³ The time required for the output to cross 0.4 V or 2.4 V .

Figure 34: Timing of operation for serial ADC.^[20]

$$\bullet V_{ref} = 2.048 \text{ V} \pm 0.5\% (\pm 0.25\% @ 25^\circ\text{C})$$

Équation 15: ADC internal reference voltage.^[20]

$$\bullet LSB = \frac{2 * V_{ref}}{4096} \cong 1 \text{ mV}$$

Équation 16: ADC resolution when 12 bits are used.^[20]

- $ADC(diff) = ADC(in+) - ADC(in-)$

Équation 17: $ADC(diff)^{[20]}$

- $ADC(com) = \frac{ADC(in+) + ADC(in-)}{2}$

Équation 18: $ADC(com)^{[20]}$

- $ADCvalue = 2048 + ADC(diff) / LSB$

Équation 19: $ADC\ value\ (12\ bits)\ vs.\ ADC\ differential\ inputs.^{[20]}$

- $ADC(diff) = SHIFT_ANALOG_NEG - 2 * SHIFT_ANALOG_POS + DATA_ANALOG_POS$

Équation 20: $ADC(diff)$

- $ADC(com) = SHIFT_ANALOG_NEG - VSS_M2V$

Équation 21: $ADC(com)$

When level-shifter operational amplifiers, DACs and ADC are set in their linear domains:

$$ADCvalue = 2048 + \frac{2048}{Vref} * [DATA_ANALOG_POS] + \frac{2048}{Vref} * \left[\frac{VDD_P2V - VSS_M2V}{1023} * ("DAC - B" - 2 * "DAC - A") - VSS_M2V \right]$$

Équation 22: $ADC\ value\ (12\ bits)\ vs.\ DAC-A, DAC-B, Vin, VDD_P2V, VSS_M2V\ and\ Vref.$

The physics signal being a negative pulse on P-side of the detector and a positive pulse on N-side of the detector, ADC output data bits are complemented when the ladder card is connected to P-side of the detector, in order to avoid developing and maintaining different software for P-side and N-side when computing the pedestal and doing zero suppression. Data is complemented in ladder FPGA when bit "hv side" in "Table 73: ladder-FPGA slow-control register (config)" is asserted (=1).

5.11 Ladder-FPGA

The firmware for this ALTERA^[8] CycloneIII EP3C16F484C6^[9] FPGA was developed in VHDL using Synplify_pro^[10] as a synthesizer, Quartus II^[11] as place & route software and ModelSim-Altera^[12] as a simulation software. Complete project can be found in [0]

5.11.1 Ladder-FPGA configuration

In normal operation, the ladder-FPGA is configured in PS mode. Configuration lines transit via the GBIC optical interface. The configuration serial PROM (EPSC4) is on the RDO card, connected to the slave FPGA. The VME processor initiates the configuration by sending a "transfer configuration data to ladder FPGA" command to the slave FPGA. The VME processor can load new configuration data via the slave FPGA, into the EPSC4 (see §4.12.2).

For debug purpose, electronics group of SUBATECH has developed a general purpose debug card. When this debug card is connected to the ladder card, it is possible to use a standard cable to configure the ladder-FPGA in JTAG mode from a PC (see §5.12.2). A micro-switch on the debug card permits to select between the two sources of configuration (optical-link or debug-connector)

5.11.2 Ladder-FPGA detection of configuration error due to SEU (Single Event Upset)^[9]

SEU can generate some bit flip in the internal CRAM (Configuration Random Access Memory) of the CycloneIII^[9] FPGA.

Altera^[8] has implemented in its FPGAs a CRC (Cyclical Redundancy Check) functionality that is able to detect and report configuration error.

When detection of configuration error is enabled, CRC_ERROR signal is connected to pin L21 of the CycloneIII^[9] FPGA.

Christophe-M.-A. Renard, Subatech, IN2P3/CNRS-L'UNAM, Nantes, F44307, France

73/121

Micheal-J. LeVine, BNL, Upton, NY, USA

Stéphane Bouvier, Subatech, IN2P3/CNRS-L'UNAM, Nantes, F44307, France

Slow-control can check `crc_error` signal in bit 16 of status register [Table 72: *ladder-FPGA slow-control register (status)*]

5.11.3 Ladder-FPGA event controller

To prevent the system from hanging if one Alice128 chip does not return the token, the ladder FPGA counts the RCLK ticks and generates a new token for each hybrid.

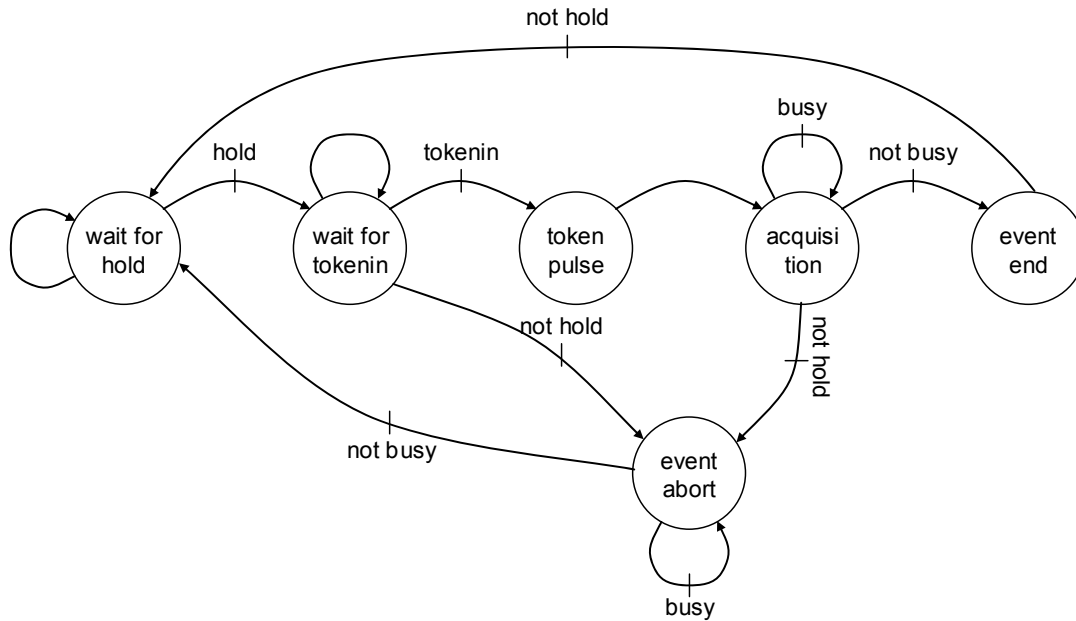


Figure 35: *ladder-FPGA event controller state machine*

5.11.4 Ladder-FPGA data packer

The data packer job consists to compress 256 bits into 168 bits: 16 ADC words of 16 bit each are packed into 8 words in a FIFO that has 16 bit wide words. There is no loss of data.

The data packer gets the serial flux from the 16 ADCs. Each of the serial ADC fluxes is composed of 16 bit wide words where only 10 bits are useful for the SSD. The flux compactor transmits only the relevant part of the 16 serial fluxes to the five 16 bit wide packer words. The packer concatenates bits from two packer words to create the 20 bit words that it writes into the 21 bit wide FIFO (this FIFO is named FIFO21). The 21st bit is used to tag the first word.

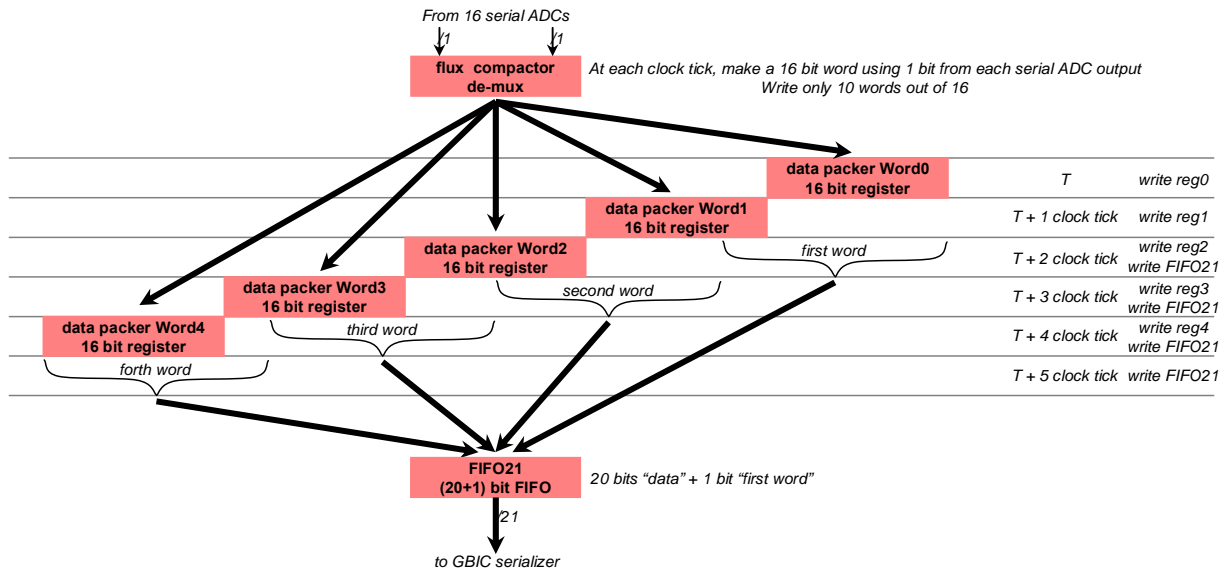


Figure 36: ladder-FPGA data packer

80MHz clock																				
ADC cs input																				
ADC serial data out	HZ	0	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0	0	HZ	HZ	0	D11	D10
Flux compactor memorized data	D10 D9 D8 D7 D6 D5 D4 D3 D2 D1																			

Table 87: ladder-FPGA data packer flux compactor

HZ = High Z

	FIFO21 word 0	FIFO21 word 1	FIFO21 word 2	FIFO21 word 3	FIFO21 word 4	FIFO21 word 5	FIFO21 word 6	FIFO21 word 7
FIFO21 bit 20	1	0	0	0	1	0	0	0
FIFO21 bit 19	Ade3 bit 8	Ade7 bit 7	Ade11 bit 6	Ade15 bit 5	Ade3 bit 3	Ade7 bit 2	Ade11 bit 1	Ade15 bit 0
FIFO21 bit 18	Ade2 bit 8	Ade6 bit 7	Ade10 bit 6	Ade14 bit 5	Ade2 bit 3	Ade6 bit 2	Ade10 bit 1	Ade14 bit 0
FIFO21 bit 17	Ade1 bit 8	Ade5 bit 7	Ade9 bit 6	Ade13 bit 5	Ade1 bit 3	Ade5 bit 2	Ade9 bit 1	Ade13 bit 0
FIFO21 bit 16	Ade0 bit 8	Ade4 bit 7	Ade8 bit 6	Ade12 bit 5	Ade0 bit 3	Ade4 bit 2	Ade8 bit 1	Ade12 bit 0
FIFO21 bit 15	Ade15 bit 9	Ade3 bit 7	Ade7 bit 6	Ade11 bit 5	Ade15 bit 4	Ade3 bit 2	Ade7 bit 1	Ade11 bit 0
FIFO21 bit 14	Ade14 bit 9	Ade2 bit 7	Ade6 bit 6	Ade10 bit 5	Ade14 bit 4	Ade2 bit 2	Ade6 bit 1	Ade10 bit 0
FIFO21 bit 13	Ade13 bit 9	Ade1 bit 7	Ade5 bit 6	Ade9 bit 5	Ade13 bit 4	Ade1 bit 2	Ade5 bit 1	Ade9 bit 0
FIFO21 bit 12	Ade12 bit 9	Ade0 bit 7	Ade4 bit 6	Ade8 bit 5	Ade12 bit 4	Ade0 bit 2	Ade4 bit 1	Ade8 bit 0
FIFO21 bit 11	Ade11 bit 9	Ade15 bit 8	Ade3 bit 6	Ade7 bit 5	Ade11 bit 4	Ade15 bit 3	Ade3 bit 1	Ade7 bit 0
FIFO21 bit 10	Ade10 bit 9	Ade14 bit 8	Ade2 bit 6	Ade6 bit 5	Ade10 bit 4	Ade14 bit 3	Ade2 bit 1	Ade6 bit 0
FIFO21 bit 9	Ade9 bit 9	Ade13 bit 8	Ade1 bit 6	Ade5 bit 5	Ade9 bit 4	Ade13 bit 3	Ade1 bit 1	Ade5 bit 0
FIFO21 bit 8	Ade8 bit 9	Ade12 bit 8	Ade0 bit 6	Ade4 bit 5	Ade8 bit 4	Ade12 bit 3	Ade0 bit 1	Ade4 bit 0
FIFO21 bit 7	Ade7 bit 9	Ade11 bit 8	Ade15 bit 7	Ade3 bit 5	Ade7 bit 4	Ade11 bit 3	Ade15 bit 2	Ade3 bit 0
FIFO21 bit 6	Ade6 bit 9	Ade10 bit 8	Ade14 bit 7	Ade2 bit 5	Ade6 bit 4	Ade10 bit 3	Ade14 bit 2	Ade2 bit 0
FIFO21 bit 5	Ade5 bit 9	Ade9 bit 8	Ade13 bit 7	Ade1 bit 5	Ade5 bit 4	Ade9 bit 3	Ade13 bit 2	Ade1 bit 0
FIFO21 bit 4	Ade4 bit 9	Ade8 bit 8	Ade12 bit 7	Ade0 bit 5	Ade4 bit 4	Ade8 bit 3	Ade12 bit 2	Ade0 bit 0
FIFO21 bit 3	Ade3 bit 9	Ade7 bit 8	Ade11 bit 7	Ade15 bit 6	Ade3 bit 4	Ade7 bit 3	Ade11 bit 2	Ade15 bit 1
FIFO21 bit 2	Ade2 bit 9	Ade6 bit 8	Ade10 bit 7	Ade14 bit 6	Ade2 bit 4	Ade6 bit 3	Ade10 bit 2	Ade14 bit 1
FIFO21 bit 1	Ade1 bit 9	Ade5 bit 8	Ade9 bit 7	Ade13 bit 6	Ade1 bit 4	Ade5 bit 3	Ade9 bit 2	Ade13 bit 1
FIFO21 bit 0	Ade0 bit 9	Ade4 bit 8	Ade8 bit 7	Ade12 bit 6	Ade0 bit 4	Ade4 bit 3	Ade8 bit 2	Ade12 bit 1

Table 88: ladder-FPGA data packer FIFO21 data organization

5.12 Debug interface

⚠ Ladder card analog reference (AGND) being connected to high voltage (~50V) on one side of the ladder (P side), particular care must be provided to electrically isolate the debug interfaces

direction	name	top	bottom	name	direction
gnd	Gnd	1	2	Gnd	gnd
lad2pc	LED ok Gnd	3	4	LED ok Gnd	lad2pc
gnd	Gnd	5	6	Gnd	gnd
bidir	usb_data(0)	7	8	usb_data(0)	bidir
gnd	Gnd	9	10	Gnd	gnd
Power2dbg	3.3V	11	12	3.3V	Power2dbg
bidir	usb_data(4)	13	14	usb_data(4)	bidir
bidir	usb_data(2)	15	16	usb_data(2)	bidir
gnd	Gnd	17	18	Gnd	gnd
bidir	usb_data(1)	19	20	usb_data(1)	bidir
lad2pc	Led6	21	22	Led6	lad2pc
lad2pc	Led5	23	24	Led5	lad2pc
lad2pc	Led4	25	26	Led4	lad2pc
gnd	Gnd	27	28	Gnd	gnd
Power2dbg	3.3V	29	30	3.3V	Power2dbg
gnd	Gnd	31	32	Gnd	gnd
pc2lad	Dbg_pres	33	34	Dbg_pres	pc2lad
pc2lad	Usb_rx_empty	35	36	Usb_rx_empty	pc2lad
gnd	Gnd	37	38	Gnd	gnd
pc2lad	Usb_tx_full	39	40	Usb_tx_full	pc2lad
pc2lad	Usb_present	41	42	Usb_present	pc2lad
Power2dbg	3.3V	43	44	3.3V	Power2dbg
gnd	Gnd	45	46	Gnd	gnd
pc2lad	Fpga_jtag_tms	47	48	Fpga_jtag_tms	pc2lad
pc2lad	Usb_ready_n	49	50	Usb_ready_n	pc2lad
gnd	Gnd	51	52	Gnd	gnd
lad2pc	Conf_fib	53	54	Conf_fib	lad2pc
lad2pc	Usb_rd_n	55	56	Usb_rd_n	lad2pc
lad2pc	Usb_wr_n	57	58	Usb_wr_n	lad2pc
Power2dbg	3.3V	59	60	3.3V	Power2dbg

Table 89: *Edge connector to debug interface*

Colors in Table 89 do not use general typographic convention defined in §2.1 of this document

⚠ Ladder card analog reference (AGND) being connected to high voltage (~50V) on one side of the ladder (P side), particular care must be provided to electrically isolate the debug interfaces

5.12.2 Debug interface: FPGA configuration (FPGA-Blaster)

⚠ Ladder card analog reference (AGND) being connected to high voltage (~50V) on one side of the ladder (P side), particular care must be provided to electrically isolate the debug interfaces

It is connected to the serial JTAG configuration interface of the ladder FPGA.

5.12.3 Debug interface: JTAG slow-control

⚠ Ladder card analog reference (AGND) being connected to high voltage (~50V) on one side of the ladder (P side), particular care must be provided to electrically isolate the debug interfaces

It is connected to the serial JTAG slow-control interface of the ladder FPGA

5.12.4 Debug interface: 8 bit FIFO

This implementation is complying with LAL USB Interface protocol^[30]

5.13 Ladder card printed circuit

5.13.1 Ladder card printed circuit cross-section

		<i>rigid</i>	<i>flex</i>	<i>rigid</i>	<i>flex</i>	<i>rigid</i>
1 top	conductor	top		top		top
	dielec					
2 Agnd_Gnd_Gnd	plane	Agnd		Gnd		Gnd
	dielec					
3 int1_int1gnd_int1	conductor	int1		int1+Gnd		int1
	dielec					
4 Avdd_25V_Gnd	plane	Avdd		2.5V		Gnd
	dielec					
5 int2	conductor	int2		int2		int2
	dielec					
6 int3	conductor	int3		int3		int3
	dielec					
7 flexGnd	plane	Gnd	Gnd	Gnd	Gnd	Gnd
	dielec					
8 flex	conductor	flex	flex	flex	flex	flex
	dielec					
9 33V_33V_33	plane	3.3V		3.3V		3.3
	dielec					
10 int5	conductor	int5		int5		int5
	dielec					
11 Agnd_Avdd25V_Gnd	plane	Agnd		Avdd+2.5		Gnd
	dielec					
12 int6_int6Agnd_int6	conductor	int6		int6+Agnd		int6
	dielec					
13 Agnd_Gnd12V_Gnd	plane	Agnd		Gnd+1.2V		Gnd
	dielec					
14 bottom	conductor	bottom		bottom		bottom

Table 90: ladder card cross section

3.3V and 3.3 come out of two different voltage regulators

2.5V and 2.5 come out of two different voltage regulators

5.13.2 Ladder card printed circuit board dimensions

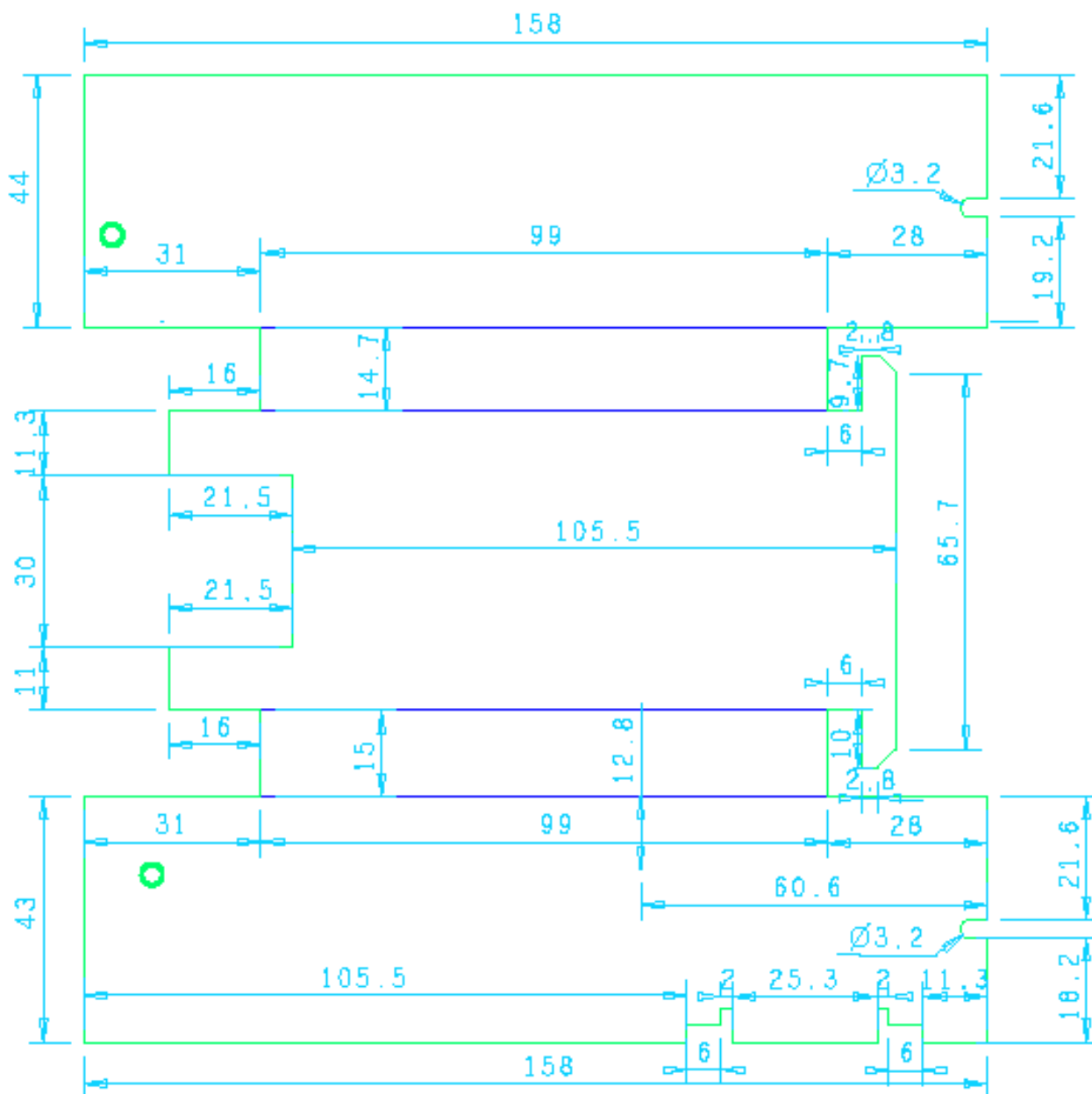


Figure 37: ladder card preproduction dimensions (in mm)

5.13.3 Ladder card printed circuit board placement top (inside, cooled)

see "Figure 78: ladder card prototype circuit top" while preprod version is under work

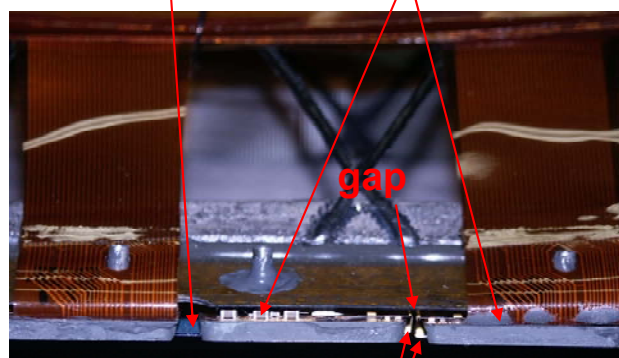
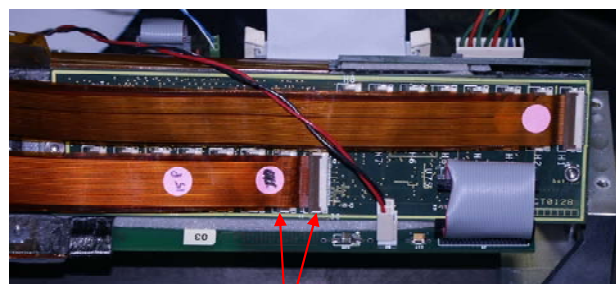
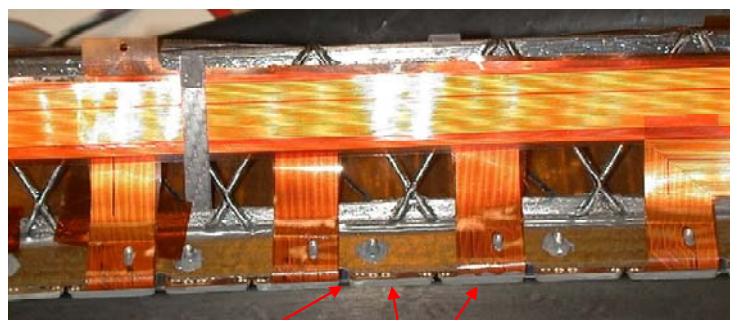
Figure 38: ladder card preproduction circuit top

5.13.4 Ladder card printed circuit board placement bottom (outside, not cooled)

see "Figure 79: ladder card prototype circuit bottom" while preprod version is under work

Figure 39: ladder card preproduction circuit bottom

5.13.5 Flex length adaptation



TAB

Connector

Figure 40: component arrangement on the ladder (old SSD)

$\Delta L = \text{TAB} + \text{module TAB} + \text{gap} + \text{connec}$						
module (mm)	TAB (mm)	gap (mm)	connec (mm)	-->	$\Delta L(\text{mm})$	$\Delta L(\mu\text{m})$
42,0	0,5	0,5	9,3		52,8	52800

$\Delta R = \rho * \Delta L / S$							
material	$\rho(\Omega\mu\text{m})$	$S=w*h*nb$	$w(\mu\text{m})$	$h(\mu\text{m})$	nb	$\Delta L(\mu\text{m})$	$\Delta R (m\Omega)$
aluminium	0,02857143	27500,00	275,00	25	4	52800,00	54,8571429

Table 91: flex cable length difference and related resistive difference

	theory	real	real	real	theory	real	real	real	real
Net Name	L (mm)	S=w*h (μm ²)	w(μm)	h(μm)	R (mΩ)	L (mm)	R (mΩ)	err (%)	Σ (mΩ)
MVDD_P2V<00>	47,0955	7822,50	447	17,5	101,03	47,0955	101,03	0,000%	923,89
MVDD_P2V<01>	37,3901	4025,00	230	17,5	155,89	37,3540	155,74	0,096%	923,74
MVDD_P2V<02>	50,5475	4025,00	230	17,5	210,75	50,5689	210,84	-0,042%	923,98
MVDD_P2V<03>	53,7338	3395,00	194	17,5	265,60	53,5263	264,58	0,386%	922,86
MVDD_P2V<04>	64,1635	3360,00	192	17,5	320,46	63,9965	319,63	0,260%	923,06
MVDD_P2V<05>	67,7107	3027,50	173	17,5	375,32	67,5373	374,36	0,256%	922,93
MVDD_P2V<06>	56,0747	2187,50	125	17,5	430,18	55,9645	429,33	0,196%	923,04
MVDD_P2V<07>	63,2255	2187,50	125	17,5	485,03	63,1072	484,12	0,187%	922,98
MVDD_P2V<08>	70,3763	2187,50	125	17,5	539,89	70,3773	539,90	-0,001%	923,90
MVDD_P2V<09>	77,5271	2187,50	125	17,5	594,75	77,5908	595,24	-0,082%	924,38
MVDD_P2V<10>	84,6779	2187,50	125	17,5	649,60	84,6337	649,26	0,052%	923,55
MVDD_P2V<11>	91,8287	2187,50	125	17,5	704,46	91,8466	704,60	-0,020%	924,03
MVDD_P2V<12>	98,9795	2187,50	125	17,5	759,32	98,9455	759,06	0,034%	923,63
MVDD_P2V<13>	106,1303	2187,50	125	17,5	814,18	106,1079	814,00	0,021%	923,72
MVDD_P2V<14>	113,2811	2187,50	125	17,5	869,03	113,2779	869,01	0,003%	923,87
MVDD_P2V<15>	120,4319	2187,50	125	17,5	923,89	120,3947	923,60	0,031%	923,60
average								0,092%	923,551

	theory	real	real	real	theory	real	real	real	real
Net Name	L (mm)	S=w*h (μm ²)	w(μm)	h(μm)	R (mΩ)	L (mm)	R (mΩ)	err (%)	Σ (mΩ)
MVSS_M2V<00>	57,0178	12250,00	700	17,5	78,11	57,0178	78,11	0,000%	900,97
MVSS_M2V<01>	35,0811	4427,50	253	17,5	132,97	35,0900	133,00	-0,025%	901,00
MVSS_M2V<02>	27,8131	2485,00	142	17,5	187,82	27,8308	187,94	-0,064%	901,09
MVSS_M2V<03>	75,4158	5215,00	298	17,5	242,68	75,3398	242,44	0,101%	900,72
MVSS_M2V<04>	68,5718	3867,50	221	17,5	297,54	68,7275	298,21	-0,227%	901,64
MVSS_M2V<05>	61,7376	2940,00	168	17,5	352,39	61,7734	352,60	-0,058%	901,17
MVSS_M2V<06>	53,0866	2187,50	125	17,5	407,25	53,0810	407,21	0,010%	900,92
MVSS_M2V<07>	60,2374	2187,50	125	17,5	462,11	60,2145	461,93	0,038%	900,79
MVSS_M2V<08>	67,3882	2187,50	125	17,5	516,97	67,3566	516,72	0,047%	900,72
MVSS_M2V<09>	74,5390	2187,50	125	17,5	571,82	74,5186	571,67	0,027%	900,81
MVSS_M2V<10>	81,6898	2187,50	125	17,5	626,68	81,7115	626,85	-0,027%	901,13
MVSS_M2V<11>	88,8406	2187,50	125	17,5	681,54	88,8636	681,71	-0,026%	901,14
MVSS_M2V<12>	95,9914	2187,50	125	17,5	736,39	95,9881	736,37	0,003%	900,94
MVSS_M2V<13>	103,1422	2187,50	125	17,5	791,25	103,1413	791,25	0,001%	900,96
MVSS_M2V<14>	110,2930	2187,50	125	17,5	846,11	110,3124	846,26	-0,018%	901,12
MVSS_M2V<15>	117,4438	2187,50	125	17,5	900,97	117,4068	900,68	0,031%	900,68
average								-0,012%	900,989

material	ρ(Ωμm)	S=w*h*nb	w(μm)	h(μm)	nb	ΔL(μm)	ΔR (mΩ)
copper	0,0168	2187,50	125,00	17,50	1	7150,80	54,8571

Table 92: Flex length difference adaptation for +2V (VDD) and -2V (VSS) on ladder card prototype printed circuit

6 Annexes

6.1 Pin swapping rules for Altera^[8] FPGA

After place and route operation, Quartus II^[11] software generates a .pin file that gives the assignment for all pins of the components.

Power pins and dedicated pins are automatically assigned by Quartus II^[11] software.

When a dedicated functionality is not used, some of the corresponding dedicated pins can be used as user I/O. User I/O pins can be assigned by user. Quartus II^[11] automatically assigns user I/O pins that were not assigned by user.

6.1.1 Power pins

Power pins must not be swapped

6.1.2 Dedicated pins

Dedicated pins are directly connected to internal control blocks. They must not be swapped.

6.1.3 user I/O pins

It is better to swap inputs with inputs and outputs with outputs.

Special care must be taken before swapping a clock signal because there are some "clock dedicated" inputs that can be directly connected to the internal PLLs.

6.1.3.1 LVDS pins

LVDS (Low Voltage Differential Signal) pin pairs must be located in a 2.5V powered "I/O Bank".

LVDS pin pair can be swapped (exchanged) only with another LVDS pin pair. The polarity is important: swap "signalA" with "signalB" and "signalA(n)" with "signalB(n)".

6.1.3.2 3.3-V LVC MOS pins

3.3-V LVC MOS (Low Voltage CMOS) pins must be located in a 3.3V powered "I/O Bank".

6.2 Simulations

6.2.1 VHDL simulations

6.2.1.1 VHDL simulation of a complete ladder side (acquisition of ladder-FPGA + 16 hybrids)

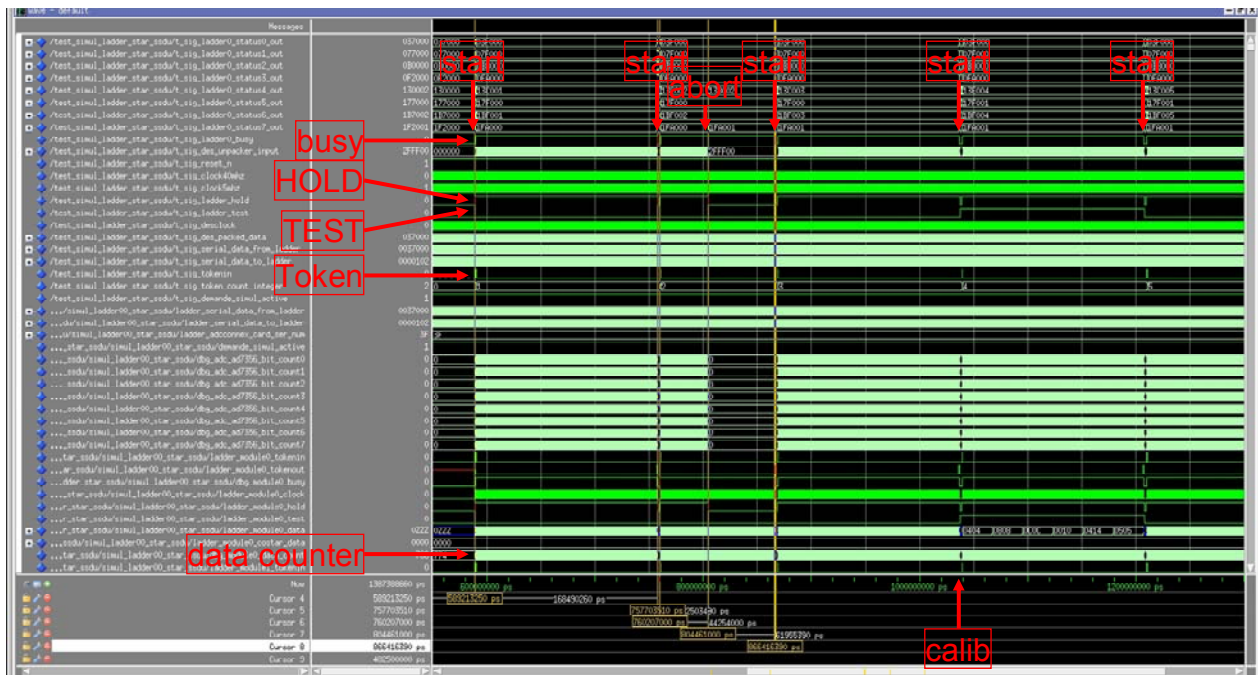


Figure 41: VHDL simulation of a complete ladder side (acquisition of ladder-FPGA + 16 hybrids) using ModelSim-Altera^[12] 6.4a

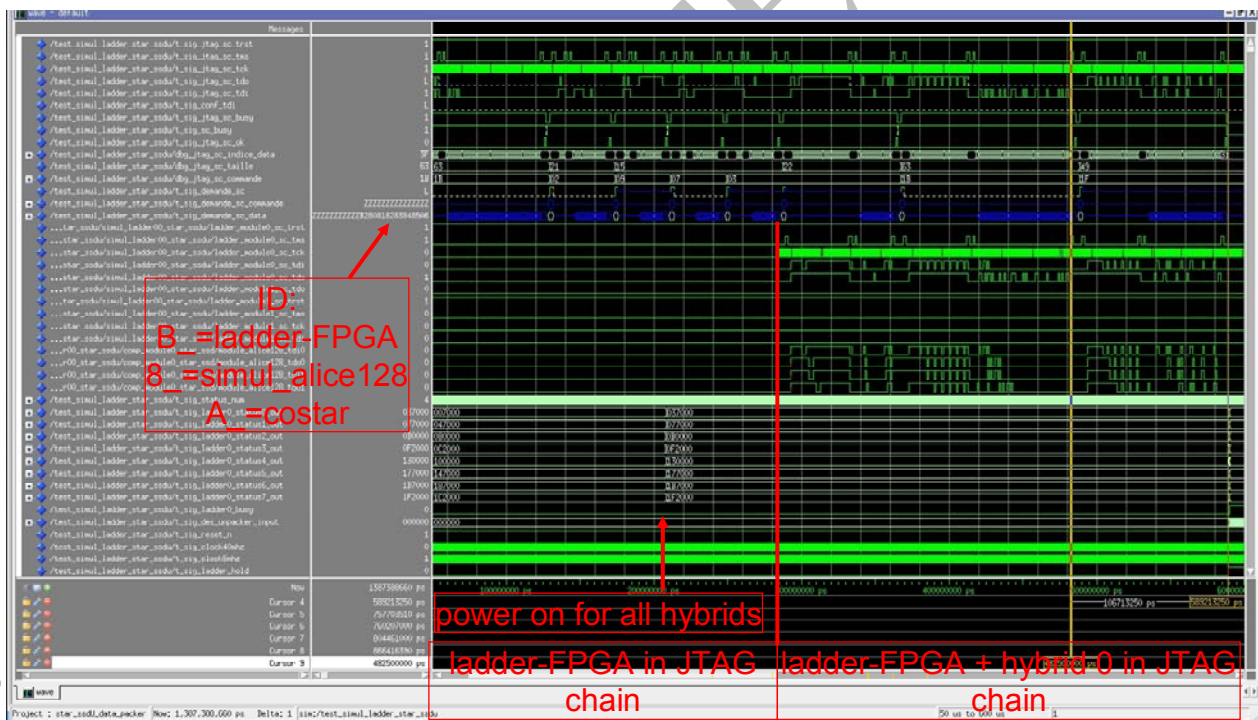


Figure 42: VHDL simulation of a complete ladder side (slow-control of ladder-FPGA + 16 hybrids) using ModelSim-Altera^[12] 6.4a

6.2.1.2 VHDL simulation of a readout card connected to complete ladder sides

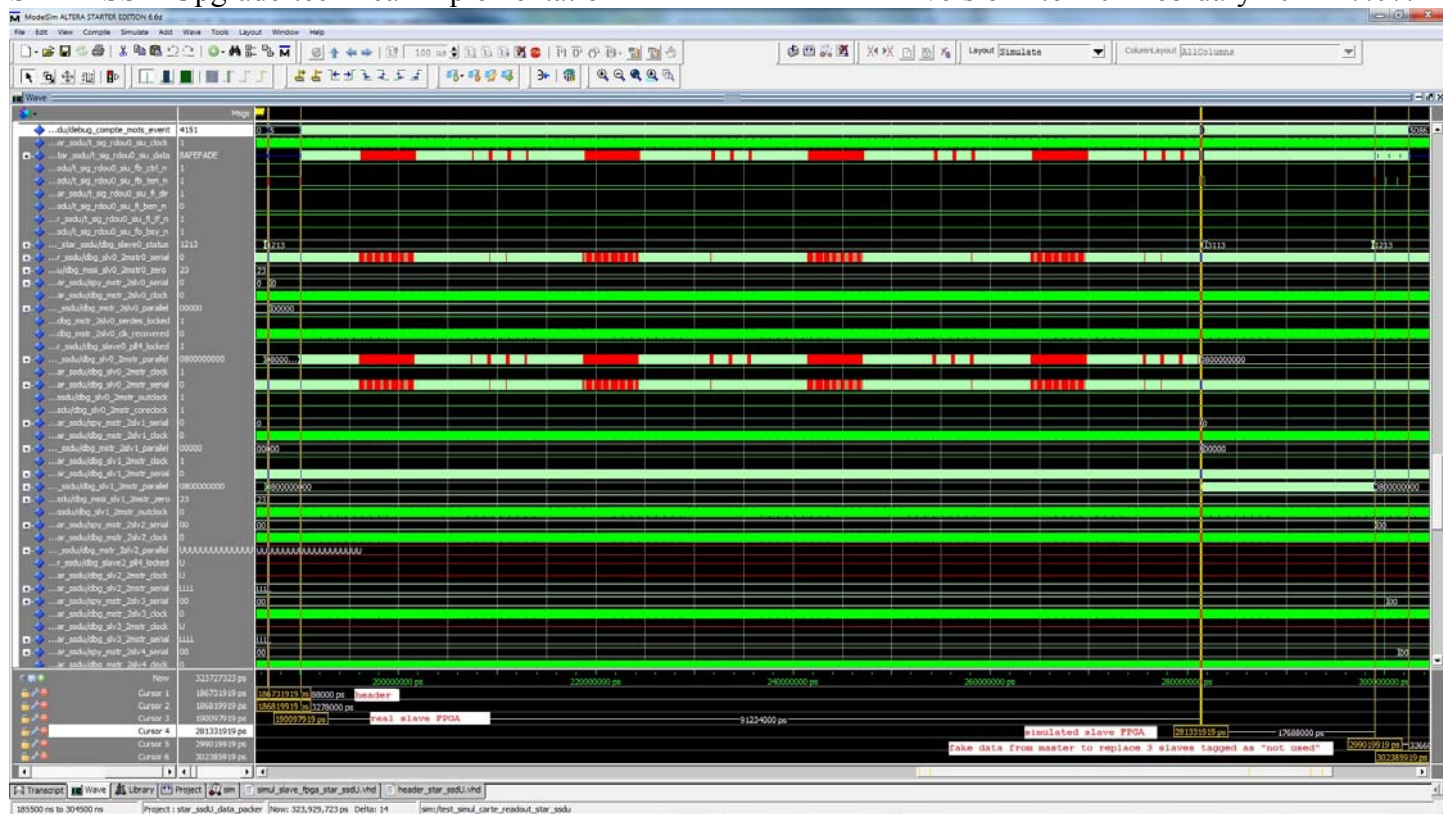


Figure 43: VHDL simulation of a readout card connected to complete ladder sides (L0+L2a) using ModelSim-Altera^[12] starter edition 6.6d

[Ctrl+click here to get image star_ssdU_simul_L2_111018.jpg from \[0\]](#)

6.2.2 Level shifter simulation (from Alice128 to ADC)

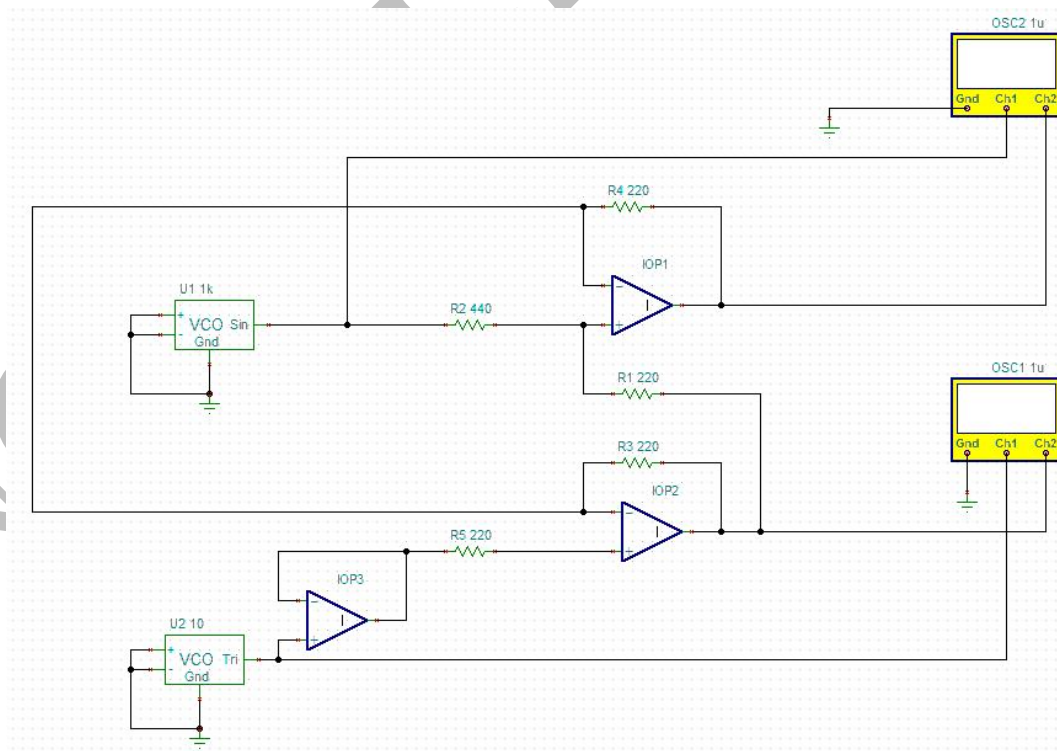


Figure 44: Level shifter (first simulation schematics using Tina)

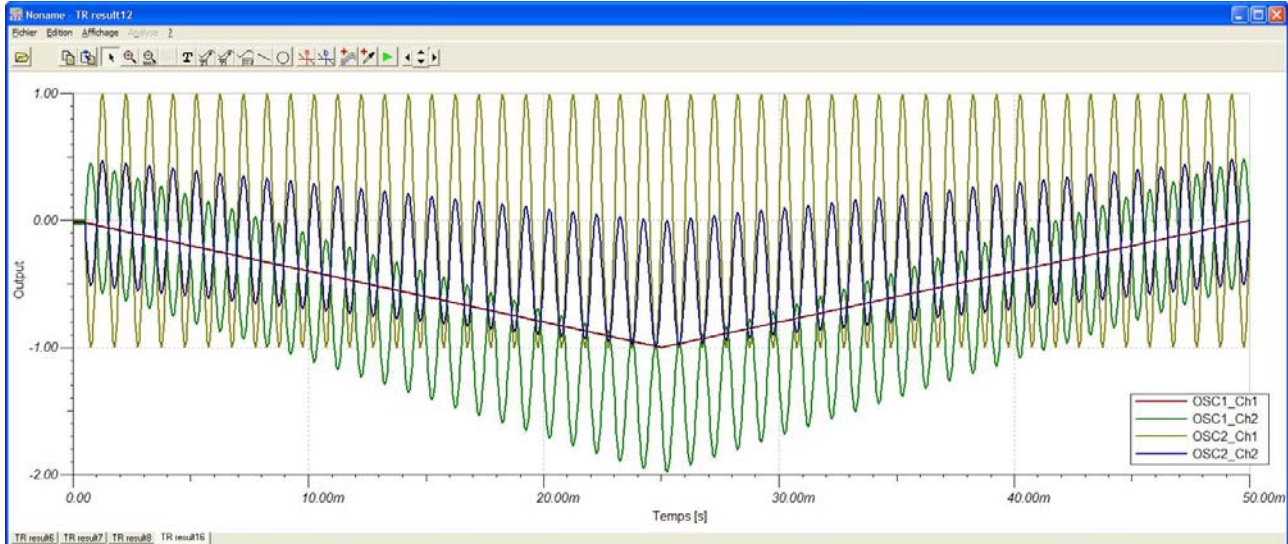
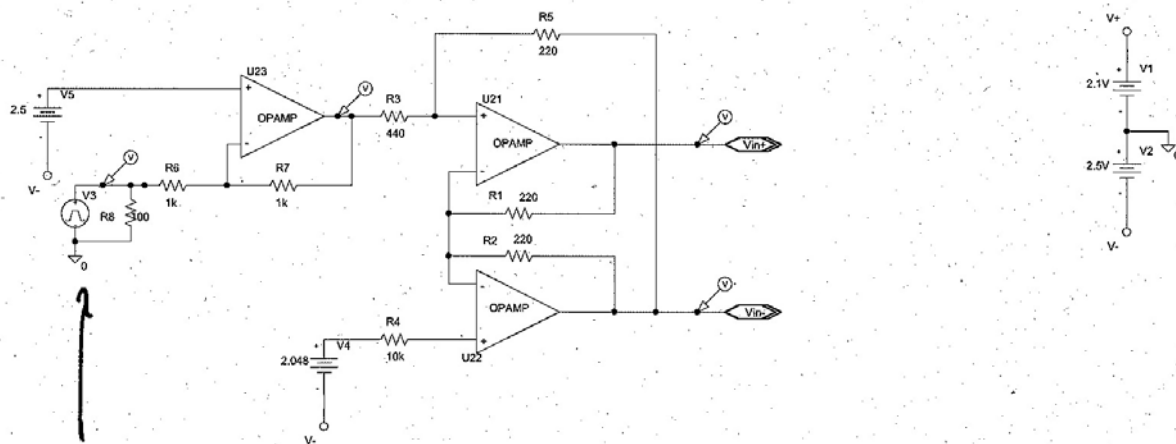


Figure 45: Level shifter (first simulation result using Tina)



signal inversé car sur entrée - de
l'ampli op.

⇒ il faut échanger V_{in+} et V_{in-} .

Figure 46: Level shifter (second simulation schematics using Cadence)

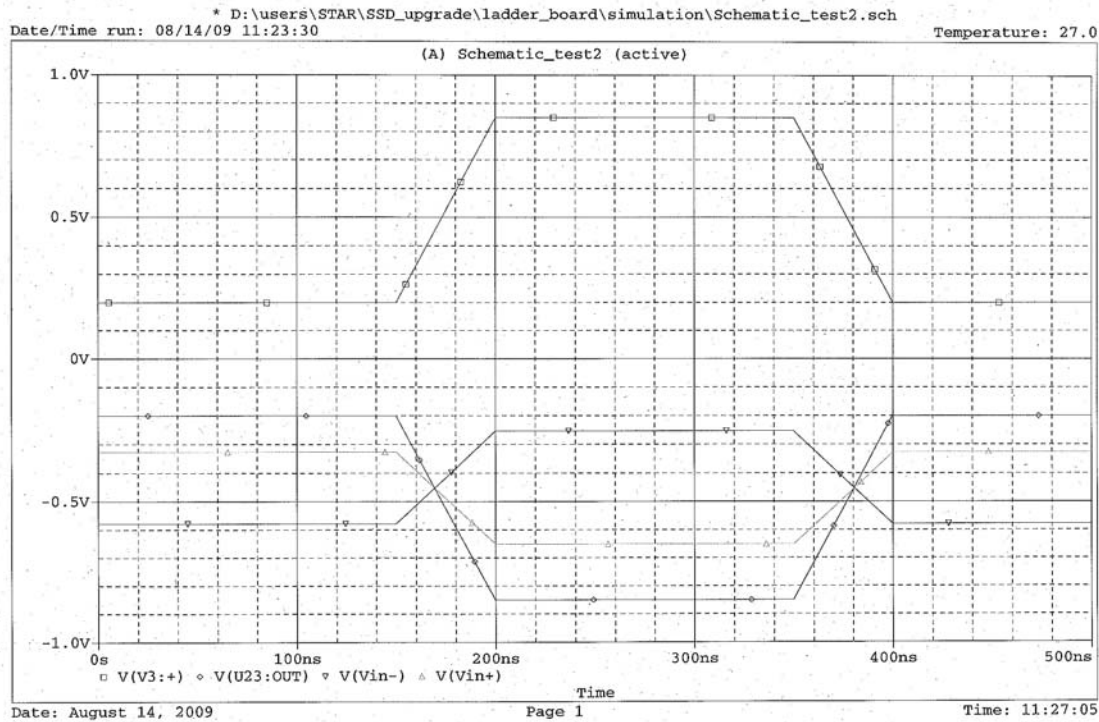


Figure 47: Level shifter (second simulation output using Cadence PSPICE)

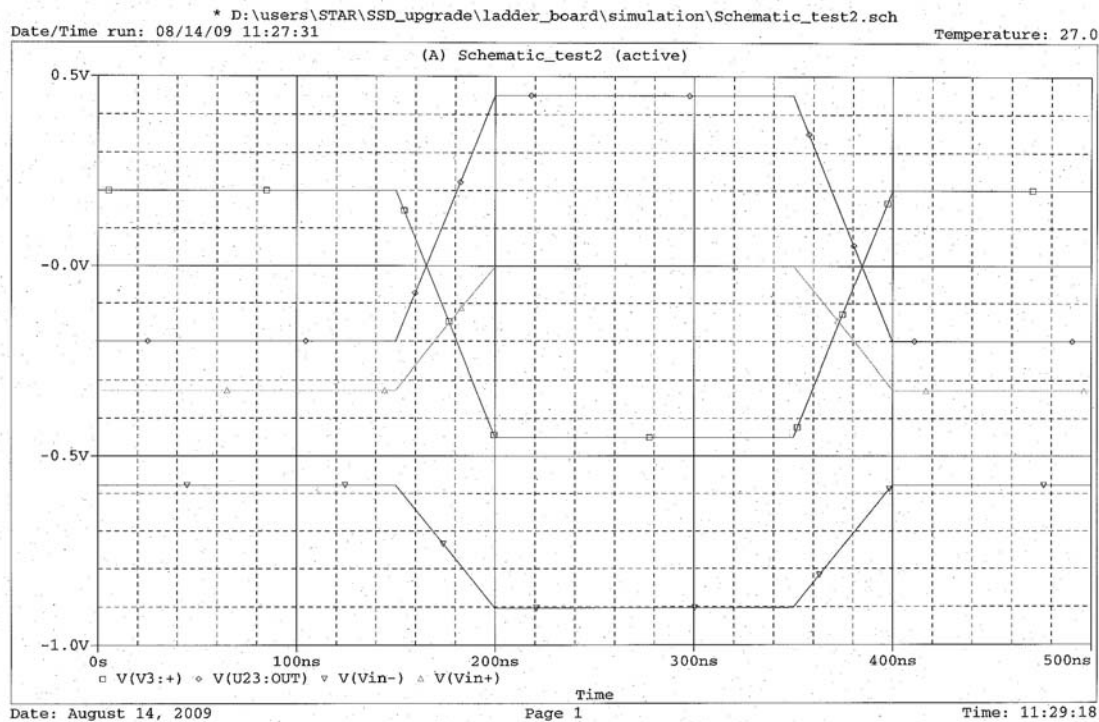


Figure 48: Level shifter (second simulation output using Cadence PSPICE)

6.3 Complete DAQ raw data

name	bits																														Number of words			
	3	3	2	2	2	2	2	2	2	2	2	1	18	1	16	15	14	13	12	1	1	9	8	7	6	5	4	3	2	1	0	mi n	ty p	ma x
Master total words	Master free															Master total words															1	1	1	
Master	Master free												Board type		Vhdl version								fre	Trg	Da	Lad pres		1	1	1				

trailer	
trailer	

	free	Hybrid 14 block										1	24	768	
trailer	C	A	F	E	F			A	D	E	1	1	1		
	free	Hybrid 15 block										1	24	768	
trailer	C	A	F	E	F			A	D	E	1	1	1		
Slave 1 total words	free	Slave free			La d ok	mode	Slave 1 total words					1	1	1	
trailer	F	A	C	E	B			E	E	F	1	1	1		
Ladder 1 total words	free	Slave free				Slave #			Ladder 1 total words				1	1	1
Ladder 1 status	free	Slave free			000	pow 0-1	bsy	1	ok	des clk	Temperature 0	1	1	1	
	free	Slave free			001	pow 2-3	bsy	des lock	des pass	free	Temperature 1	1	1	1	
	free	Slave free			010	pow 4-5	bsy	fib fault	fib abs	fib los	Temperature 2	1	1	1	
	free	Slave free			011	pow 6-7	bsy	usb pres	usb rdyn	free	Temperature 3	1	1	1	
	free	Slave free			100	pow 8-9	bsy	test	hold	free	Nbr hold	1	1	1	
	free	Slave free			101	pow 10-11	bsy	ser(5)	ser(4)	ser(3)	Nbr test	1	1	1	
	free	Slave free			110	pow 12-13	bsy	ser(2)	ser(1)	ser(0)	Nbr token	1	1	1	
	free	Slave free			111	pow 14-15	bsy	ladder #		Nbr abort		1	1	1	
Hybrid total words	free	Hybrid 2 total words			Hybrid 1 total words					Hybrid 0 total words			1	1	1
	free	Hybrid 5 total words			Hybrid 4 total words					Hybrid 3 total words			1	1	1
	free	Hybrid 8 total words			Hybrid 7 total words					Hybrid 6 total words			1	1	1
	free	Hybrid 11 total words			Hybrid 10 total words					Hybrid 9 total words			1	1	1
	free	Hybrid 14 total words			Hybrid 13 total words					Hybrid 12 total words			1	1	1
	free	Slave free			Slave free					Hybrid 15 total words			1	1	1
trailer	B	E	E	F	F			A	C	E	1	1	1		
	free	Hybrid 0 block										1	24	768	
trailer	C	A	F	E	F			A	D	E	1	1	1		
	free	Hybrid 1 block										1	24	768	
trailer	C	A	F	E	F			A	D	E	1	1	1		
	free	Hybrid 2 block										1	24	768	
trailer	C	A	F	E	F			A	D	E	1	1	1		
	free	Hybrid 3 block										1	24	768	
trailer	C	A	F	E	F			A	D	E	1	1	1		
	free	Hybrid 4 block										1	24	768	
trailer	C	A	F	E	F			A	D	E	1	1	1		
	free	Hybrid 5 block										1	24	768	
trailer	C	A	F	E	F			A	D	E	1	1	1		
	free	Hybrid 6 block										1	24	768	
trailer	C	A	F	E	F			A	D	E	1	1	1		
	free	Hybrid 7 block										1	24	768	
trailer	C	A	F	E	F			A	D	E	1	1	1		
	free	Hybrid 8 block										1	24	768	
trailer	C	A	F	E	F			A	D	E	1	1	1		
	free	Hybrid 9 block										1	24	768	
trailer	C	A	F	E	F			A	D	E	1	1	1		
	free	Hybrid 10 block										1	24	768	
trailer	C	A	F	E	F			A	D	E	1	1	1		
	free	Hybrid 11 block										1	24	768	
trailer	C	A	F	E	F			A	D	E	1	1	1		
	free	Hybrid 12 block										1	24	768	
trailer	C	A	F	E	F			A	D	E	1	1	1		
	free	Hybrid 13 block										1	24	768	
trailer	C	A	F	E	F			A	D	E	1	1	1		
	free	Hybrid 14 block										1	24	768	
trailer	C	A	F	E	F			A	D	E	1	1	1		
	free	Hybrid 15 block										1	24	768	
trailer	C	A	F	E	F			A	D	E	1	1	1		
Slave 2 total words	free	Slave free			La d ok	mode	Slave 2 total words					1	1	1	
trailer	F	A	C	E	B			E	E	F	1	1	1		
Ladder 2 total	free	Slave free				Slave #			Ladder 2 total words				1	1	1

words															
Ladder 2 status	free	Slave free		000	pow 0-1	bsy	1	ok	des clk	Temperature 0			1	1	1
	free	Slave free		001	pow 2-3	bsy	des lock	des pass	free	Temperature 1			1	1	1
	free	Slave free		010	pow 4-5	bsy	fib fault	fib abs	fib los	Temperature 2			1	1	1
	free	Slave free		011	pow 6-7	bsy	usb pres	usb rdyn	free	Temperature 3			1	1	1
	free	Slave free		100	pow 8-9	bsy	test	hold	free	Nbr hold			1	1	1
	free	Slave free		101	pow 10-11	bsy	ser(5)	ser(4)	ser(3)	Nbr test			1	1	1
	free	Slave free		110	pow 12-13	bsy	ser(2)	ser(1)	ser(0)	Nbr token			1	1	1
	free	Slave free		111	pow 14-15	bsy	ladder #		Nbr abort			1	1	1	
Hybrid total words	free	Hybrid 2 total words		Hybrid 1 total words					Hybrid 0 total words			1	1	1	
	free	Hybrid 5 total words		Hybrid 4 total words					Hybrid 3 total words			1	1	1	
	free	Hybrid 8 total words		Hybrid 7 total words					Hybrid 6 total words			1	1	1	
	free	Hybrid 11 total words		Hybrid 10 total words					Hybrid 9 total words			1	1	1	
	free	Hybrid 14 total words		Hybrid 13 total words					Hybrid 12 total words			1	1	1	
	free	Slave free		Slave free					Hybrid 15 total words			1	1	1	
trailer	B	E	E	F	F			A	C	E	1	1	1		
	free	Hybrid 0 block											1	24	768
trailer	C	A	F	E	F			A	D	E	1	1	1		
	free	Hybrid 1 block											1	24	768
trailer	C	A	F	E	F			A	D	E	1	1	1		
	free	Hybrid 2 block											1	24	768
trailer	C	A	F	E	F			A	D	E	1	1	1		
	free	Hybrid 3 block											1	24	768
trailer	C	A	F	E	F			A	D	E	1	1	1		
	free	Hybrid 4 block											1	24	768
trailer	C	A	F	E	F			A	D	E	1	1	1		
	free	Hybrid 5 block											1	24	768
trailer	C	A	F	E	F			A	D	E	1	1	1		
	free	Hybrid 6 block											1	24	768
trailer	C	A	F	E	F			A	D	E	1	1	1		
	free	Hybrid 7 block											1	24	768
trailer	C	A	F	E	F			A	D	E	1	1	1		
	free	Hybrid 8 block											1	24	768
trailer	C	A	F	E	F			A	D	E	1	1	1		
	free	Hybrid 9 block											1	24	768
trailer	C	A	F	E	F			A	D	E	1	1	1		
	free	Hybrid 10 block											1	24	768
trailer	C	A	F	E	F			A	D	E	1	1	1		
	free	Hybrid 11 block											1	24	768
trailer	C	A	F	E	F			A	D	E	1	1	1		
	free	Hybrid 12 block											1	24	768
trailer	C	A	F	E	F			A	D	E	1	1	1		
	free	Hybrid 13 block											1	24	768
trailer	C	A	F	E	F			A	D	E	1	1	1		
	free	Hybrid 14 block											1	24	768
trailer	C	A	F	E	F			A	D	E	1	1	1		
	free	Hybrid 15 block											1	24	768
trailer	C	A	F	E	F			A	D	E	1	1	1		
Slave 3 total words	free	Slave free		La d ok	mode	Slave 3 total words							1	1	1
trailer	F	A	C	E	B			E	E	F	1	1	1		
Ladder 3 total words	free	Slave free			Slave #			Ladder 3 total words					1	1	1
Ladder 3 status	free	Slave free		000	pow 0-1	bsy	1	ok	des clk	Temperature 0			1	1	1
	free	Slave free		001	pow 2-3	bsy	des lock	des pass	free	Temperature 1			1	1	1
	free	Slave free		010	pow 4-5	bsy	fib fault	fib abs	fib los	Temperature 2			1	1	1
	free	Slave free		011	pow 6-7	bsy	usb pres	usb rdyn	free	Temperature 3			1	1	1
	free	Slave free		100	pow 8-9	bsy	test	hold	free	Nbr hold			1	1	1

	free	Slave free			101	pow 10- 11	bsy	ser(5)	ser(4)	ser(3)	Nbr test			1	1	1		
	free	Slave free			110	pow 12- 13	bsy	ser(2)	ser(1)	ser(0)	Nbr token			1	1	1		
	free	Slave free			111	pow 14- 15	bsy	ladder #			Nbr abort			1	1	1		
Hybrid total words	free	Hybrid 2 total words			Hybrid 1 total words						Hybrid 0 total words			1	1	1		
	free	Hybrid 5 total words			Hybrid 4 total words						Hybrid 3 total words			1	1	1		
	free	Hybrid 8 total words			Hybrid 7 total words						Hybrid 6 total words			1	1	1		
	free	Hybrid 11 total words			Hybrid 10 total words						Hybrid 9 total words			1	1	1		
	free	Hybrid 14 total words			Hybrid 13 total words						Hybrid 12 total words			1	1	1		
	free	Slave free			Slave free						Hybrid 15 total words			1	1	1		
trailer	B	E	E	F	F			A	C		E	1	1	1				
	free	Hybrid 0 block														1	24	768
trailer	C	A	F	E	F			A	D		E	1	1	1				
	free	Hybrid 1 block														1	24	768
trailer	C	A	F	E	F			A	D		E	1	1	1				
	free	Hybrid 2 block														1	24	768
trailer	C	A	F	E	F			A	D		E	1	1	1				
	free	Hybrid 3 block														1	24	768
trailer	C	A	F	E	F			A	D		E	1	1	1				
	free	Hybrid 4 block														1	24	768
trailer	C	A	F	E	F			A	D		E	1	1	1				
	free	Hybrid 5 block														1	24	768
trailer	C	A	F	E	F			A	D		E	1	1	1				
	free	Hybrid 6 block														1	24	768
trailer	C	A	F	E	F			A	D		E	1	1	1				
	free	Hybrid 7 block														1	24	768
trailer	C	A	F	E	F			A	D		E	1	1	1				
	free	Hybrid 8 block														1	24	768
trailer	C	A	F	E	F			A	D		E	1	1	1				
	free	Hybrid 9 block														1	24	768
trailer	C	A	F	E	F			A	D		E	1	1	1				
	free	Hybrid 10 block														1	24	768
trailer	C	A	F	E	F			A	D		E	1	1	1				
	free	Hybrid 11 block														1	24	768
trailer	C	A	F	E	F			A	D		E	1	1	1				
	free	Hybrid 12 block														1	24	768
trailer	C	A	F	E	F			A	D		E	1	1	1				
	free	Hybrid 13 block														1	24	768
trailer	C	A	F	E	F			A	D		E	1	1	1				
	free	Hybrid 14 block														1	24	768
trailer	C	A	F	E	F			A	D		E	1	1	1				
	free	Hybrid 15 block														1	24	768
trailer	C	A	F	E	F			A	D		E	1	1	1				
Slave 4 total words	free	Slave free			La d ok	mode	Slave 4 total words						1	1	1			
trailer	F	A	C	E	B			E	E		F	1	1	1				
Ladder 4 total words	free	Slave free			Slave #			Ladder 4 total words						1	1	1		
Ladder 4 status	free	Slave free			000	pow 0-1	bsy	1	ok	des clk	Temperature 0			1	1	1		
	free	Slave free			001	pow 2-3	bsy	des lock	des pass	free	Temperature 1			1	1	1		
	free	Slave free			010	pow 4-5	bsy	fib fault	fib abs	fib los	Temperature 2			1	1	1		
	free	Slave free			011	pow 6-7	bsy	usb pres	usb rdyn	free	Temperature 3			1	1	1		
	free	Slave free			100	pow 8-9	bsy	test	hold	free	Nbr hold			1	1	1		
	free	Slave free			101	pow 10- 11	bsy	ser(5)	ser(4)	ser(3)	Nbr test			1	1	1		
	free	Slave free			110	pow 12- 13	bsy	ser(2)	ser(1)	ser(0)	Nbr token			1	1	1		
	free	Slave free			111	pow 14- 15	bsy	ladder #			Nbr abort			1	1	1		
Hybrid total	free	Hybrid 2 total words			Hybrid 1 total words						Hybrid 0 total words			1	1	1		
	free	Hybrid 5 total words			Hybrid 4 total words						Hybrid 3 total words			1	1	1		

Table 93: *complete raw data*

6.4 SSD before Upgrade

6.4.1 SSD power supply management before Upgrade





Figure 50: Power supply and sense pigtails (AWG24) and connectors for ADC card

6.4.2 SSD acquisition and slow-control before Upgrade

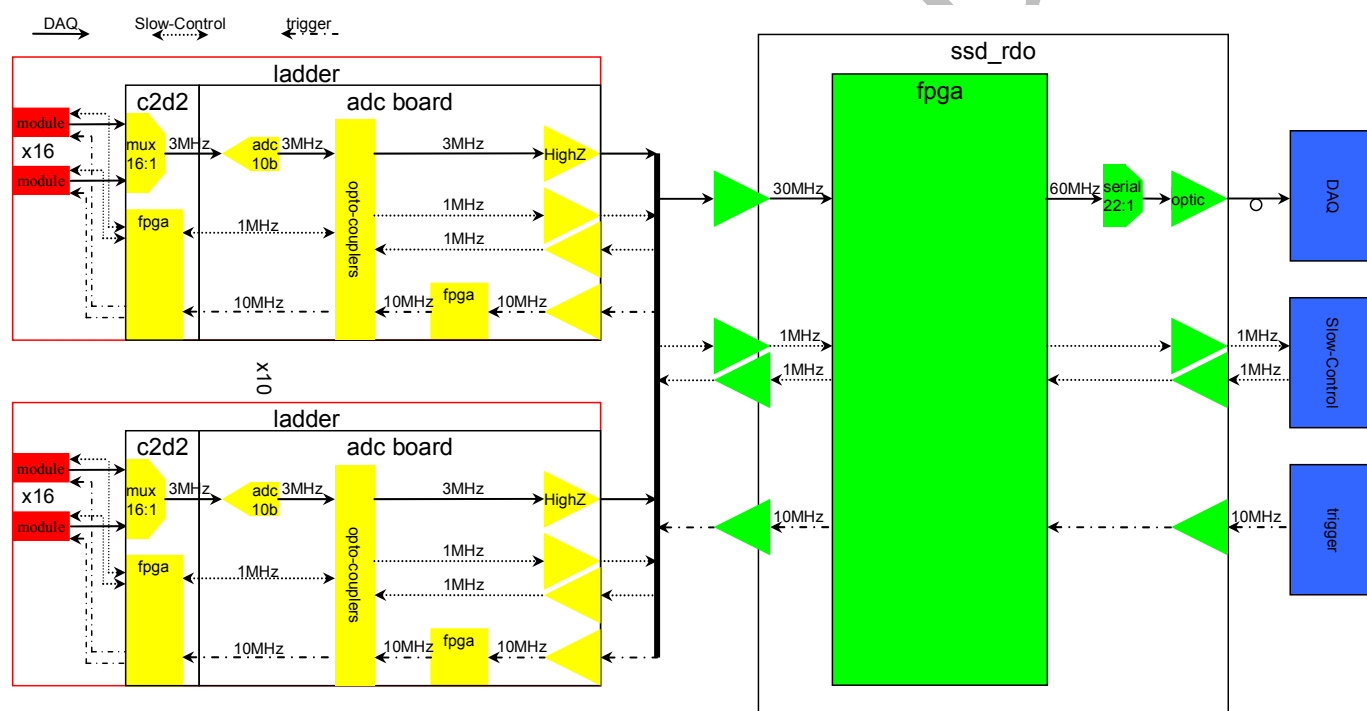
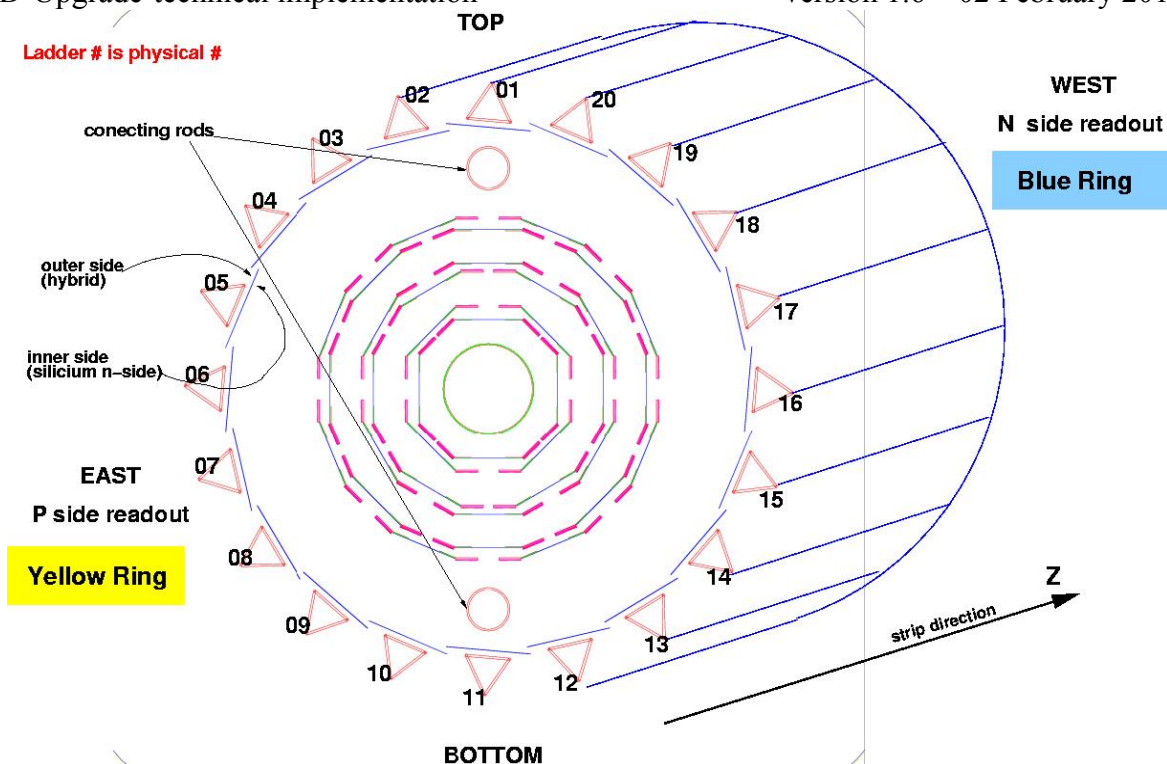
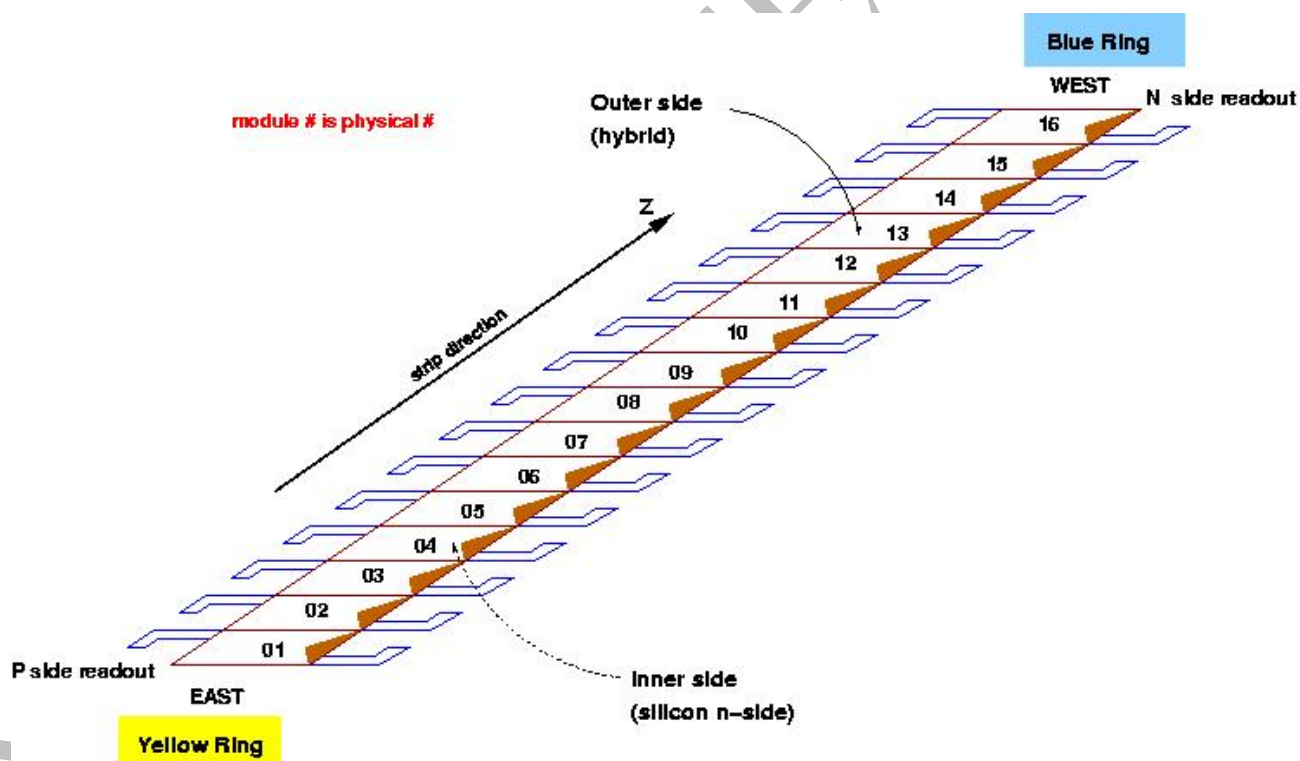
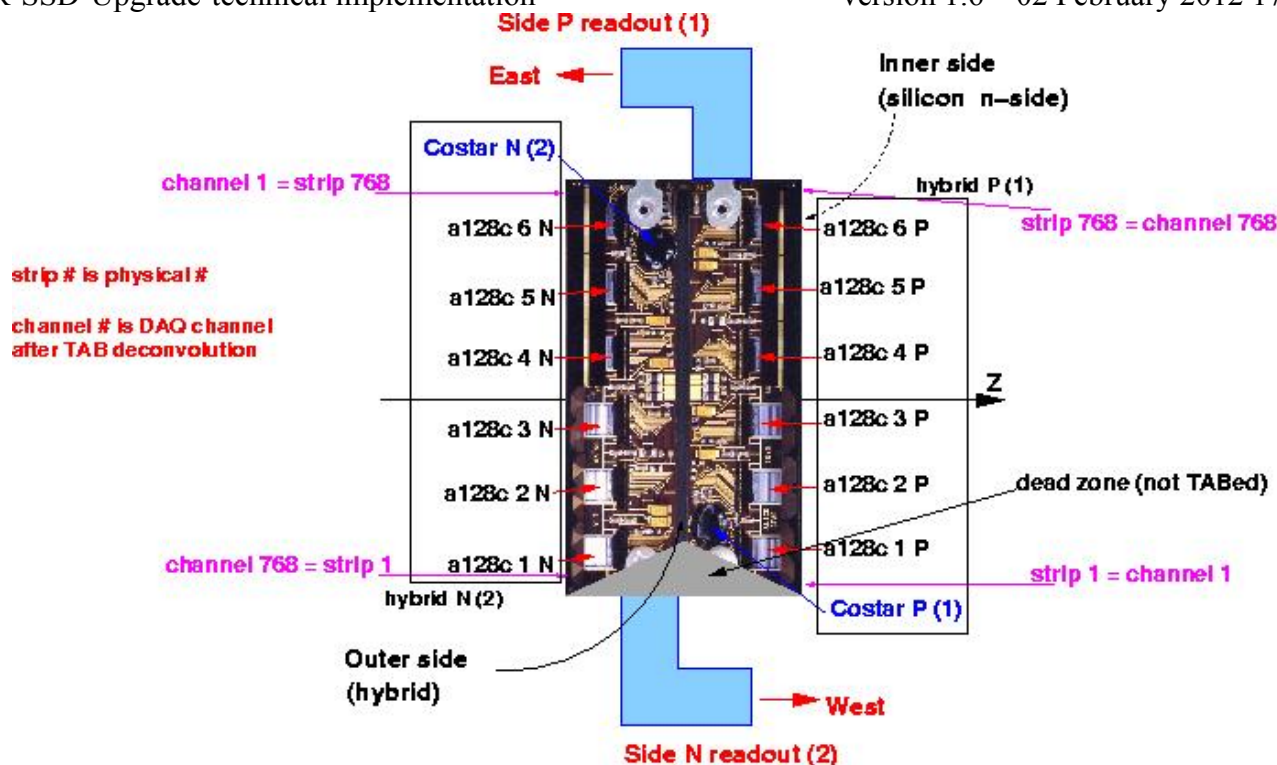
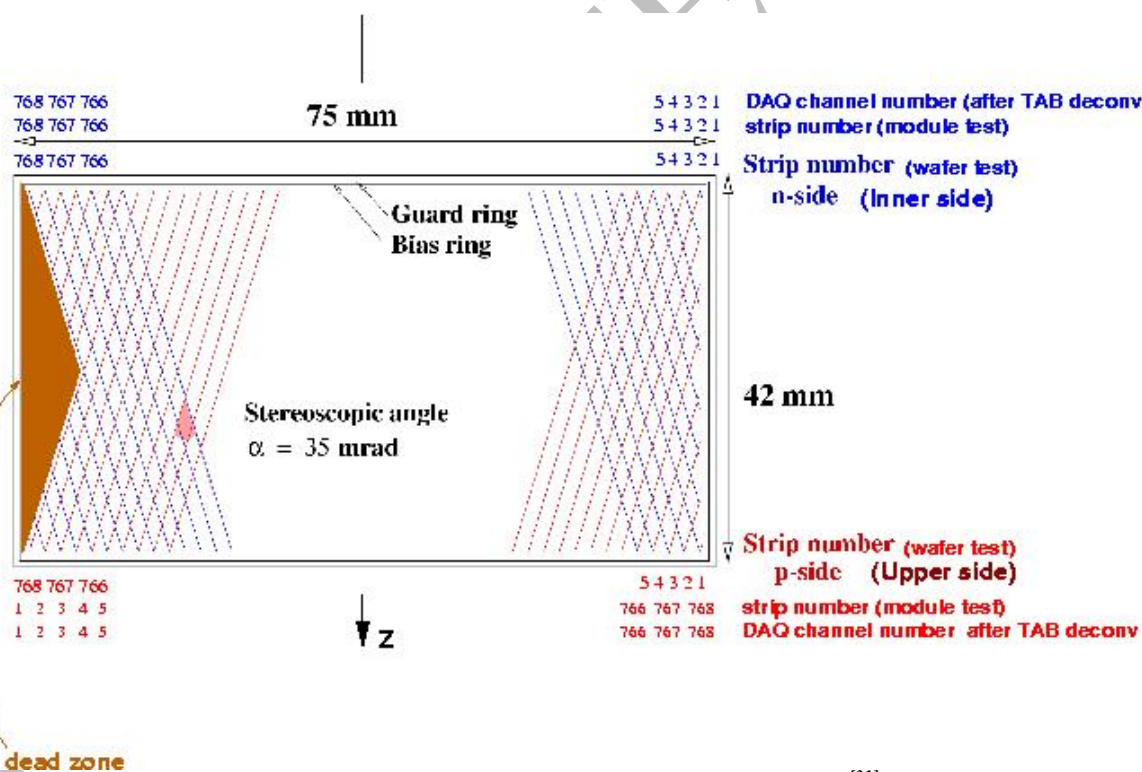


Figure 51: SSD acquisition and slow-control diagram before Upgrade

6.4.3 SSD numbering before Upgrade

Figure 52: SSD numbering at SSD level before Upgrade^[31]Figure 53: SSD numbering at ladder level before Upgrade^[31]

Figure 54: SSD numbering at module level before Upgrade^[31]Figure 55: SSD numbering at Wafer level before Upgrade^[31]

6.4.4 SSD ADC before Upgrade

AD9200

REV. E

DIFFERENTIAL INPUT OPERATION
The AD9200 will accept differential input signals. This function may be used by shorting REFTS and REFBS and driving them as one leg of the differential signal (the top leg is driven into AIN). In the configuration below, the AD9200 is accepting a 1 V p-p signal. See Figure 29.

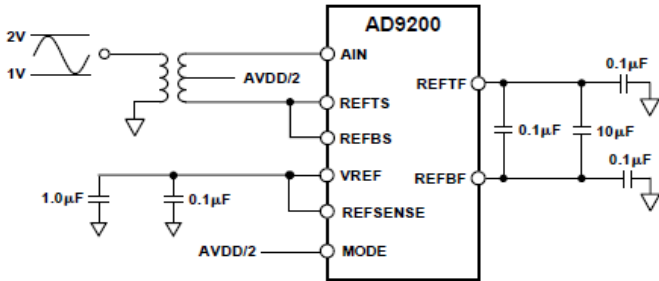


Figure 29. Differential Input

Figure 56: SSD ADC mode of operation before Upgrade^[32]

An input span of 1Vpp for 10 bit resolution led to:

• $LSB = \frac{1V}{1024} \approx 1mV$

Équation 23: LSB value in the ADC before upgrade

currents needed (in μA) by each sub-part					
sub-part	Qty	-2V	Agnd	+2V	usage
Alice128 preamplifier (x128)	128	12	91	103	sampling and acquisition
Alice128 shaper (X128)	128	17	15	32	sampling and acquisition
Alice128 intermediate buffer (x4)	4	1850	0	1850	acquisition
Alice128 output buffer (x1)	1	8450	0	8450	acquisition

currents needed (in mA) by each chip					
chip	Qty	-2V	Agnd	+2V	state
Alice128	6	3,712	13,568	17,28	sampling
Alice128	note1	19,562	13,568	33,13	acquisition
costar	1	11	0	11	sampling and acquisition

currents needed (in mA) by each hybrid					
hybrid	Qty	-2V	Agnd	+2V	state
hybrid	16	33,272	81,408	114,68	sampling
hybrid	note2	49,122	81,408	130,53	acquisition

currents needed (in mA) by the hybrids on each ladder end					
SSD version	Qty	-2V	Agnd	+2V	state
old SSD	1	532,352	1302,528	1834,88	sampling
old SSD	note3	548,202	1302,528	1850,73	acquisition
SSD Upgrade	1	532,352	1302,528	1834,88	sampling
SSD Upgrade	1	785,952	1302,528	2088,48	acquisition

power consumed (in mW) by the hybrids on each ladder end				state
SSD version	Qty	power		
old SSD		4734,464		sampling
old SSD		4797,864		acquisition
SSD Upgrade		4734,464		sampling
SSD Upgrade		5748,864		acquisition

note1: For one hybrid, only one Alice128 chip should be in acquisition state

note2: (1 acquisition + 15 sampling) in old SSD and (16 acquisition) in Upgraded SSD

note3: in the old SSD, after an abort, there could be more than one hybrid in acquisition state (token out of the first hybrid but not out of the ladder)

Table 94: currents needed by the Front-End Electronics (FEE) ^{[33][34]}

6.4.6 Alice128C^{[4][5]} timing

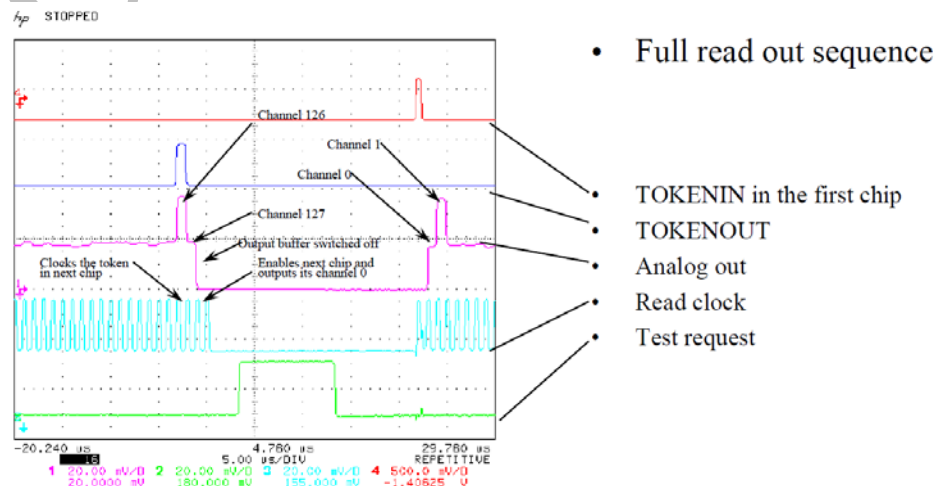


Figure 57: Alice128C timing^[5]

6.4.7 Alice128^{[4][5]} internal shaper output signal

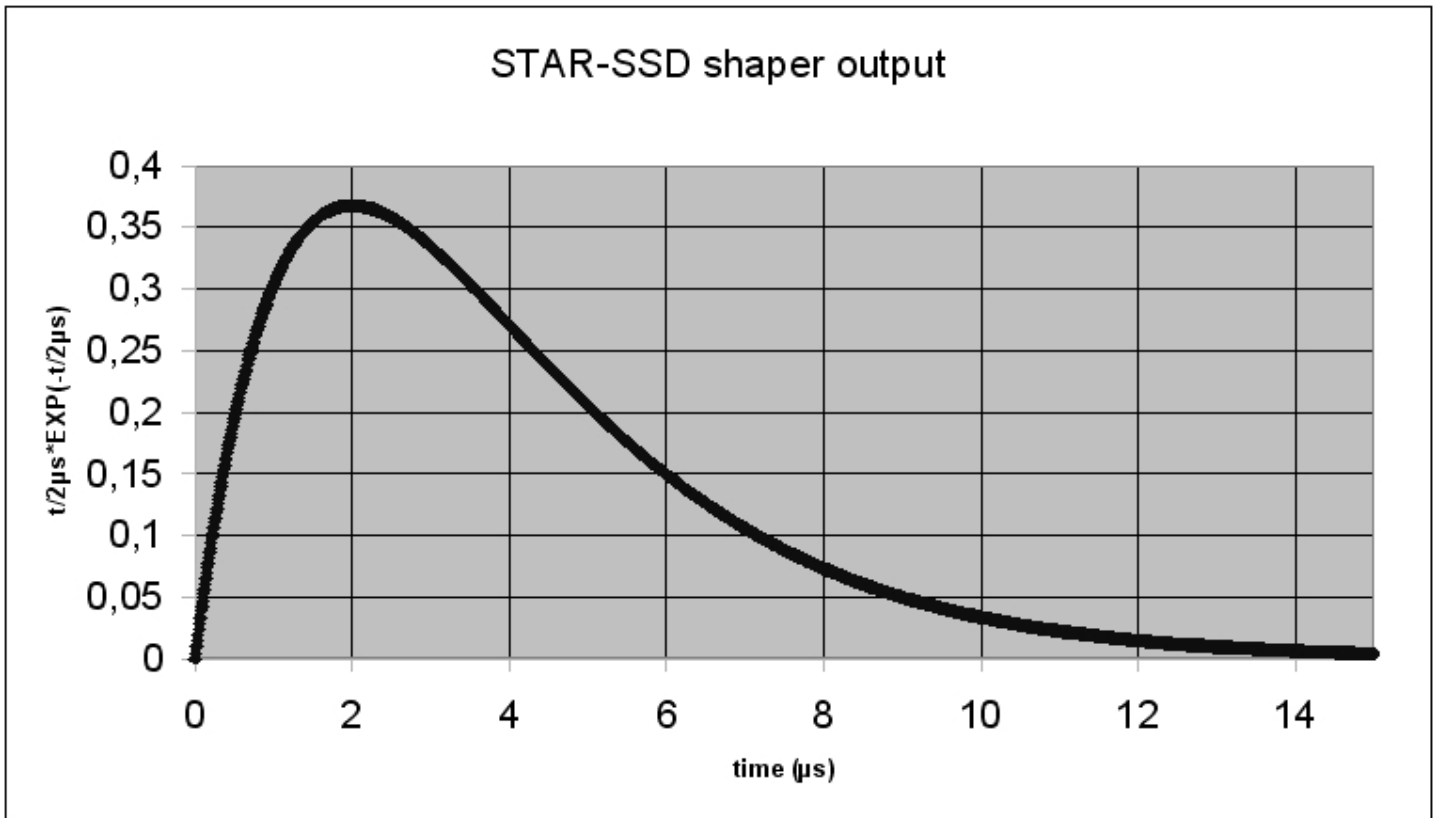


Figure 58: Alice128 internal shaper output signal (calculated by Stéphane Bouvier)

25ns jitter on the HOLD signal sample time would lead to less than 0.008% error on the sampled value.

6.4.8 Test of the readout frequency with old readout card

6.4.8.1 Overview

The purpose of this test is to find out the maximum speed that can be used to readout the hybrids, without degrading the signal. To increase the readout speed, we decided to decrease the number of ladders that the readout card reads. To make this test possible, the VHDL code inside of the FPGA in the old readout card has been modified. This special code is referenced as version 6d: `fe_nb_pre_clk(2 downto 0)` in the slow-control register “adjust” (address 0x12)^[35] that was never used has been renamed `sc_fe_nb_echelle`. It is now used to tell how many ladder(s) the readout card must not read, out of the ten ladders that are connected to it.

<i>sc_fe_nb_echelle</i>	<i>Number of ladders that are readout</i>	<i>Readout frequency for the hybrids (MHz)</i>	<i>Readout cycle (ladder #)</i>	<i>Daq cycle (raw data)</i>
0	10	3.000	1-3-5-7-9-10-8-6-4-2	6 data + 14 idle
1	9	3.333	1-3-5-7-9-8-6-4-2	6 data + 12 idle
2	8	3.750	1-3-5-7-8-6-4-2	6 data + 10 idle
3	7	4.286	1-3-5-7-6-4-2	6 data + 8 idle
4	6	5.000	1-3-5-6-4-2	6 data + 6 idle
5	5	6.000	1-3-5-4-2	6 data + 4 idle
6	4	7.500	1-3-4-2	6 data + 2 idle
7	3	10.000	1-3-2	6 data

Table 95: number of ladders read out and equivalent readout speed for the hybrids

6.4.8.2 Simulation

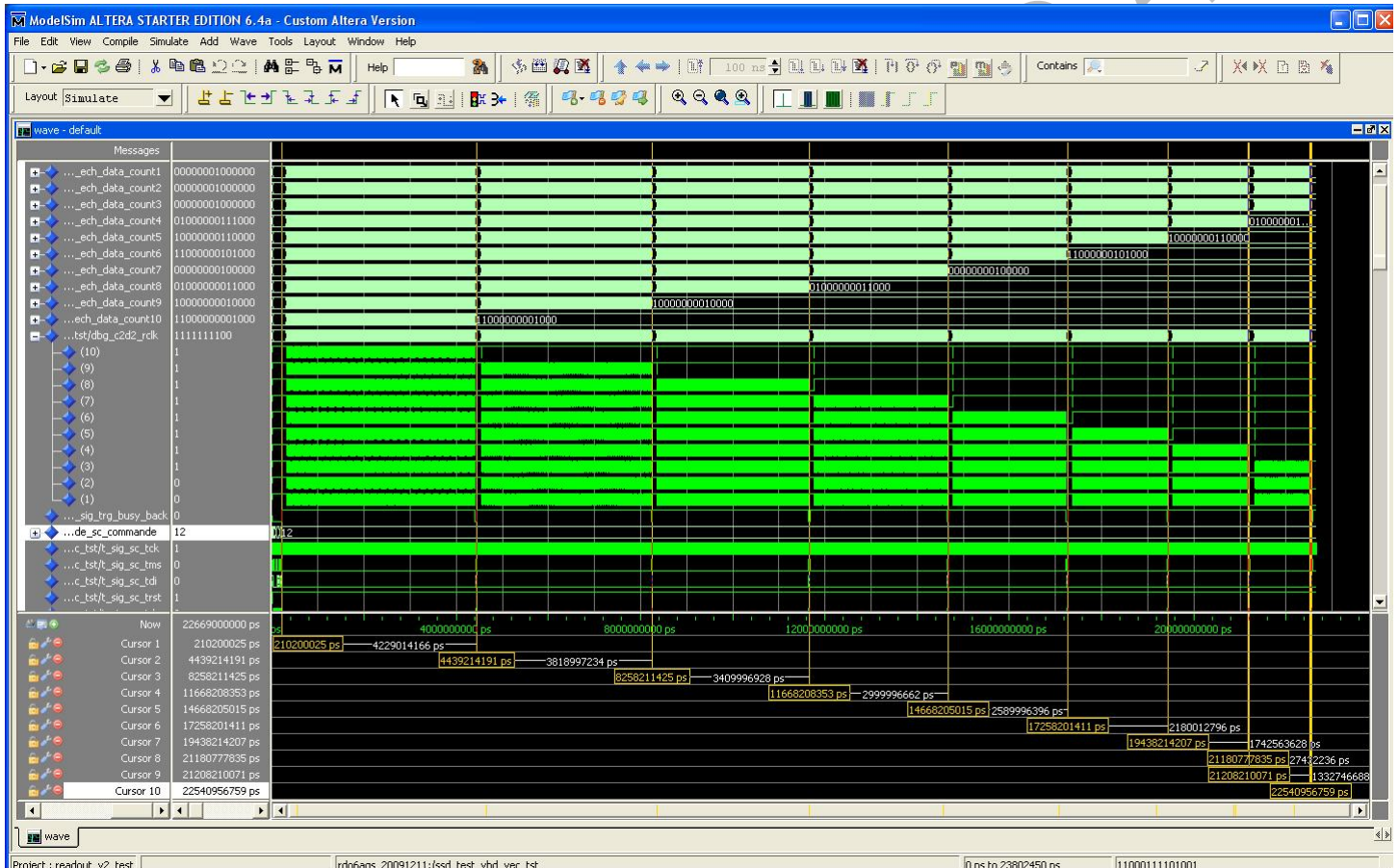


Figure 59: VHDL simulation (1xrdo+10xadc+10xc2d2+10x16xhybrids) of a hybrid readout frequency test using ModelSim-Altera^[12] 6.4a

[Ctrl+click here to get image star_ssd_simul_ladder_rdo6aqs.jpg from \[0\]](#)

The time indicated between these consecutive cursors (c8-c9) corresponds to the slow-control time to change the quantity of rejected (not acquired) ladders.

The time indicated between these consecutive cursors (c7-c8, and c9-c10) correspond to the full acquisition time (SSD busy).

The time indicated between these consecutive cursors (c1-c2, c2-c3, c3-c4, c4-c5, c5-c6, and c6-c7) correspond to the full acquisition time plus the slow-control time to change the quantity of rejected (not acquired) ladders.

6.4.8.3 Tests

Tests were done at Nantes in January 2010^[36].

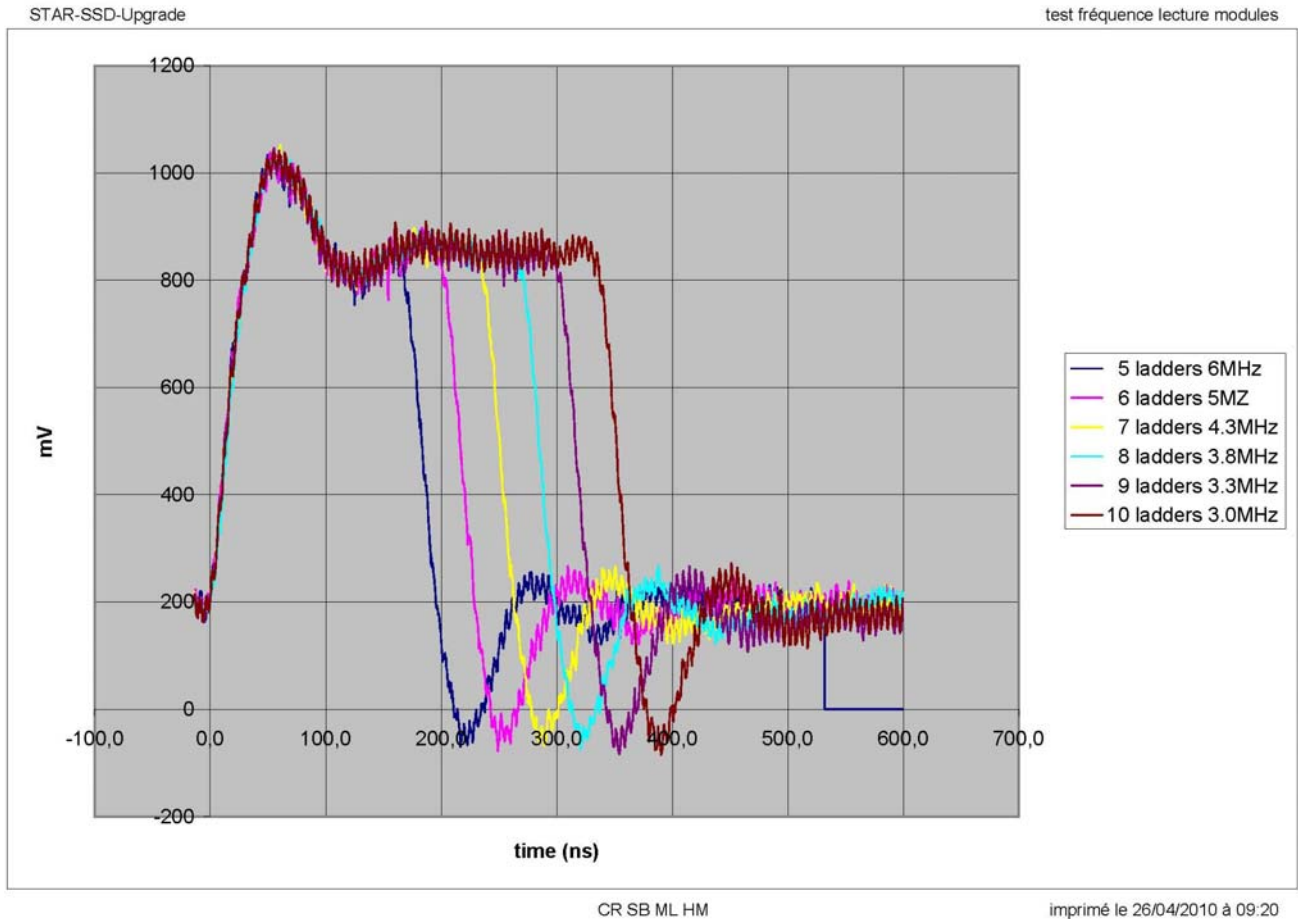


Figure 60: Superposition of test signals, for different readout speeds, recorded with an oscilloscope.^[36]

6.5 Simulation of JTAG master ACT8990 emulator (slave-FPGA)^[37]

Test results for the shifting operations with internal flow control

M.J.LeVine 15 January 2011

6.5.1 Test 1: Write 2 words to write buffer, separated by 25µs (independent writes)

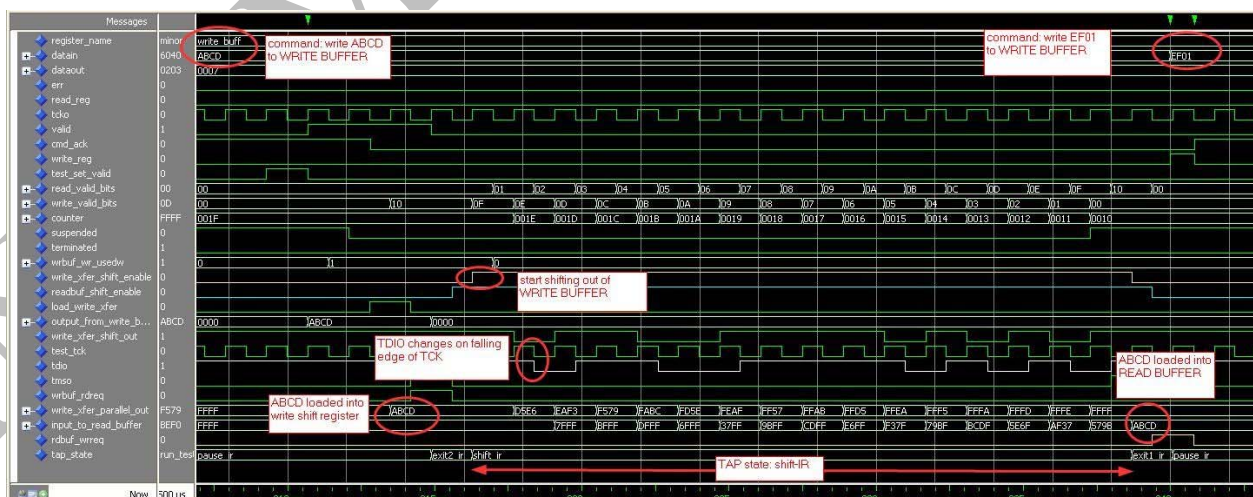


Figure 61: First word = "x"ABCD". Value is loaded into write shift register, which is the source for the outgoing TDIO. Note that changes occur on falling edge of TCK, as required. TDIO is looped back to TDO (input to read shift register). At the end of the process, "x"ABCD" is loaded into READ BUFFER, where it can be read by the host.

[Ctrl+click here to get image from \[0\]](#)

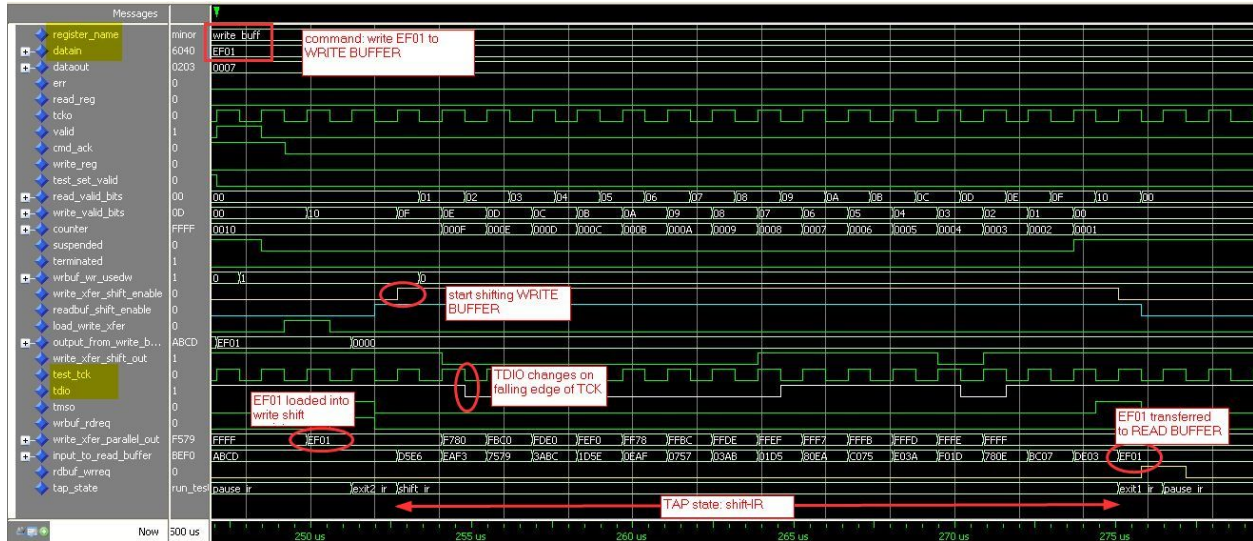


Figure 62: Second word (x'EF01') is loaded into WRITE BUFFER. It is shifted as in Fig.1, and reappears in the READ BUFFER at the end of the process.

[Ctrl+click here to get image from \[0\]](#)

6.5.2 Test 2: Overlapping writes

Two words are written to the WRITE BUFFER in quick succession, without even testing for WRITE BUFFER being full. This procedure is not recommended for general use: it represents a worst case test for the internal flow controls of the 8990 emulator.

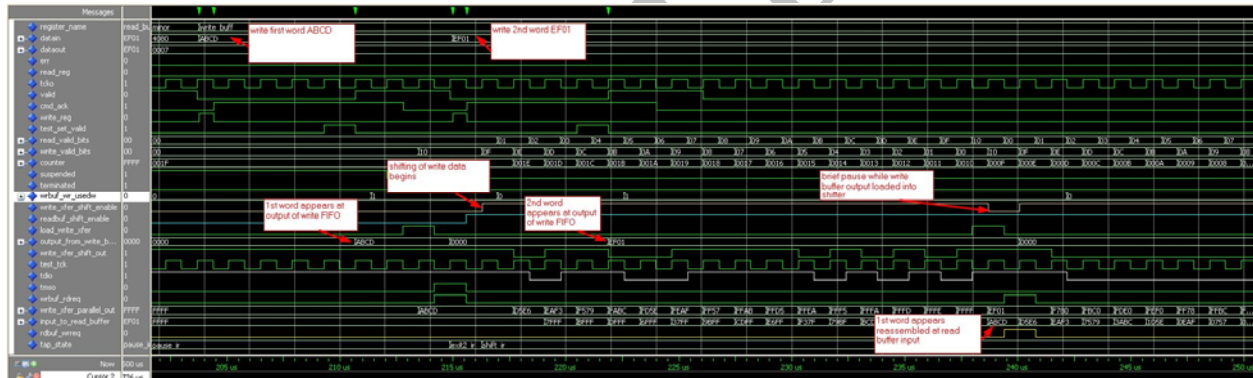


Figure 63: At $t=210\mu s$, the first word appears at the output of the WRITE BUFFER (FIFO). The word is loaded into the write shift register at $t=212\mu s$. At $222\mu s$, the second word appears at the FIFO output. At $t=238\mu s$, $write_valid_bits = 0$, and $load_write_xfer$ is asserted. The new value (EF01) appears in the write shift register. At the same time, the first word (ABCD) appears reassembled at the output of the READ shift register.

[Ctrl+click here to get image from \[0\]](#)

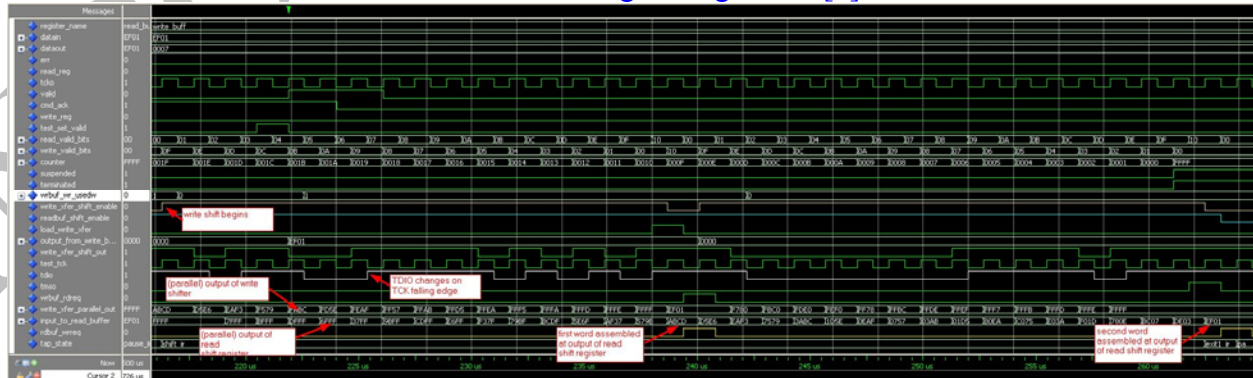


Figure 64: These waveforms cover the entire duration of the shifting of both words. Note that TAP state = Shift-IR from $216\mu s$ to $263\mu s$, without any pauses while the second word is being loaded into the WRITE shift register.

[Ctrl+click here to get image from \[0\]](#)

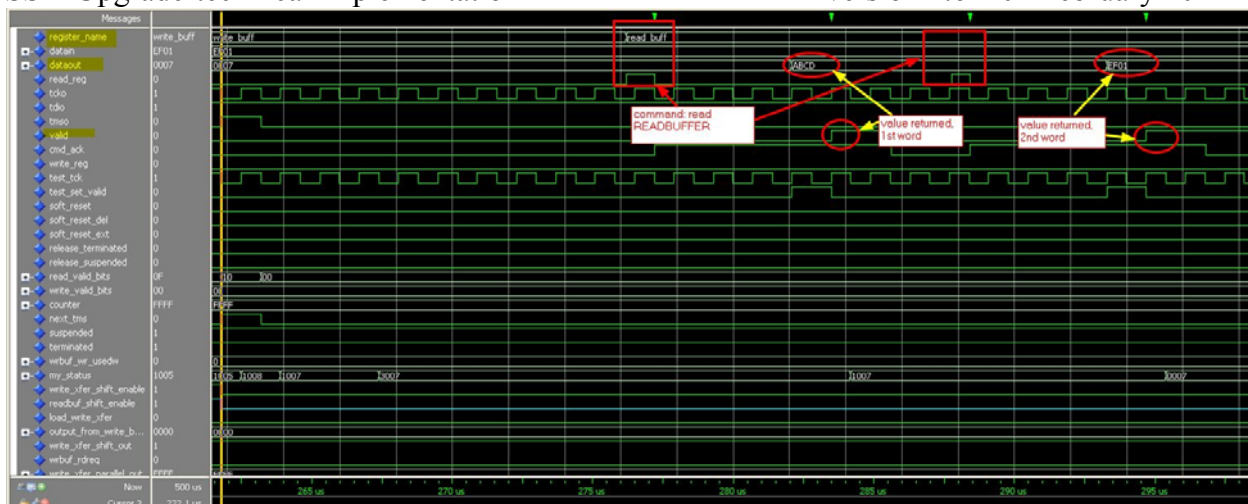


Figure 65: Two successive commands: read READ BUFFER return the words ABCD, EF01 written at the beginning of this sequence. The valid bit is returned to the host on each read command.

[Ctrl+click here to get image from \[0\]](#)

Note that the VHDL code for the emulator is the same for both tests; only the test bench has been modified to change the test sequence.

6.6 Tests equipment

6.6.1 Mechanical validation

6.6.1.1 Validation in CAD software

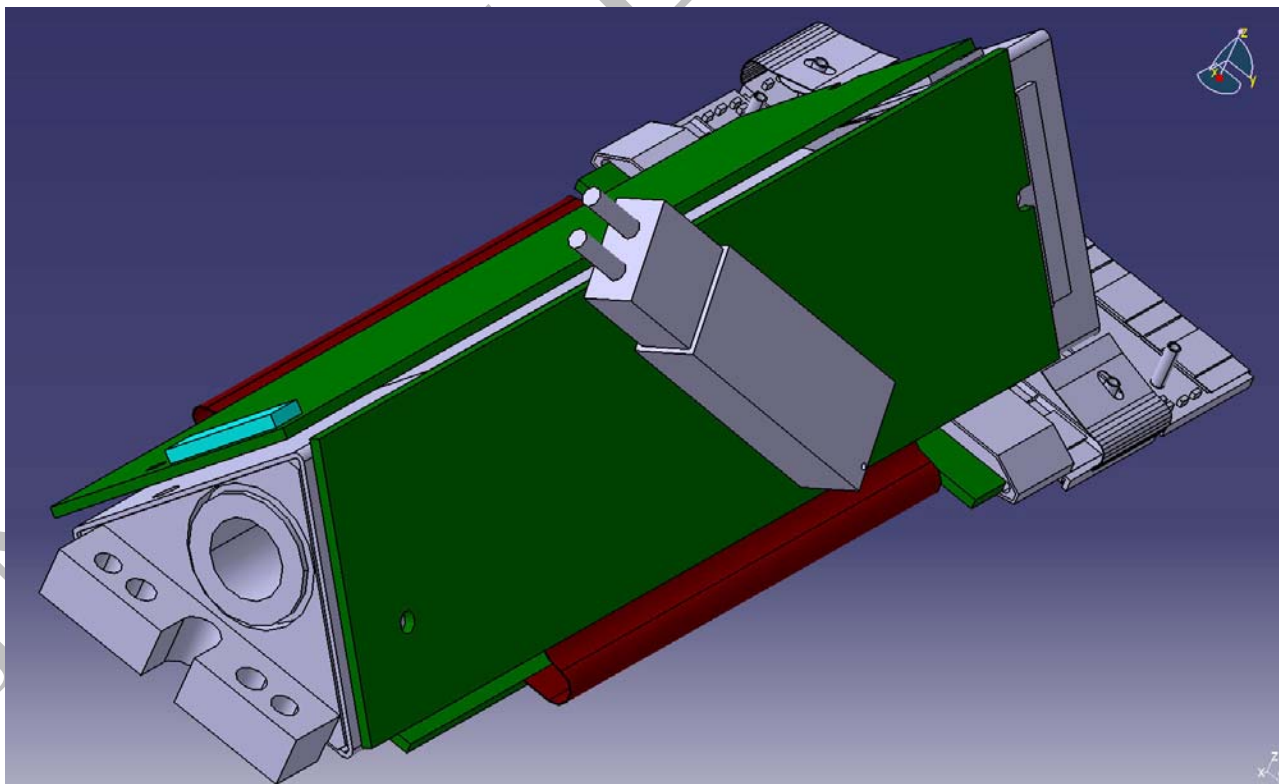


Figure 66: mechanical prototype of ladder card in place on a ladder (CAD drawing by Gérard Guilloux)

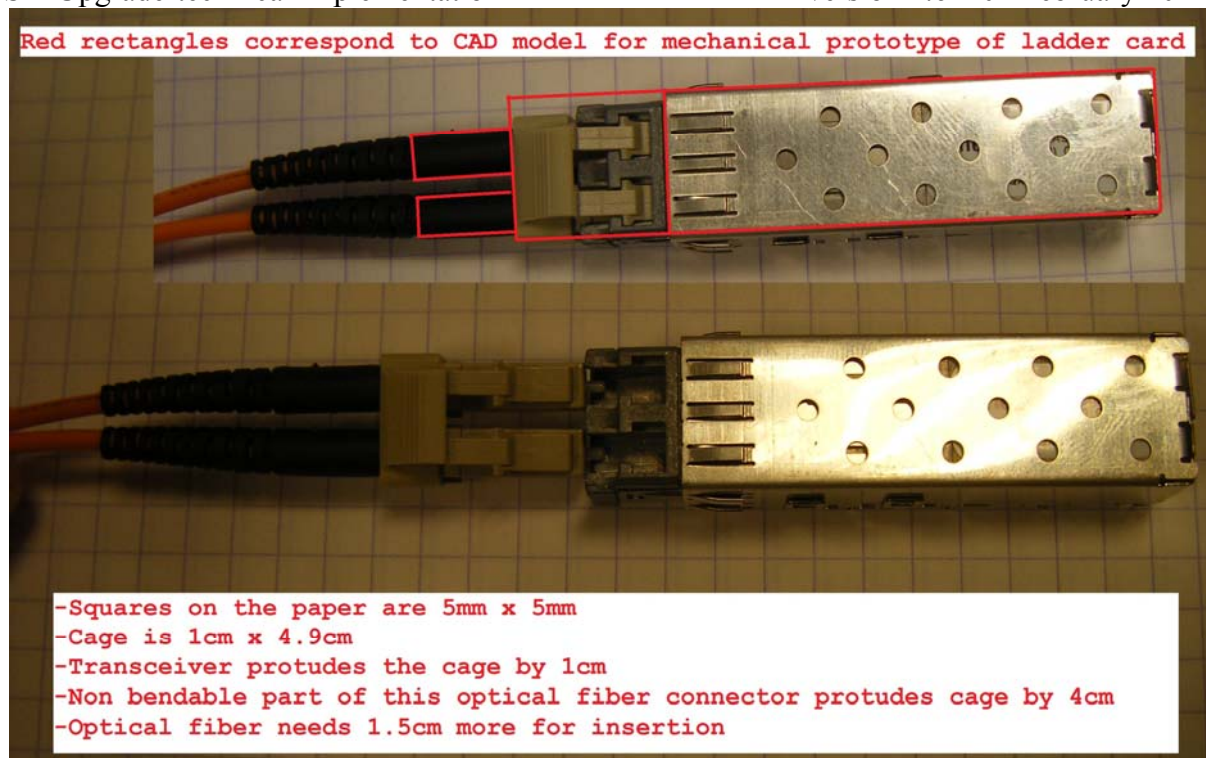


Figure 67: optical transceiver CAD model for mechanical prototype (CAD drawing by Gérard Guilloux)

6.6.1.2 Mechanical prototype of ladder card

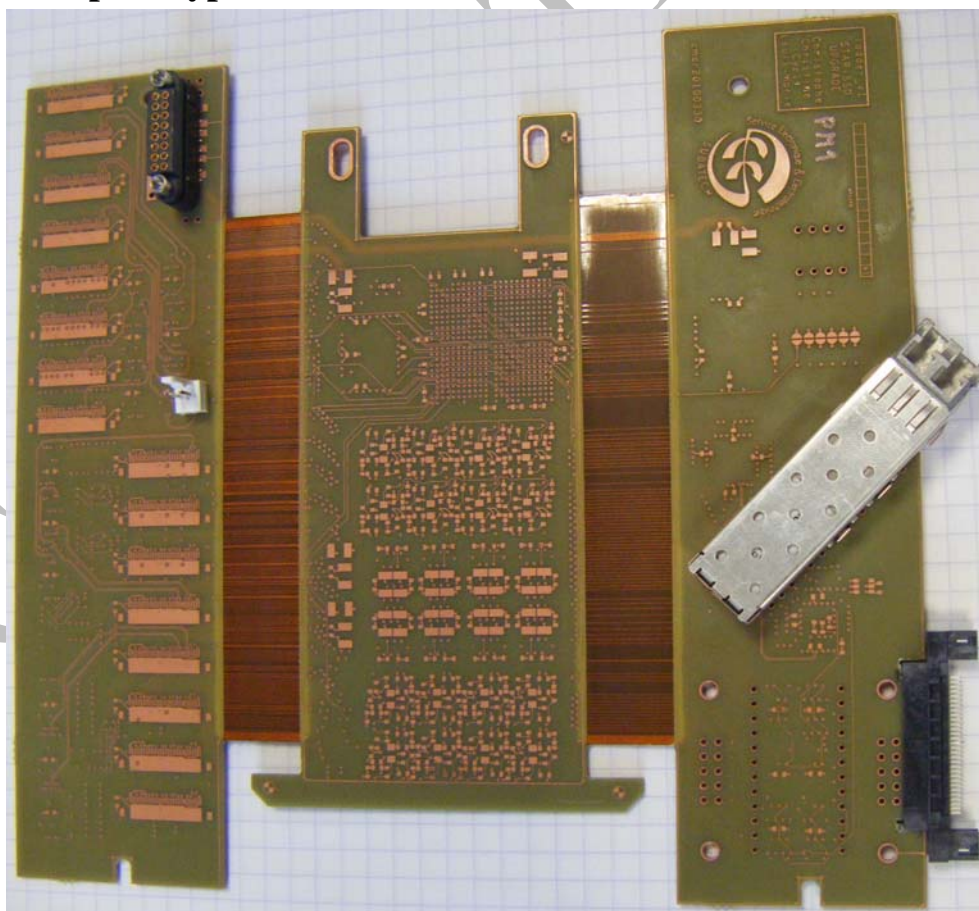


Figure 68: Mechanical prototype of ladder card

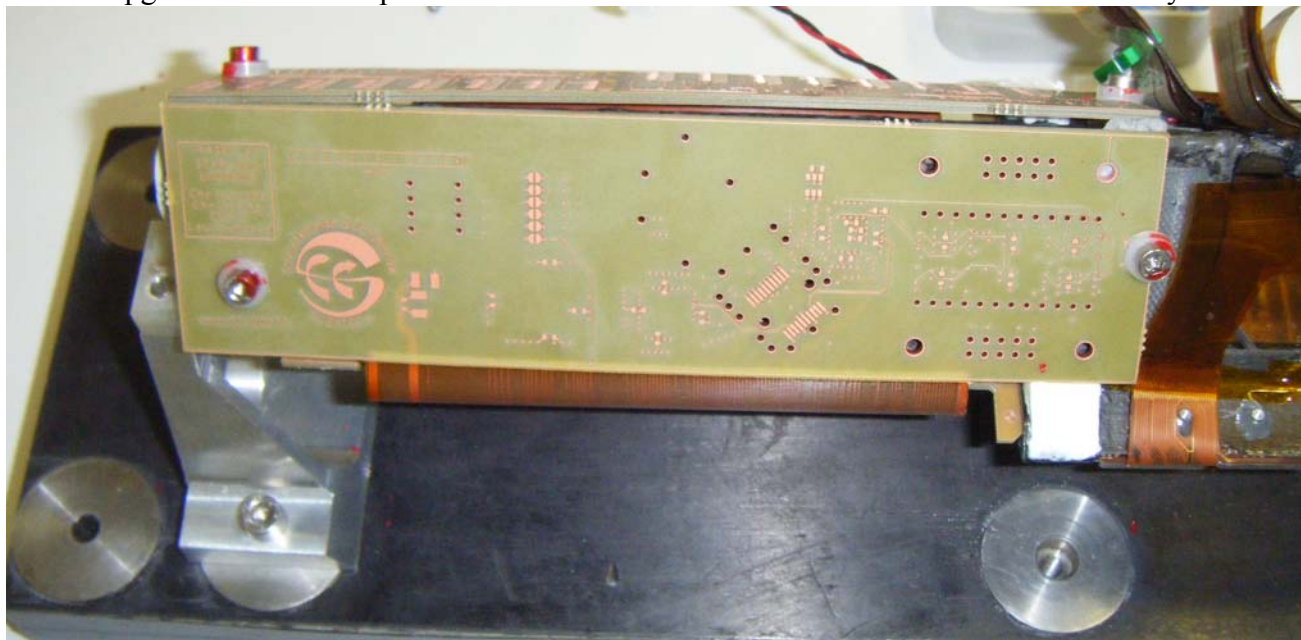


Figure 69: Mechanical prototype of ladder card fixed on a real ladder

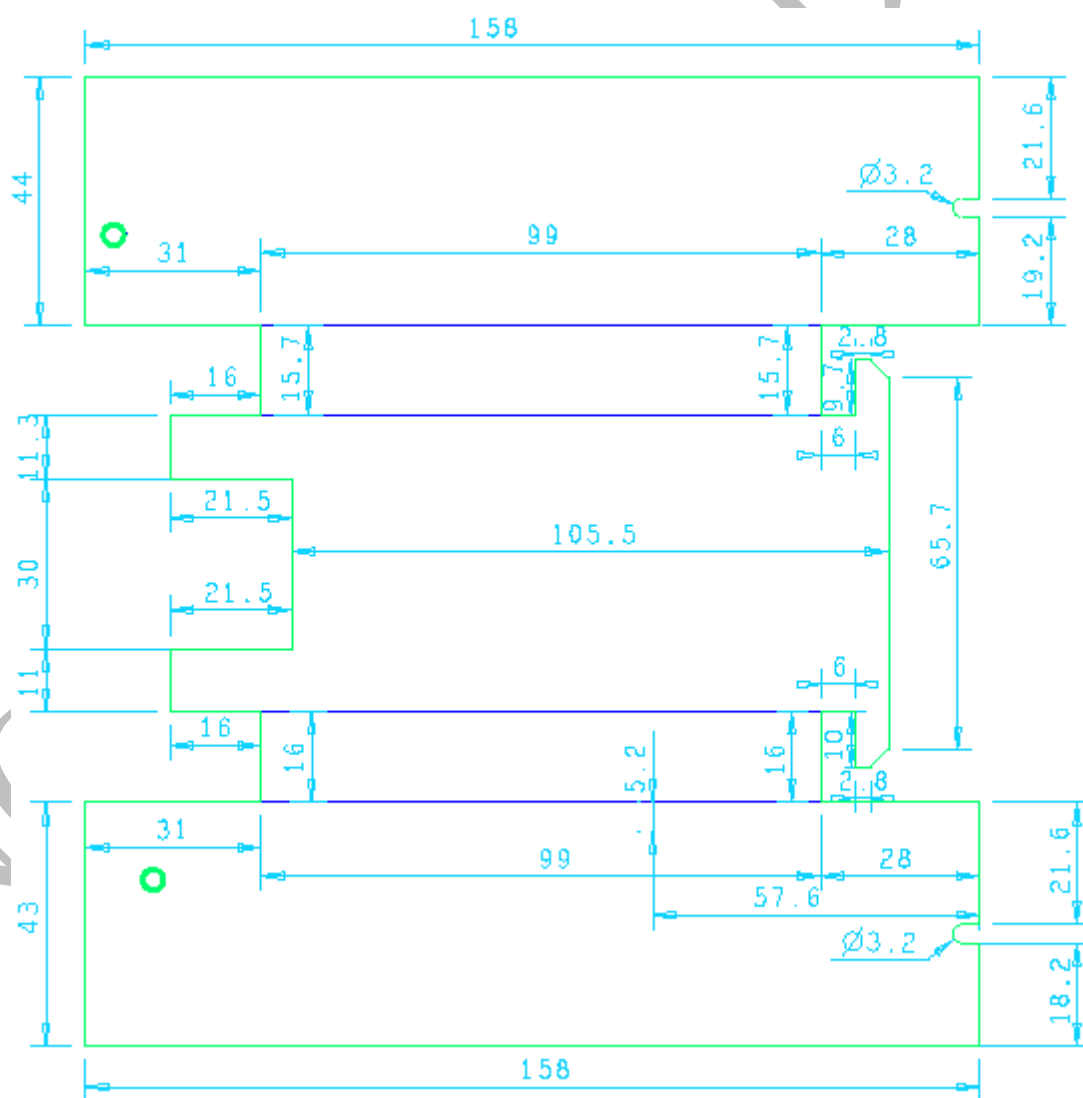


Figure 70: ladder card mechanical prototype dimensions (in mm)

6.6.1.3 Fake ladder end for mechanical validation of ladder card

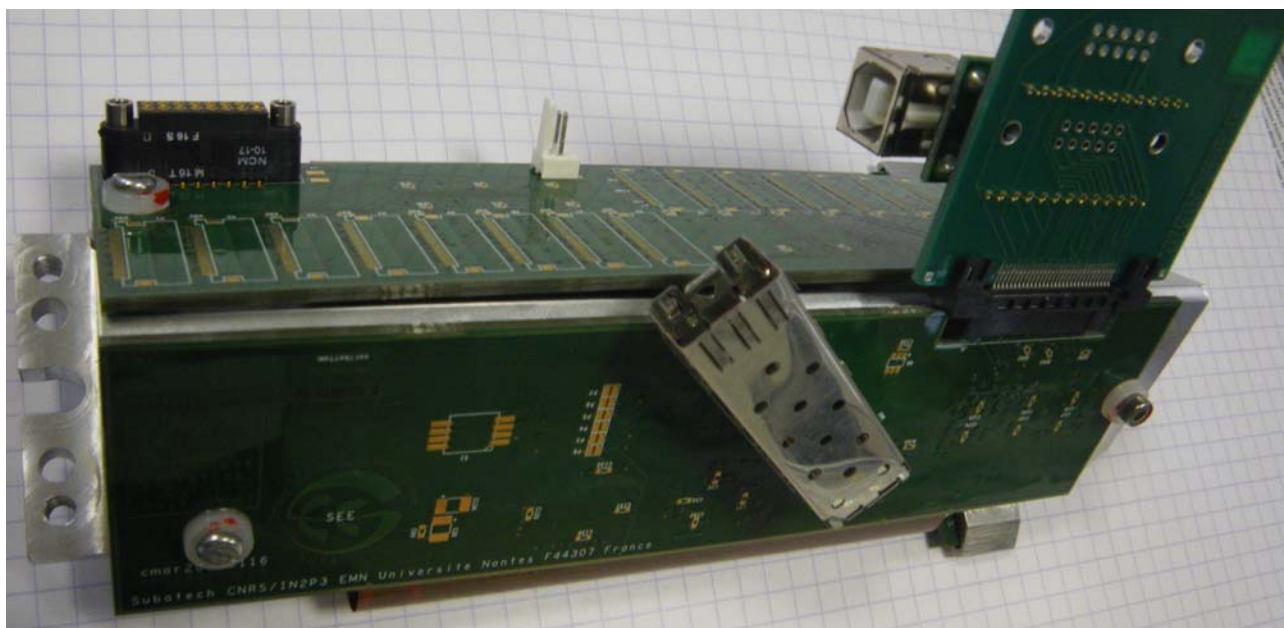


Figure 71: Mechanical test of prototype ladder card fixed on a fake ladder end and equipped with a debug card

6.6.1.4 Fake ladder for mechanical validation of ladder card and new support cylinder

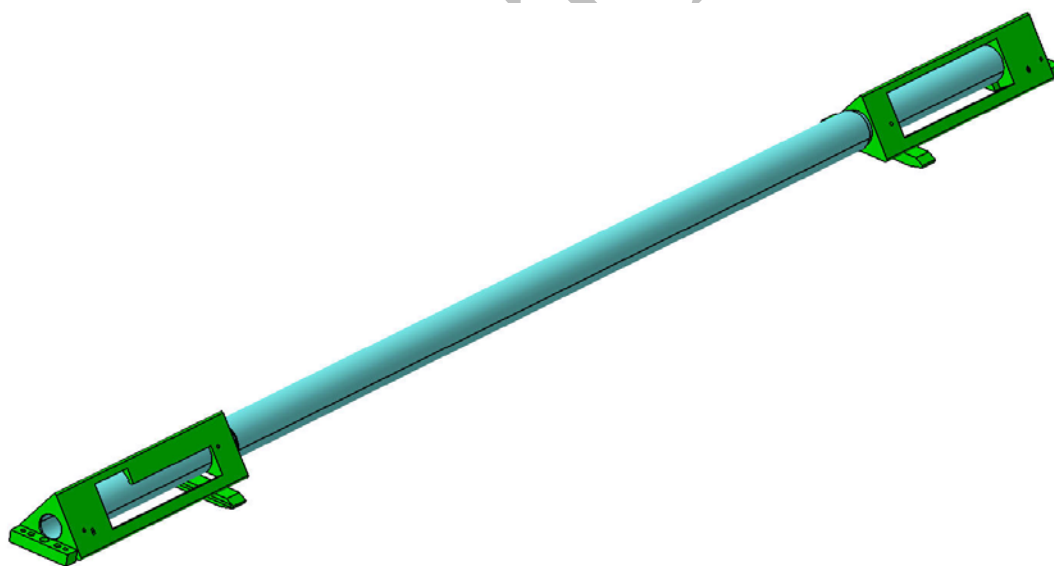


Figure 72: fake ladder (CAD drawing by Gérard Guilloux)

6.6.1.5 Problem with the shroud

There seems to be a problem with the shroud that is supposed to isolate SSD and TPC: It seems to be too close from the optical transceiver (see Figure 73).

WITHOUT ANY DIMENSION GIVEN, IT IS HARD TO TELL WHETHER THE PROBLEM IS OF THE ORDER OF CENTIMETERS OR OF MILLIMETERS.

Figure 74, show the ladder card version "mechanical prototype" (most probably used for the CAD model in Figure 73). Figure 75 show the ladder card version "prototype" (under test at BNL). Figure 76 show the ladder card version "preprod" (under development at Subatech)

The light blue area in these images identifies the place where access to both sides of the printed circuit is mandatory for normal insertion and operation of the optical transceiver. The pink area in these images identifies the place where the gasket sits (to avoid air flow leakage).

One can see that only the prototype version (Figure 79) allows for normal insertion and operation of the optical transceiver with no risk of air leakage.

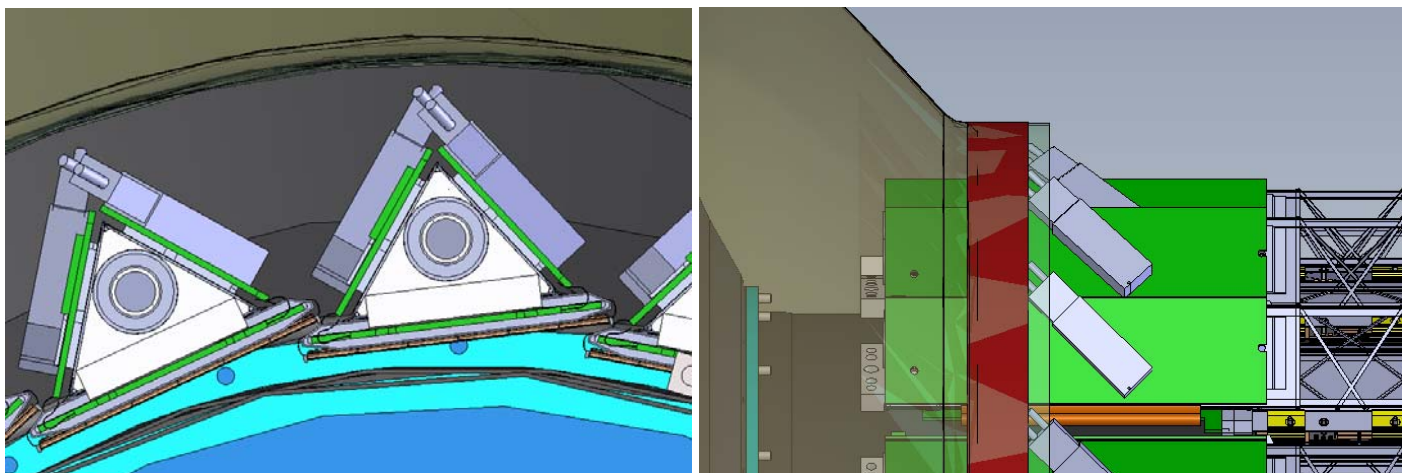


Figure 73: mechanical CAD drawing of ladders on the support, with the shroud on top. NO SCALE INDICATED!!!

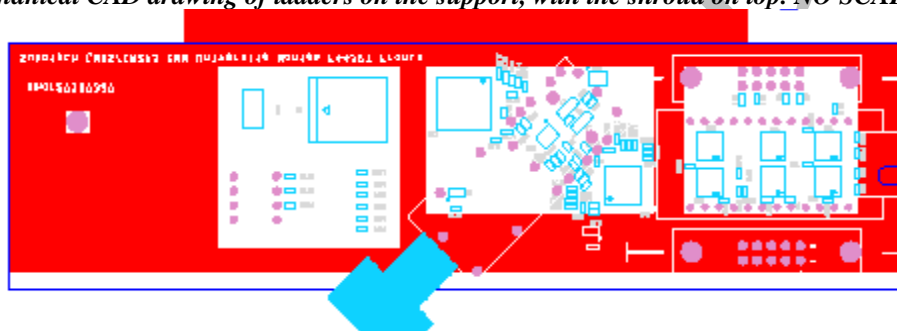


Figure 74: ladder card mechanical prototype with 45° rotated optical transceiver

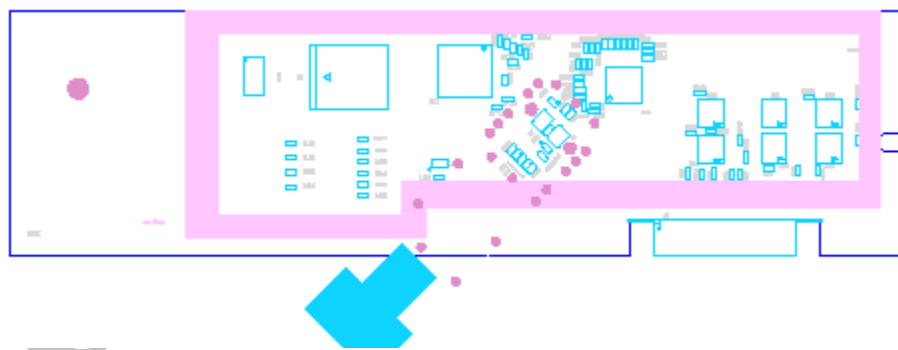


Figure 75: ladder card prototype with 45° rotated optical transceiver

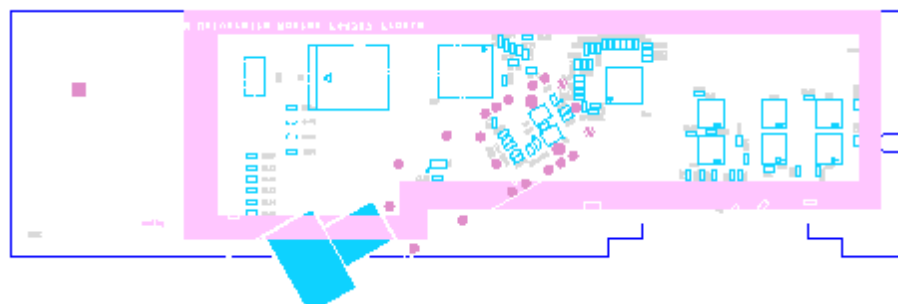


Figure 76: ladder card preprod with 30° rotated optical transceiver

6.6.2 Ladder card prototype

6.6.2.1 Ladder card printed circuit board dimensions

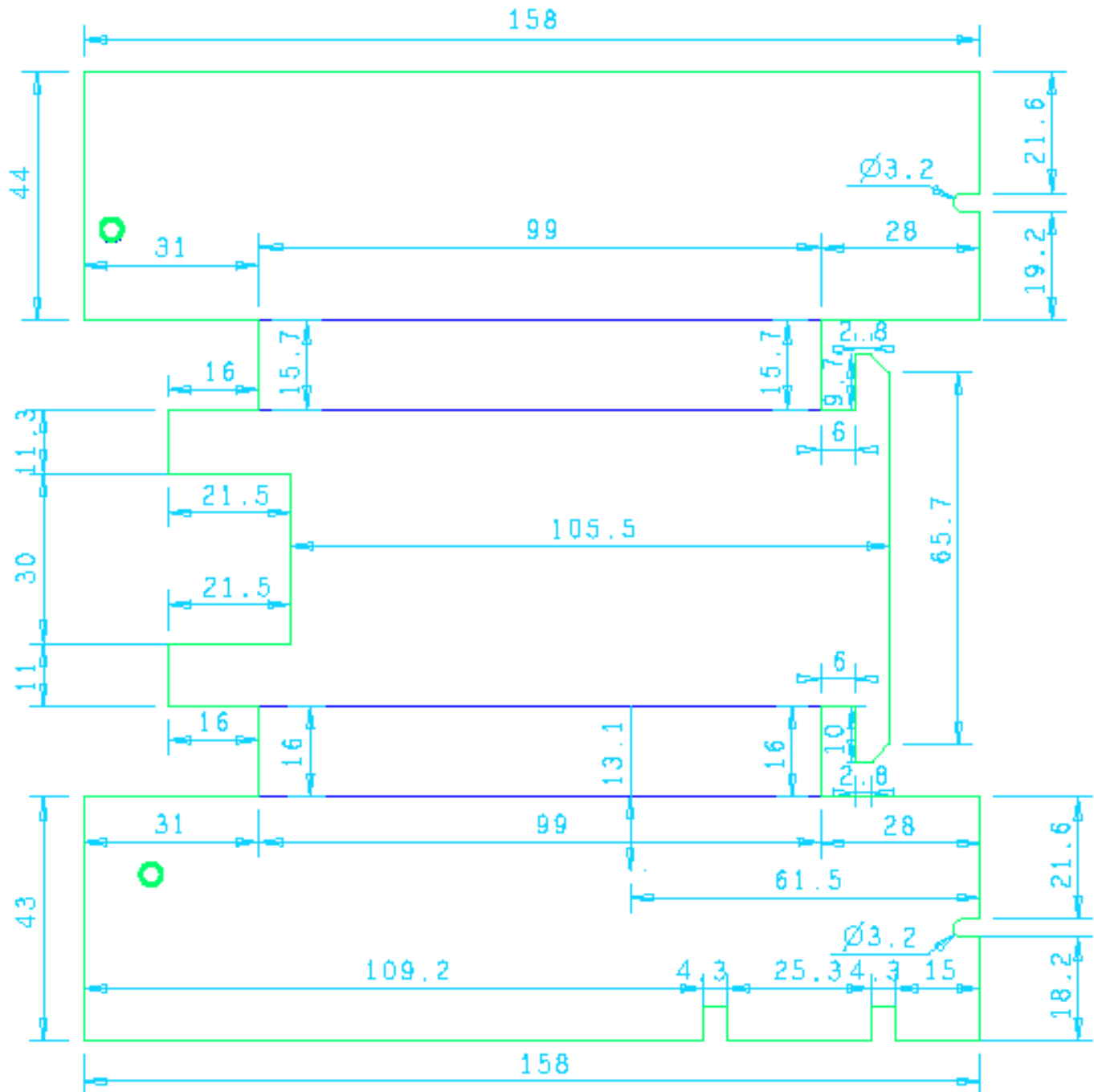


Figure 77: ladder card prototype dimensions (in mm)

6.6.2.2 Ladder card printed circuit board placement top (inside, cooled)

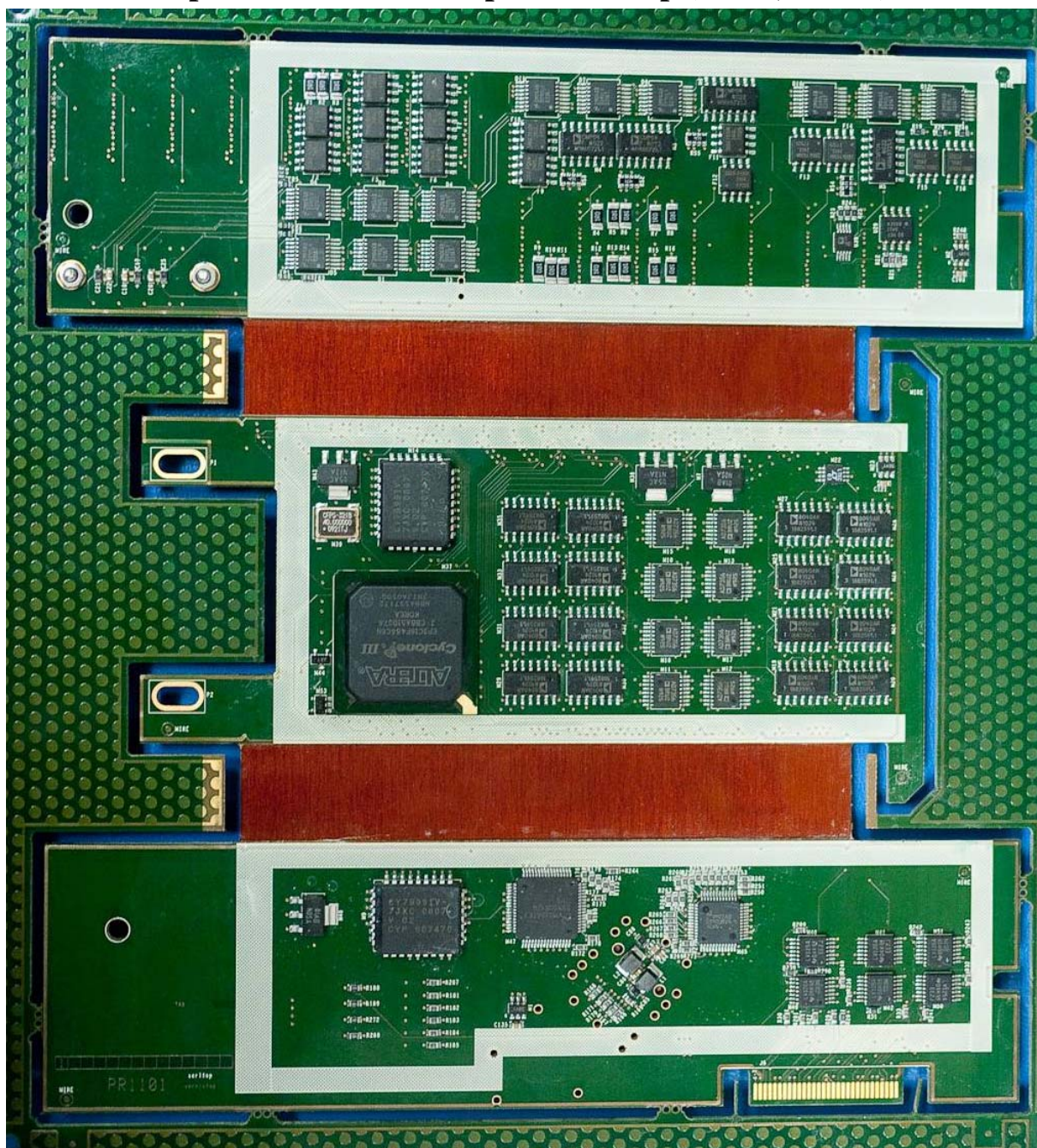


Figure 78: ladder card prototype circuit top

6.6.2.3 Ladder card printed circuit board placement bottom (outside, not cooled)

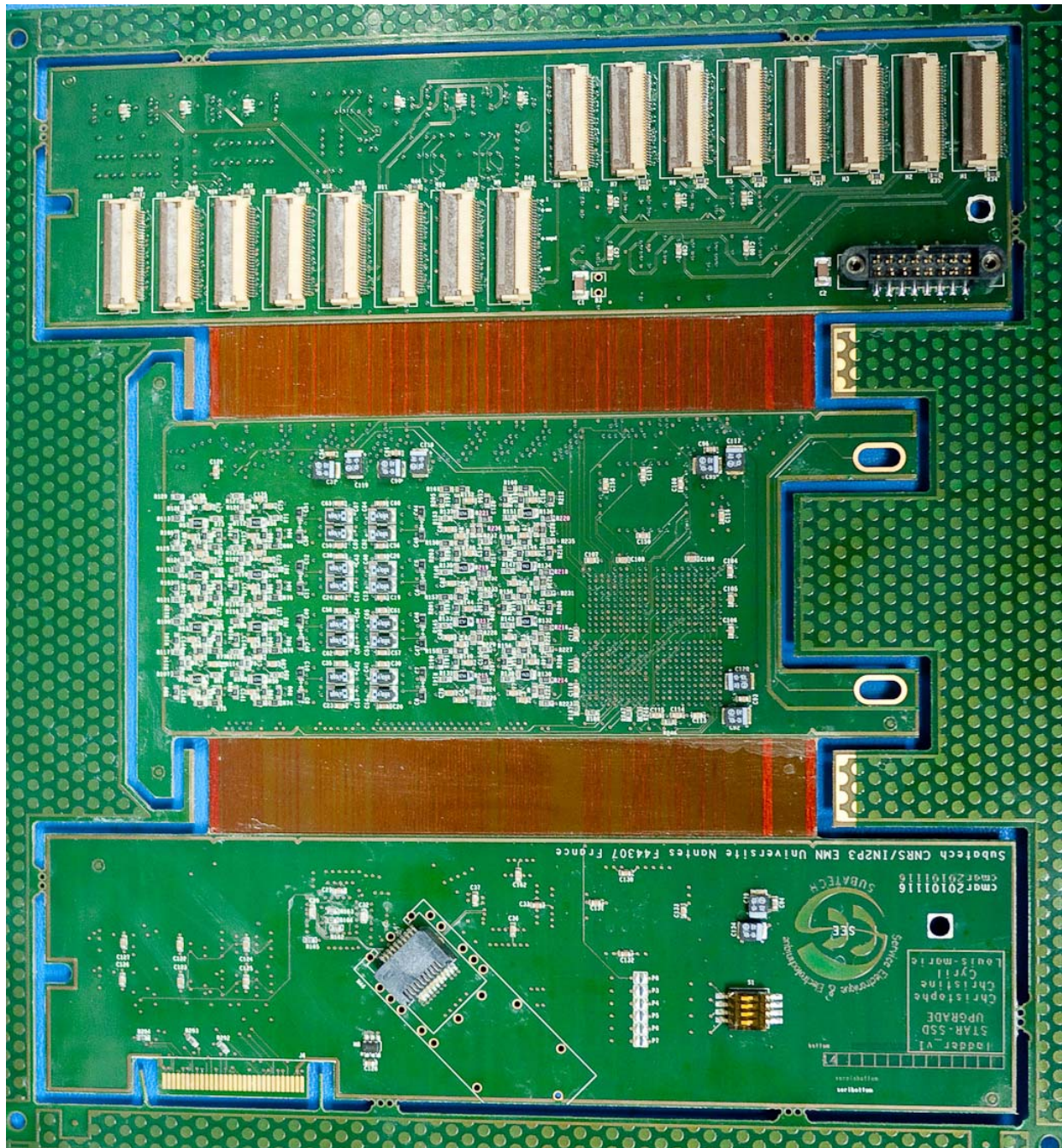


Figure 79: ladder card prototype circuit bottom

6.6.3 Debug card

6.6.3.1 Overview

The debug card is the interface between a computer and a card under test.

6.6.3.2 Debug card: FPGA configuration (FPGA-Blaster)

FPGA-Blaster connector is the 2x5 pin male HE10 closest to the LEDs. It is used to connect an Altera^[8] blaster (eg. USB-Blaster)

power	power	NC	NC	power
GND	+3.3V			GND
2	4	6	8	10
inout	out			inout
in	out	in		in
1	3	5	7	9
TCK	TDI	TMS		TDO
LVTTL	LVTTL	LVTTL	NC	LVTTL

Table 96: *FPGA-Blaster connector*

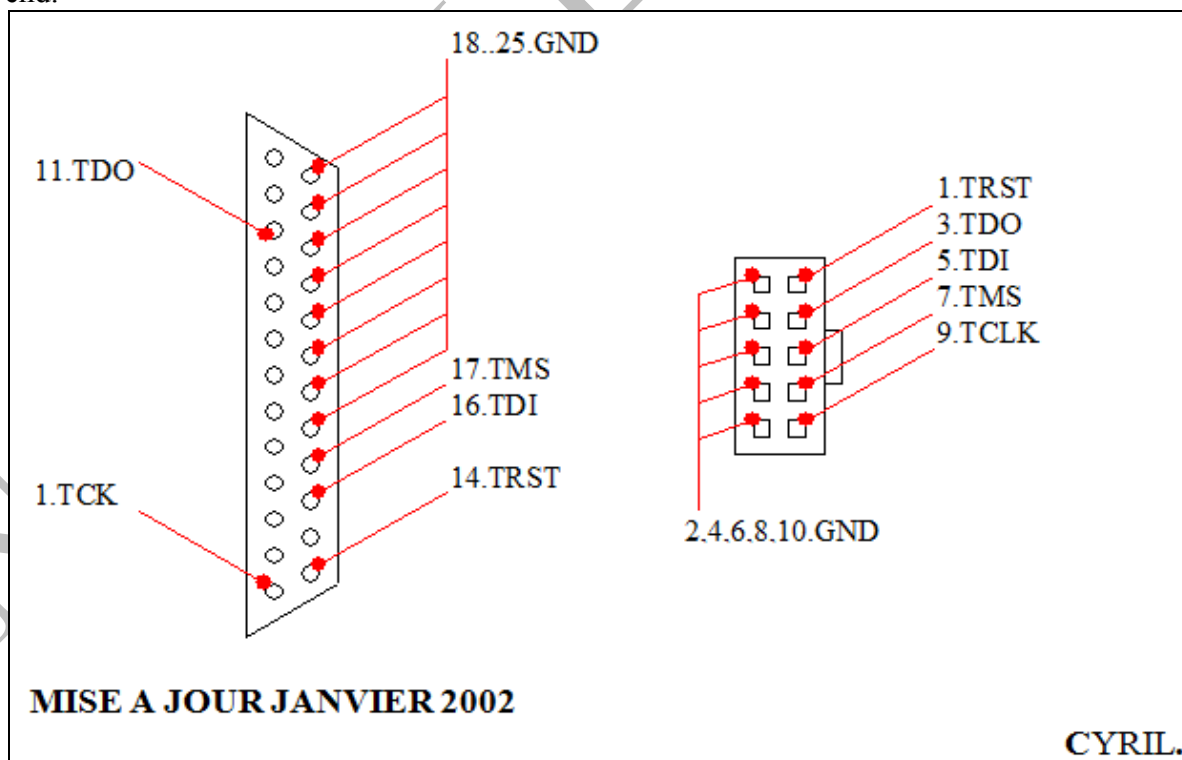
6.6.3.3 Debug card: JTAG slow-control

JTAG slow-control connector is the 2x5 pin male HE10 furthest from the LEDs. It is used to connect a STAR-SSD slow-control cable.

LVTTL	LVTTL	LVTTL	LVTTL	LVTTL
TCK	TMS	TDI	TDO	TRST
9	7	5	3	1
in	in	in	out	in
inout	inout	inout	inout	inout
10	8	6	4	2
GND	GND	GND	GND	GND
power	power	power	power	power

Table 97: *JTAG slow-control connector*

STAR-SSD slow-control cable has a male DB25 connector at one end and a 2x5pin female HE10 connector at the other end.

Figure 80: *STAR-SSD slow-control cable from PC parallel DB25 to STAR-SSD JTAG HE10*

6.6.3.4 Debug card: debug interface (to/from card under test)

direction	name	top	bottom	name	direction
Power2dbg	3.3V	1	2	Gnd	gnd
lad2pc	Jtag_sc_tdo	3	4	LED ok moins	lad2pc
lad2pc	Jtag_fpga_tdo	5	6	Gnd	gnd
lad2pc	Led7	7	8	usb_data(0)	bidir
gnd	Gnd	9	10	Gnd	gnd
pc2lad	Jtag_sc_trst	11	12	3.3V	Power2dbg
pc2lad	Jtag_sc_tdi	13	14	usb_data(4)	bidir
gnd	Gnd	15	16	usb_data(2)	bidir
Power2dbg	3.3V	17	18	Gnd	gnd
pc2lad	Jtag_sc_tms	19	20	usb_data(1)	bidir
pc2lad	Jtag_sc_tck	21	22	Led6	lad2pc
gnd	Gnd	23	24	Led5	lad2pc
pc2lad	Jtag_fpga_tck	25	26	Led4	lad2pc
pc2lad	Jtag_fpga_tdi	27	28	Gnd	gnd
gnd	Gnd	29	30	3.3V	Power2dbg
Power2dbg	3.3V	31	32	Gnd	gnd
gnd	Gnd	33	34	Dbg_pres=Gnd	pc2lad
lad2pc	Led1	35	36	Usb_rx_empty	pc2lad
lad2pc	Led2	37	38	Gnd	gnd
lad2pc	Led3	39	40	Usb_tx_full	pc2lad
bidir	usb_data(7)	41	42	Usb_present	pc2lad
gnd	Gnd	43	44	3.3V	Power2dbg
bidir	usb_data(5)	45	46	Gnd	gnd
bidir	usb_data(6)	47	48	Fpga_jtag_tms	pc2lad
Power2dbg	3.3V	49	50	Usb_ready_n	pc2lad
gnd	Gnd	51	52	Gnd	gnd
bidir	usb_data(3)	53	54	Led0	lad2pc
bidir	usb_reset_n	55	56	Usb_rd_n	lad2pc
lad2pc	LED ok plus	57	58	Usb_wr_n	lad2pc
gnd	Gnd	59	60	3.3V	Power2dbg

Table 98: debug interface connector

Colors in Table 98 do not use general typographic convention defined in §2.1 of this document

6.6.3.5 Debug card V2 (with UM245R USB FIFO daughter card)

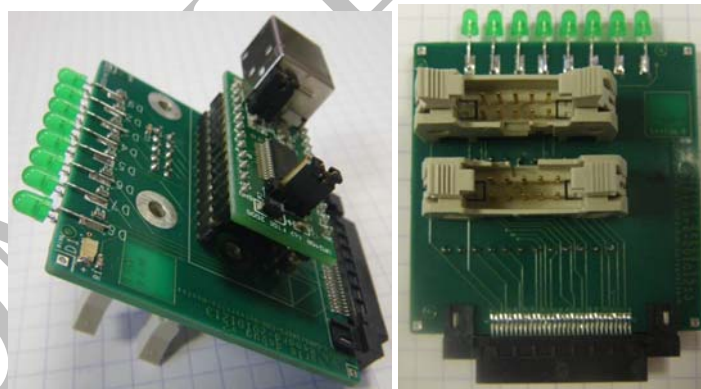


Figure 81: debug card V2 equipped with UM245R daughter card

⚠ The triangle on the edge connector points to pin 60 instead of pin 1. LED D1 lights up red when the card is connected in the right way.

⚠ On FPGA-Blaster connector (close to the LEDs): TDI and TDO are interchanged

6.6.3.6 Debug card V3 (with FT245R USB FIFO chip onboard)

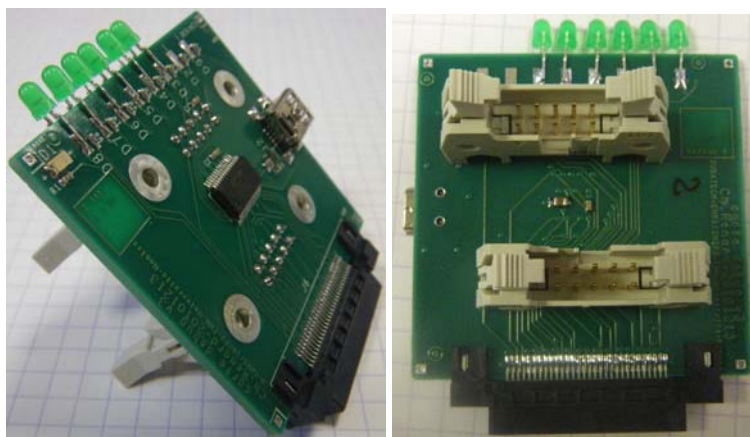


Figure 82: debug card V3

- ⚠ The triangle on the edge connector points to pin 60 instead of pin 1. LED D1 lights up red when the card is connected in the right way.
- ⚠ On FPGA-Blaster connector (close to the LEDs): TDI and TDO are interchanged
- ⚠ Signals usb_tx_empty and usb_rx_full are interchanged

6.6.4 Debug isolator (EVAL-ADUM4160EBZ USB isolator card)

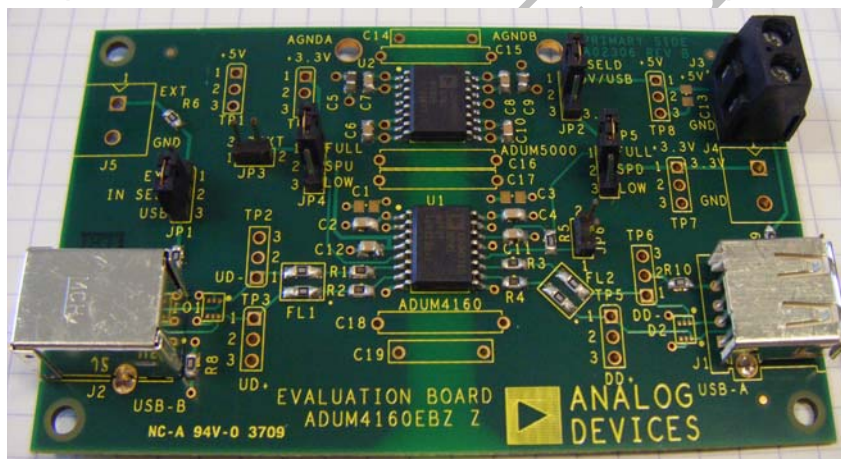


Figure 83: EVAL-ADUM4160EBZ USB isolator card

- ⚠ Ladder card analog reference (AGND) being connected to high voltage (~50V) on one side of the ladder (P side), particular care must be provided to electrically isolate the debug interfaces

6.6.5 Static Fake Hybrid card

6.6.5.1 Overview

Static Fake Hybrid card can be connected to the ladder card in place of an hybrid. It provides a static analog voltage in the range from -1V to +1V. There are sixteen different values available. The selection is done with a rotary switch.

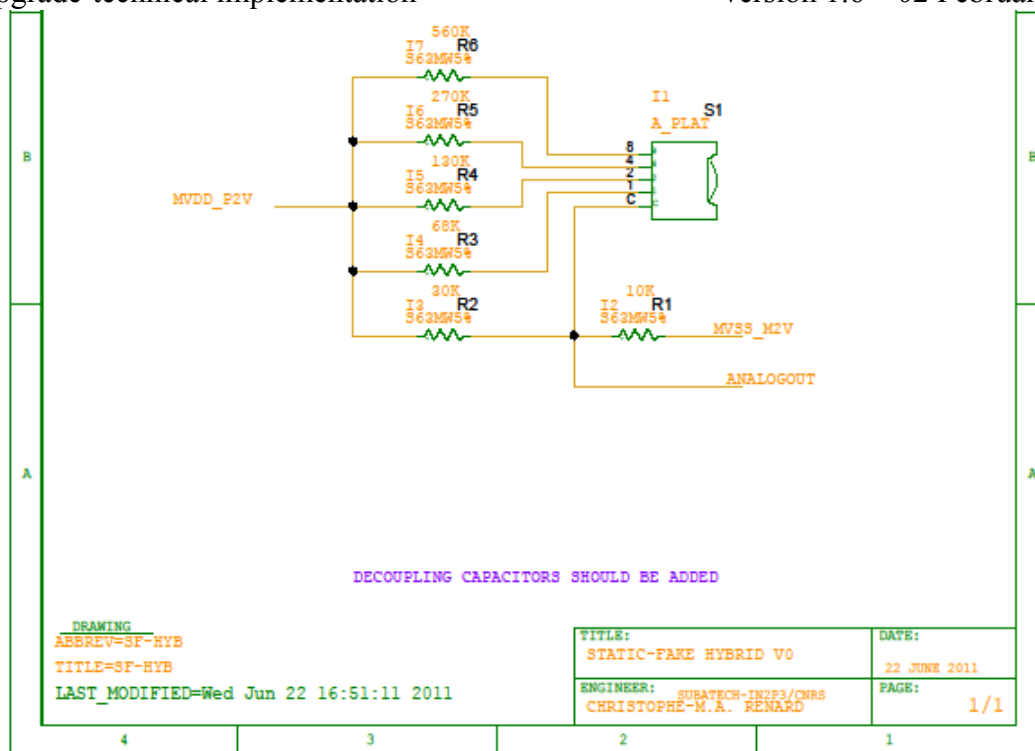


Figure 84: Static Fake Hybrid (SF-Hyb) card schematics proposed by Subatech.

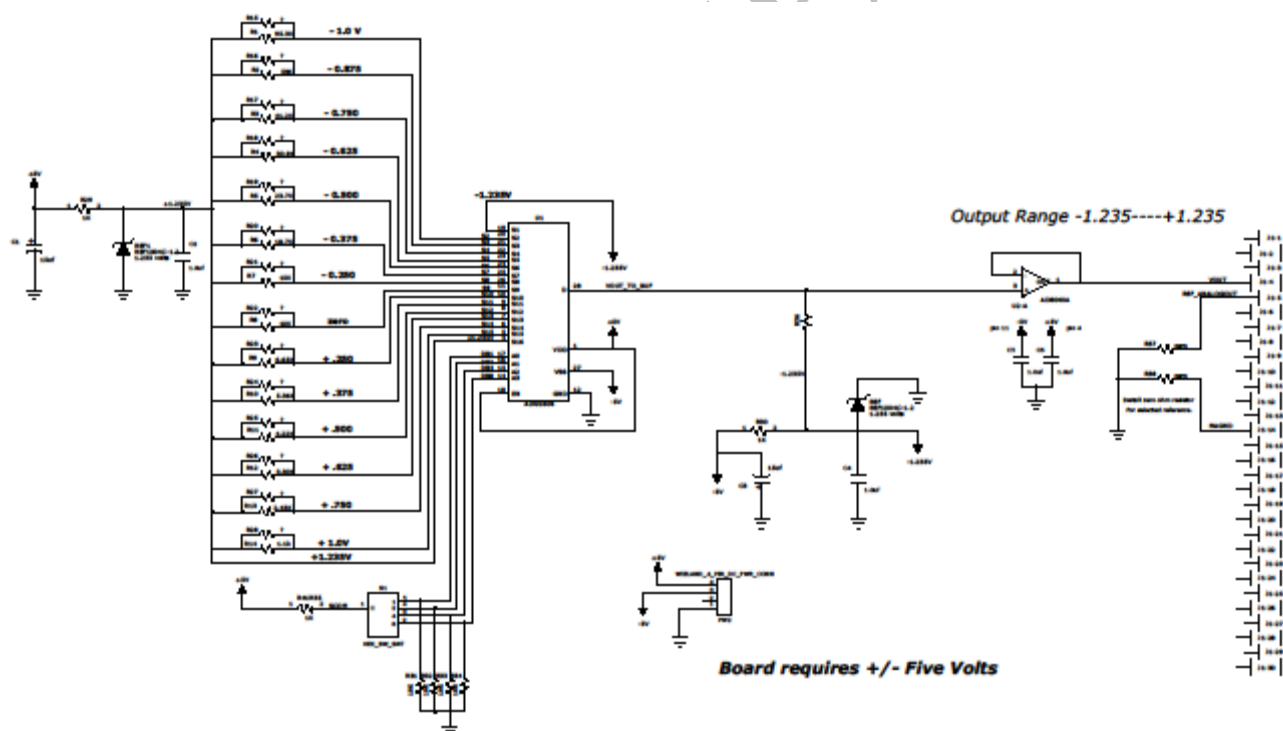


Figure 85: Static Fake Hybrid (SF-Hyb) card schematics implemented by BNL.

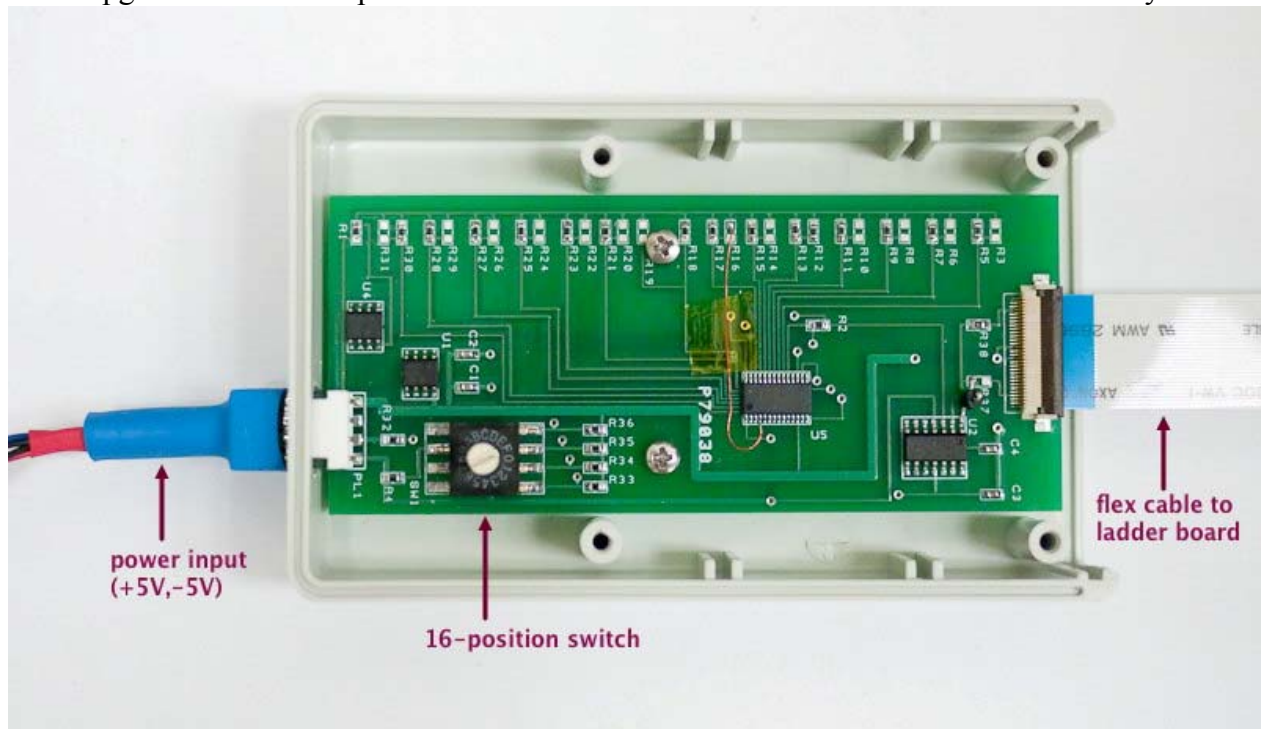


Figure 86: Static Fake Hybrid (SF-Hyb) card implemented by BNL.

6.6.6 Dynamic Fake Hybrid card

Figure 87: Dynamic Fake Hybrid (DF-Hyb) card schematics.

6.6.7 Quick RDO (QRDO)

6.6.7.1 Overview

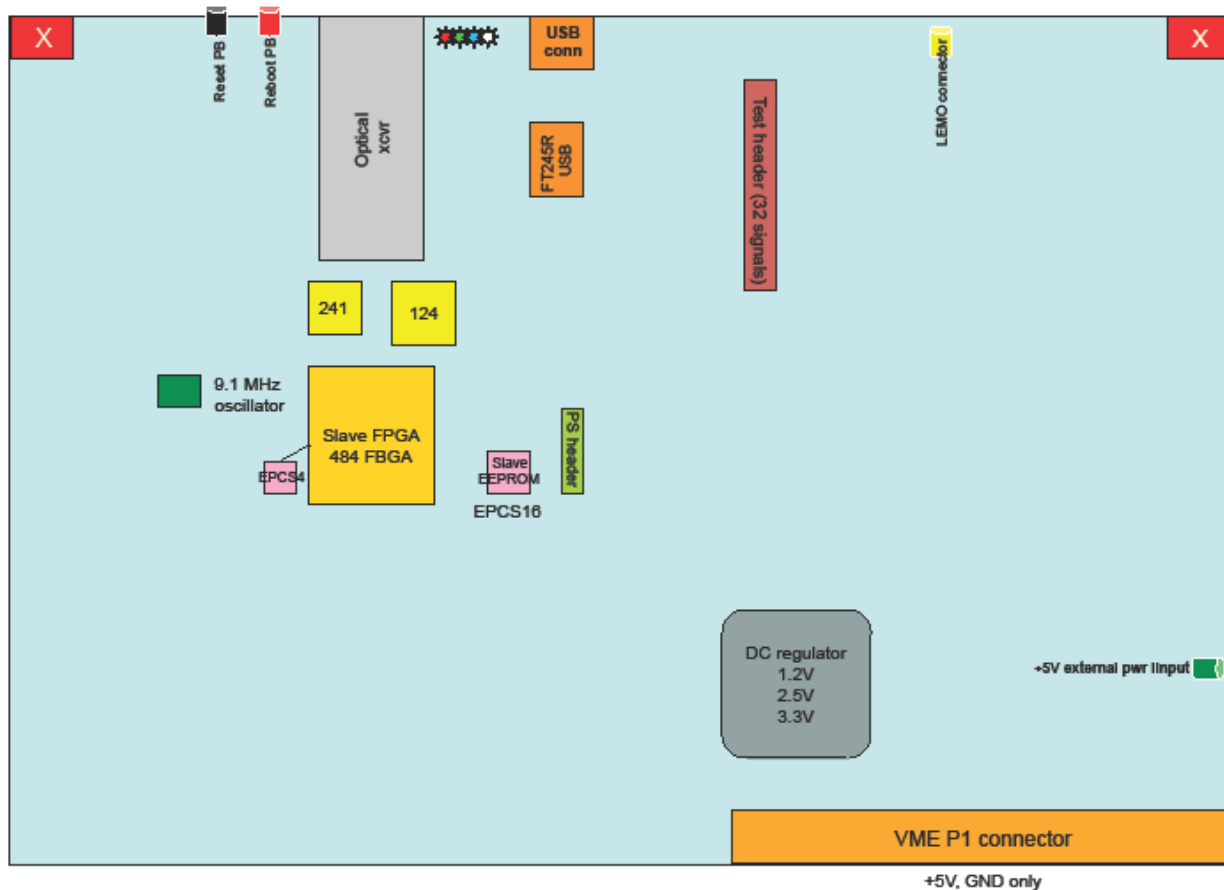


Figure 88: QRDO placement.

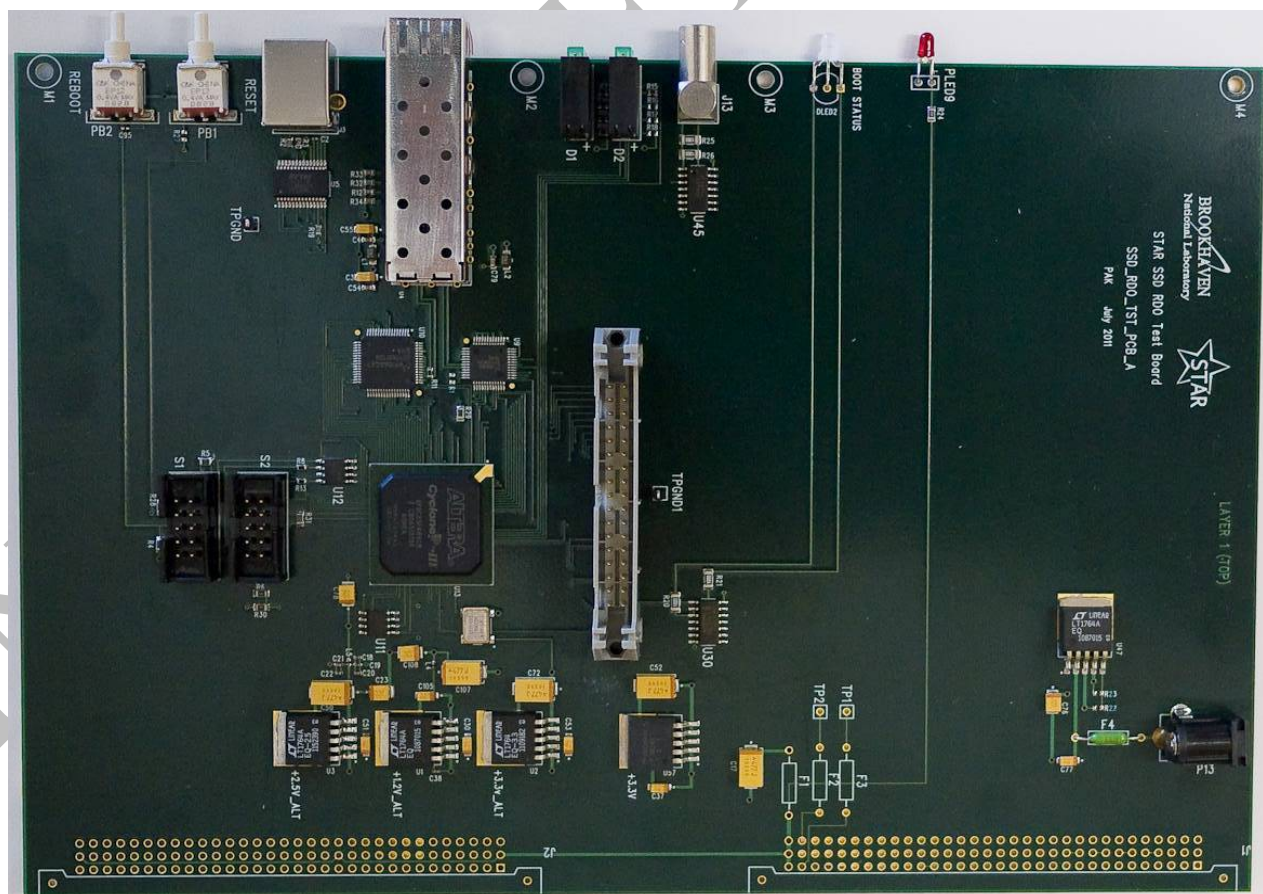


Figure 89: QRDO partially assembled.

6.6.8 FPGADC_star_v1

6.6.8.1 Overview

This card is the interface between a "debug card" (any version) and one (or two) ladder(s) equipped with the old electronics (ADC card & C2D2 card).

It contains the same 68pin connector as the old RDO card and the edge connector compatible with the debug cards.

Coupled with a debug card, it permits to:

- download the firmware into the different Altera FPGAs present in the programming chain (Altera Blaster).
- configure the ladders equipped with the old electronics and do some slow-control (JTAG).
- generate a trigger and get corresponding data (USB).

It also contains one ADC chain identical to the one on the prototype version of ladder card (AD7356^[20] + AD8040^[21] + LTC1662) and another ADC chain with a different level shifter mechanism that should be less sensible to power supplies' variations (AD7356^[20] + AD8040^[21] + AD5449^[28] + REF3318^[29]).

The firmware for the ALTERA^[8] CycloneIII EP3C16E144C8N^[9] FPGA was developed in VHDL using Synplify_pro^[10] as a synthesizer, Quartus II^[11] as place & route software and ModelSim-Altera^[12] as a simulation software. Complete project can be found in [0]

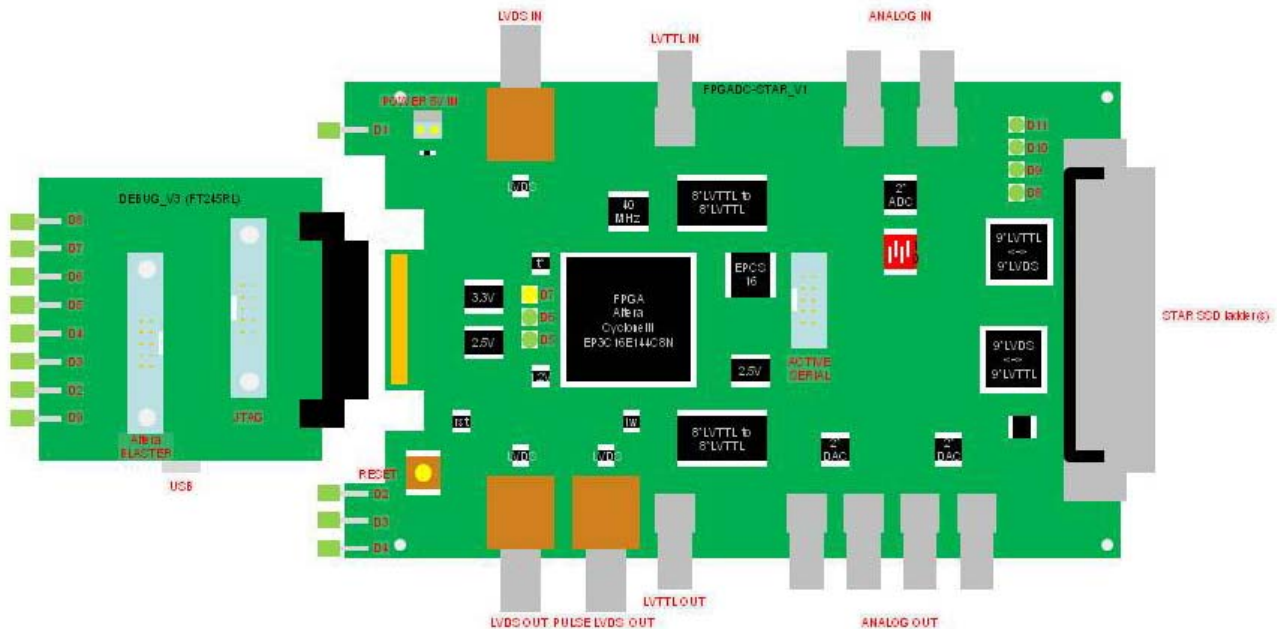


Figure 90: FPGADC_V1 pre-placement.

6.6.8.2 Debug interface: edge connector on FPGADC_v1 card

direction	name	top	bottom	n a m e	direction
Power2dbg	3.3V	1	2	Gnd	gnd
lad2pc	Jtag_sc_tdo	3	4	LED ok Gnd	lad2pc
lad2pc	Jtag_fpga_tdo	5	6	Gnd	gnd
lad2pc	Led7	7	8	usb_data(0)	bidir
gnd	Gnd	9	10	Gnd	gnd
pc2lad	Jtag_sc_trst	11	12	3.3V	Power2dbg
pc2lad	Jtag_sc_dti	13	14	usb_data(4)	bidir
gnd	Gnd	15	16	usb_data(2)	bidir
Power2dbg	3.3V	17	18	Gnd	gnd
pc2lad	Jtag_sc_tms	19	20	usb_data(1)	bidir
pc2lad	Jtag_sc_tck	21	22	Led6	lad2pc
gnd	Gnd	23	24	Led5	lad2pc
pc2lad	Jtag_fpga_tck	25	26	Led4	lad2pc
pc2lad	Jtag_fpga_tdi	27	28	Gnd	gnd
gnd	Gnd	29	30	3.3V	Power2dbg
Power2dbg	3.3V	31	32	Gnd	gnd
gnd	Gnd	33	34	Dbg_pres	pc2lad
lad2pc	Led1	35	36	Usb_rx_empty	pc2lad
lad2pc	Led2	37	38	Gnd	gnd
lad2pc	Led3	39	40	Usb_tx_full	pc2lad
bidir	usb_data(7)	41	42	Usb_present	pc2lad
gnd	Gnd	43	44	3.3V	Power2dbg
gnd	Gnd	45	46	Gnd	gnd
bidir	usb_data(5)	47	48	Fpga_jtag_tms	pc2lad
bidir	usb_data(6)	49	50	Usb_ready_n	pc2lad
Power2dbg	3.3V	51	52	Gnd	gnd
gnd	Gnd	53	54	Led0	lad2pc
bidir	usb_data(3)	55	56	Usb_rd_n	lad2pc
bidir	usb_reset_n	57	58	Usb_wr_n	lad2pc
lad2pc	LED ok 3.3V	59	60	3.3V	Power2dbg
gnd	Gnd				gnd

Table 99: Edge connector to debug interface

Colors in Table 99 do not use general typographic convention defined in §2.1 of this document

6.6.8.3 Front-End interface: SCSI68 connector on FPGADC_V1 card

It uses the same 2x34 (type SCSI68) Male right angle connector and the same buffer electronics as the old RDO card.

direction	name	top	bottom	n a m e	direction
pc2lad	Scsi_test_n	1	2	Scsi_test_p	gnd
Power2lad	Vcc_scsi	3	4	Vcc_scsi	Power2lad
lad2pc	Sesi_hold	5	6	Sesi_hold_p	gnd
Power2lad	Vcc_scsi	7	8	Vcc_scsi	Power2lad
pc2lad	Sesi_en_addr_n	9	10	Sesi_en_addr_p	gnd
Power2lad	Vcc_scsi	11	12	Vcc_scsi	Power2lad
pc2lad	Scsi_addr0_n	13	14	Sesi_addr0_p	bidir
Power2lad	Vcc_scsi	15	16	Vcc_scsi	Power2lad
pc2lad	Scsi_addr1_n	17	18	Sesi_addr1_p	gnd
Power2lad	Vcc_scsi	19	20	Vcc_scsi	Power2lad
pc2lad	Scsi_addr2_n	21	22	Sesi_addr2_p	lad2pc
Power2lad	Vcc_scsi	23	24	Vcc_scsi	Power2lad
pc2lad	Scsi_addr3_n	25	26	Sesi_addr3_p	lad2pc
Power2lad	Vcc_scsi	27	28	Vcc_scsi	Power2lad
lad2pc	Scsi_data0_n	29	30	Sesi_data0_p	lad2pc
gnd	Gnd	31	32	Gnd	gnd
lad2pc	Scsi_data1_n	33	34	Sesi_data1_p	lad2pc
gnd	Gnd	35	36	Sesi_tck	gnd
lad2pc	Scsi_data2_n	37	38	Sesi_data2_p	lad2pc
gnd	Gnd	39	40	Sesi_tdo	gnd
lad2pc	Scsi_data3_n	41	42	Sesi_data3_p	lad2pc
gnd	Gnd	43	44	Sesi_tdi	gnd
lad2pc	Scsi_data4_n	45	46	Sesi_data4_p	lad2pc
gnd	Gnd	47	48	Scsi_tms_fpga	lad2pc
lad2pc	Scsi_data5_n	49	50	Sesi_data5_p	lad2pc
gnd	Gnd	51	52	Scsi_tck_fpga	lad2pc
lad2pc	Scsi_data6_n	53	54	Sesi_data6_p	lad2pc
gnd	Gnd	55	56	Scsi_tdo_fpga	lad2pc
lad2pc	Scsi_data7_n	57	58	Sesi_data7_p	lad2pc
gnd	Gnd	59	60	Sesi_tdi_fpga	lad2pc
lad2pc	Scsi_data8_n	61	62	Sesi_data8_p	lad2pc
gnd	Gnd	63	64	Sesi_latchup	lad2pc
lad2pc	Scsi_data9_n	65	66	Sesi_data9_p	lad2pc
gnd	Gnd	67	68	Gnd	gnd

Table 100: SCSI68 connector to Front-End interface

Colors in Table 100 do not use general typographic convention defined in §2.1 of this document

Christophe-M.-A. Renard, Subatech, IN2P3/CNRS-L'UNAM, Nantes, F44307, France

Micheal-J. LeVine, BNL, Upton, NY, USA

Stéphane Bouvier, Subatech, IN2P3/CNRS-L'UNAM, Nantes, F44307, France

6.6.8.4 comparizon between two DACs: LTC1662 and AD5449^[28]

LTC1662 is used on ladder card version "prototype" while AD5449^[28] is a candidate to replace it on ladder card version "pre-production".

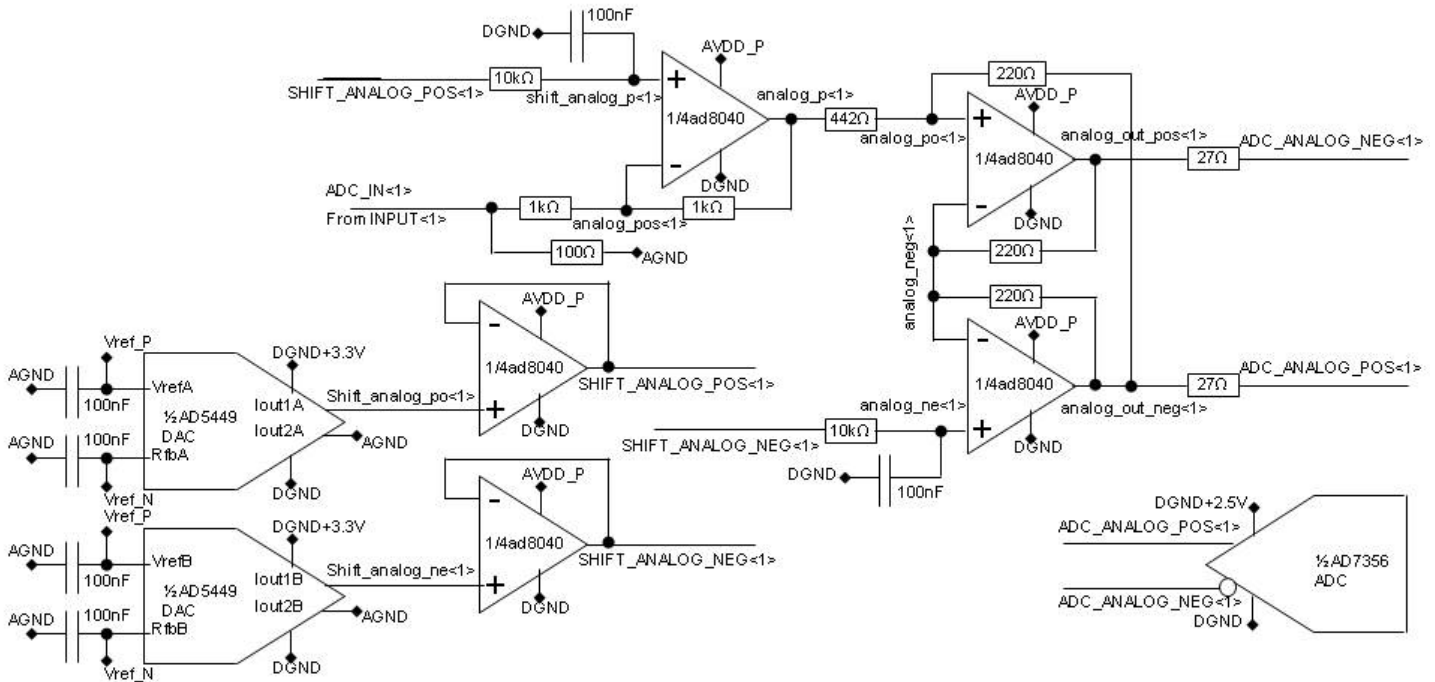


Figure 91: FPGAADC_V1 level shifter using AD5449^[28].

Vref_P = AGND+1.8V

Vref_N = AGND-1.8V

6.7 Tests of the ladder card

6.7.1 Tests using debug card

These tests can be done using debug card version v2 or v3 (see §5.12 Debug interface).

6.7.1.1 Test using debug card: ladder FPGA programming (FPGA-blaster connector).

connector pin out: Table 96: *FPGA-Blaster connector*.

SOF file and complete VHDL project are on the web^[0]

6.7.1.2 Test using debug card: ladder status read (JTAG-slow-Control connector).

Verify the switch setup on the ladder card (see §5.5 Ladder card switches).

Connector pin out: Table 97: *JTAG slow-control connector*.

See: Table 64: *ladder-FPGA slow-control registers*, version register (Table 79: *ladder-FPGA slow-control register (version)*), identity register (Table 81: *ladder-FPGA slow-control register (identity)*), status register (Table 72: *ladder-FPGA slow-control register (status)*) and temperature register (Table 80: *ladder-FPGA slow-control register (temperature)*)

6.7.1.3 Test using debug card: ladder DAQ setup (JTAG-slow-Control connector).

The serial protocol used to set the DAC values has to be verified, using an oscilloscope (Figure 30: *schematics of the analog part in the ladder card*^{[20][21][22]}). Some of the slow-control registers in the ladder FPGA are used to set the output value of the two DACs that control the level shifter: You write the values in one register (Table

74: *ladder-FPGA slow-control register (level-shifter DAC values)*); You cycle a bit in another register (Table 73: *ladder-FPGA slow-control register (config)*) to start the loading into the DACs.

6.7.1.4 Test using debug card: level-shifter with static voltage using SF-Hyb card (JTAG-slow-Control connector and USB connector).

Once you are able to set the DAC values, you need to set an analog voltage at one card input. This voltage should be between -1V and 0V or between 0V and +1V depending which side (P or N) you want to test. A simple Static-Fake-Hybrid (SF-Hyb) card (see §6.6.5 *Static Fake Hybrid card* and *Figure 84*) can be used to set a static voltage. Then, use an oscilloscope to verify that you are able to shift it to a value that is suitable for the ADC. See §5.10.3 *Level-shifter (from Alice128[4][5] to ADC)* to adjust DAC values in a way that gives the correct ADC inputs.

6.7.1.5 Test using debug card: one ADC with static voltage using SF-Hyb card (JTAG-slow-Control connector and USB connector).

Once you are able to set a static analog value at the ADC input, you need to read the ADC. The code of the ladder FPGA has to be modified to read the DAC via USB instead of via GBIC.

6.7.1.6 Test: 16 ADCs with static voltage (JTAG-slow-Control connector and USB connector).

Once you are able to read one ADC value in static mode, you need to verify the 16 analog channels. With nothing connected on the inputs, all 16 channels should give almost the same value.

6.7.1.7 Test: ladder packer (JTAG-slow-Control connector and USB connector).

Once you are able to read the 16 ADCs, you need to verify the data packer.

6.7.2 Tests using GBIC interface

These tests can be done either using a RDO card or a QRDO card.

6.7.2.1 Test using GBIC interface: ladder FPGA programming.

Now that you have done all tests using the debug card, it is time to verify that it works as well in normal mode, using the GBIC interface.

Verify the switch setup on the ladder card (see §5.5 *Ladder card switches*).

See: §4.12.2 *EPSC4*, Table 52: *rdoU slave-FPGA VME registers*, Table 53: *rdoU slave-FPGA VME FPGA_config_cmd register cmd bits and associated commands* and 4.13 Giga Bit Interface (GBIC) SOF file and complete VHDL project are on the web^[0]

6.7.2.2 Test using GBIC interface: ladder status read.

See: Table 52: *rdoU slave-FPGA VME registers*, Table 64: *ladder-FPGA slow-control registers*, version register (Table 79: *ladder-FPGA slow-control register (version)*), identity register (Table 81: *ladder-FPGA slow-control register (identity)*), status register (Table 72: *ladder-FPGA slow-control register (status)*) and temperature register (Table 80: *ladder-FPGA slow-control register (temperature)*)

6.7.2.3 Test using GBIC interface: ladder DAQ setup.

The serial protocol used to set the DAC values has to be verified, using an oscilloscope (*Figure 30: schematics of the analog part in the ladder card*^{[20][21][22]}). Some of the slow-control registers in the ladder FPGA are used to set the output value of the two DACs that control the level shifter: You write the values in one register (Table 74: *ladder-FPGA slow-control register (level-shifter DAC values)*); You cycle a bit in another register (Table 73: *ladder-FPGA slow-control register (config)*) to start the loading into the DACs.

6.7.2.4 Test using GBIC interface: level-shifter with static voltage using SF-Hyb card.

Once you are able to set the DAC values, you need to set an analog voltage at one card input. This voltage should be between -1V and 0V or between 0V and +1V depending which side (P or N) you want to test. A simple Static-Fake-Hybrid (SF-Hyb) card (see §6.6.5 *Static Fake Hybrid card* and *Figure 84*) can be used to set a static voltage. Then, use an oscilloscope to verify that you are able to shift it to a value that is suitable for the ADC. See §5.10.3 *Level-shifter (from Alice128[4][5] to ADC)* to adjust DAC values in a way that gives the correct ADC inputs.

6.7.2.5 Test using GBIC interface: ADC with static voltage using SF-Hyb card.

Once you are able to set a static analog value at the ADC input, you need to read the ADC.

6.7.2.6 Test using GBIC interface: ladder ADC with static voltage using 16 SF-Hyb card.

Once you are able to read one ADC value in static mode, you need to verify the 16 analog channels.

6.7.2.7 Test using GBIC interface: ladder packer.

Once you are able to read the 16 ADCs, you need to verify the data packer.

6.7.2.8 Test using GBIC interface: ladder ADC with dynamic voltage using DF-Hyb card.

Once you are able to read all the ADCs in static mode, you need to replace the SF-Hyb cards by DF-Hyb card (§6.6.6 *Dynamic Fake Hybrid card* and *Figure 87: Dynamic Fake Hybrid (DF-Hyb) card schematics.*) to see how the system reacts and learn how to adjust the ADC clock to the timing of the Fake Hybrids.

6.7.2.9 Test using GBIC interface: ladder ADC with a real hybrid.

Next step is to connect a real hybrid (there are two hybrids per module). It is better to do in only after you have a working (Q)RDO to avoid developing twice the slow-control to configure the hybrid.

6.8 Planning

07 January 2007 email from Tim Hallman: "A question"

09 July 2007 Proposal for upgrade versions "a" and "b" on the web^[38].

03 April 2008 Proposal for upgrade version "g": only optical link between rdoU and ladderU on the web^[38].

18 August 2009 Version 0.1 of this "master document" circulated

28 January 2010 First MoU signed

March 2010 First attempt for a contract between BNL and Subatech

04 May 2010 ladder card version "mechanical prototype" at BNL (not to be populated)

16 November 2010 final files for ladder card version "prototype" on the web^[0] and sent for fab

03 January 2011 debug card versions "2" and "3" at Subatech (unpopulated)

07 January 2011 ladder card version "prototype" at BNL (unpopulated)

24 January 2011 First version of VME FPGA pinout version "00" on the web^[0].

04 February 2011 debug card versions "2" and "3" at BNL (populated)

08 February 2011 ladder card version "prototype" at BNL (populated)

18 March 2011 final files for interposer version "classe 10" on the web^[0]

21 March 2011 final files for interposer version "classe 10" sent for fab

03 April 2011 Howard out of the project (replaced by Jim as project leader)

07 April 2011 Jim awakes the idea to write a contract for BNL to pay for Subatech's technical work

15 April 2011 draft version 1 of proposal for contract

20 May 2011 interposer version "classe 10" at BNL (unpopulated)

09 June 2011 proposal for contract signed

14 June 2011 programming of ladder FPGA via debug card works.

20 June 2011 USB access to ladder FPGA via debug card works.

01 July 2011 JTAG access to ladder FPGA via debug card works.

09 August 2011 First analog measurement of ladder card using a "fake static card".

31 August 2011 QRDO version "1" at BNL (populated)

06 October 2011 Second MoU signed

12 October 2011 CD 2/3 passed

18 October 2011 First version of master FPGA pinout version "03" on the web^[0].

25 January 2012 contract signed

01 February 2012 Authorisation to start working on ladder card version "pre-production".

February 2012 One ladder (old electronics) at LBL for mechanical survey

end February 2012 RDO version "1" at BNL (populated)

June 2012 FPGADC version "v1_star" at BNL (populated)

June 2012 20 ladders (old electronics) at BNL

July 2012 final files for ladder card version "pre-production" due 6 months after contract signed

June 2013 20 ladders at BNL (assembled, tested and surveyed).

7 References

- [0] <http://www.subatech.in2p3.fr/~electro/projets/star/upgrade/index.html> STAR - SSD - UPGRADE files from SUBATECH, SUBATECH, IN2P3/CNRS-L'UNAM-EMN-Université, Nantes, F44307, France C.Renard - Version 2.0 - 02 février 2012
- [1] CY7B991V 3.3-V RoboClock® Low Voltage Programmable Skew Clock Buffer, CYPRESS <http://www.cypress.com> - Revision G - September 29, 2010
- [2] Trigger/Clock Distribution Tree, Trigger => Front-End Electronics Interface Specification, version 1.1 September 10, 1995 V.Lindenstruth & al.
- [3] BNL Requirements for STAR's Detectors about trigger busy signal, version ? - 2009
- [4] ALICE128C TECHNICAL REPORT ; LEPSI-IN2P3-CNRS/ULP Strasbourg, France; L. HEBRARD, J.P. BLONDE, C. COLLEDANI - DRAFT - September 2, 1997
- [5] Circuit de lecture des détecteurs à microstrips silicium d'ALICE; LEPSI-IN2P3-CNRS/ULP Strasbourg, France; J.D.BERST et al. - Journées VLSI IN2P3- 03 juin 1998
- [6] Alice Detector Data Link ALICE-DDL Hardware Guide for the Front-end designers Revision 2.1 Alice98/21 Internal Note/DAQ 16 May 2003 C.E.R.N., EP Division, AID Group
- [7] Alice Detector Data Link ALICE-DDL Interface Control Document Appendix of the User Requirement Document ALICE-INT-2004-018 version 1.0 - 06 July 2004 C.E.R.N., PH Department, AID Group
- [8] ALTERA <http://www.altera.com>
- [9] Cyclone III Device Handbook, ALTERA <http://www.altera.com> - version CIII5V1-3.3 - January 2010
- [10] Synplify_pro, SYNOPSIS-SYNPLICITY <http://www.synopsys.com/Tools/Implementation/FPGAImplementation/Pages/default.aspx> - version 2011.09 - August, 22th 2011
- [11] Quartus II, ALTERA <http://www.altera.com> - version 10.1-64-Bit Service Pack 1 SJ Full Version - January 19th 2011
- [12] ModelSim-Altera, MENTOR GRAPHICS CORPORATION <http://www.altera.com/mySupport> - Starter Edition 6.6d 2010.11 - November 2nd 2010
- [13] SN54ACT8990, SN74ACT8990 TEST-BUS CONTROLLERS IEEE STD 1149.1 (JTAG) TAP MASTERS WITH 16-BIT GENERIC HOST INTERFACES, TEXAS INSTRUMENTS <http://www.ti.com> - SCAS190E – JUNE 1990 – REVISED JANUARY 1997
- [14] DS90UR241/DS90UR124 5-43 MHz DC-Balanced 24-Bit LVDS serializer and deserializer, NATIONAL SEMICONDUCTOR <http://www.national.com> - Revision 201945 - January 8, 2008
- [15] AFB-57M5APZ 850 nm, SFP (Small Form Pluggable), RoHS Compliant, Low Voltage (3.3 V) Digital Diagnostic Optical Transceiver, AVAGO Technologies <http://www.avagotech.com> - Revision AV01-0276EN - August 8, 2007
- [16] DS90LV031A 3V LVDS Quad CMOS Differential Line Driver - NATIONAL SEMICONDUCTOR <http://www.national.com> - Revision DS100095 - July 1999
- [17] REF1004 1.2V micropower voltage reference - BURR-BROWN <http://www.ti.com> - SBVS002/PDS-1172 - October 1993 / May 7th, 2008
- [18] ADG704 CMOS Low Voltage 4Ω, 4-Channel Multiplexer - ANALOG DEVICES <http://www.analog.com> - Revision A - 1999
- [19] CMP04 Quad Low Power, Precision Comparator - ANALOG DEVICES <http://www.analog.com> - Revision D - 2003
- [20] AD7356 Differential Input, Dual, Simultaneous Sampling, 5 MSPS, 12-Bit, SAR ADC - ANALOG DEVICES <http://www.analog.com> - Revision 0 - October 2008
- [21] AD8029/AD8030/AD8040 Low Power, High Speed Rail-to-Rail Input/Output Amplifier - ANALOG DEVICES <http://www.analog.com> - Revision A - November 2003
- [22] LTC1662 Ultralow Power, Dual 10-Bit DAC in MSOP - LINEAR TECHNOLOGY <http://www.linear.com> - - 2000
- [23] CFPS-32IB: 2.5V low voltage, surface mount oscillator with a CMOS output - IQD/C-MAC Microtechnology <http://www.cmac.com> - Issue 2 - 28 July 2005

- [24] USB-Blaster Download Cable, ALTERA <http://www.altera.com> - version 2.5 - April 2009
- [25] POLYFUSE® Resettable PTCs 0603L Series Surface Mount > 0603L Series, LITTLEFUSE <http://www.littlefuse.com> - - December 2, 2009
- [26] copper-clad aluminum cable, MWS Wire industries <http://www.mwswire.com/cca.htm> - version -
- [27] http://www.star.bnl.gov/public/ssd/STAR_technique/costarv10.pdf; Manuel d'utilisation de CoSTAR; LEPSI-IN2P3-CNRS/ULP Strasbourg, France; J.D. Berst, G. Claus, C. Colledani - Version 1.0 - Décembre, 1999
- [28] AD5449 Dual 12-Bit, High Bandwidth, Multiplying DACs with Serial Interface- ANALOG DEVICES <http://www.analog.com> - Revision D - June 2011
- [29] REF3318, REF3312, REF3318, REF3320, REF3325, REF3330, REF3333, 3.9 μ A, SC70-3, 30ppm/°C Drift VOLTAGE REFERENCE, BURR-BROWN PRODUCTS FROM TEXAS INSTRUMENTS <http://www.ti.com> - SBOS392A–AUGUST 2007–REVISED SEPTEMBER 2007
- [30] A normalized USB interface based on the FT245B chip from FTDI - Laboratoire de l'Accélérateur Linéaire (LAL) http://electronique.lal.in2p3.fr/echanges/LALUsb/Firmware/Documentation/en/USB_23_en.pdf - Version 2.3 - October 19, 2007
- [31] http://www.star.bnl.gov/public/ssd/STAR_technique/ssdNumerotation.html; Lilian Martin, Jérôme Baudot - October 30th, 2007
- [32] AD9200 Complete 10-Bit, 20 MSPS, 80 mW CMOS A/D Converter- ANALOG DEVICES <http://www.analog.com> - Revision E - 1999
- [33] Proposal for a Silicon Strip Detector for STAR (SSD), STAR note 0400.
- [34] http://www.star.bnl.gov/public/ssd/STAR_technique/ssd_cooling_system_doc_2.pdf; Version 2 - June 27th, 2002
- [35] http://www-subatech.in2p3.fr/~electro/projets/star/star_readout/doc_sc.html STAR - SSD - CARTE READOUT, INTERFACE ENTRE CARTE READOUT ET SLOW-CONTROL, SUBATECH, IN2P3/CNRS-EMN-Université, Nantes, F44307, France C.Renard - Version 3.3 - 11 juin 2007
- [36] Measurements of pulse shapes at the SSD ladder card; Ch. Renard, S. Bouvier, M. LeVine, H. Matis - February 10th, 2010
- [37] http://www-subatech.in2p3.fr/~electro/projets/star/star_readout/simulation_itag_slave_fpga/index.html Test results for the shifting operations with internal flow control, BNL, Upton NY, USA, Micheal-J. LeVine - Version 1 - 15 January 2011
- [38] http://www-subatech.in2p3.fr/~electro/projets/star/star_readout/doc_upgrade.html STAR - SSD - CARTE READOUT, UPGRADE STAR SSD, SUBATECH, IN2P3/CNRS-EMN-Université, Nantes, F44307, France C.Renard - Version 1.f - 19 novembre 2008
- [100] http://www-subatech.in2p3.fr/~electro/projets/star/star_readout/index.html STAR - SSD - CARTE READOUT, SUBATECH, IN2P3/CNRS-EMN-Université, Nantes, F44307, France C.Renard - Version 3.3 - 09 janvier 2007
- [101] <http://www-subatech.in2p3.fr/~electro/projets/alice/dimuon/trigger/darc/index.html> ALICE - DIMUON-TRIGGER - DARC, SUBATECH, IN2P3/CNRS-EMN-Université, Nantes, F44307, France C.Renard - Version 6.4 - 17 juin 2008

Information furnished by authors is believed to be accurate and reliable. However, no responsibility is assumed for its use, nor for any infringements of patents or other rights of third parties that may result from its use. Specifications subject to change without notice. Trademarks and registered trademarks are the property of their respective owners