TESLA Report 2006-05
DESY Thesis 2006-000

WARSAW UNIVERSITY OF TECHNOLOGY
Faculty of Electronics and Informational Technologies
Institute of Electronic Systems
ELHEP Laboratory
DESY TESLA LLRF Team

Rafał Pietrasik

(B.Sc. Thesis in Electrical Engineering)

SIMCON 3.1
LLRF System control board measurements
(for TESLA Test Facility)

Thesis supervisors:

Wojciech Giergusiewicz, M.Sc, ELHEP WUT

Krzysztof T.Pozniak, Ph.D. , ELHEP, WUT

Stefan Simrock, Ph.D., DESY, Hamburg

Ryszard S.Romaniuk, Prof. D.Sc., ELHEP, WUT

Warsaw, Hamburg, March 2006

Contents:

1.Short introduction to the SIMCON devices family.

a) Purpose of the SIMCON devices **3**

b) Short history **4**

c) Characteristic of the SIMCON3.1 device **5**

2.SIMCON3.1 board description and schematics. 5

1.Analogue I/O **8**

2. Digital I/O **9**

3.Clock system **10**

4.Ethernet link **12**

5.Digital optical input **13**

6.Analogue optical input **13**

7. RS232 Interface **14**

8.VME section and booting FPGA devices. **14**

9. Virtex2Pro section **19**

a) Jumpers and switches connected with Virtex2Pro FPGA **19**

b) Schema of connection SRAM and DRAM to Virtex2Pro FPGA. **19**

b)Schema of connection temperature sensors with Virtex2Pro **21**

3. Measurements

a) Measurement of range of the input voltage and quantization level of analog to digital converters input. **22**

b) Measurements of the input offsets **24**

c)Measurement of the voltage Ganges with and without 50 [*Ohm*] load **25**

d)Offset and delays measurement of an analogue SIMCON3.1 path **27**

f) Measurement of SNR, SFDR, SINAD and effective number of bits (ENOB) **32**

g) Measurement of the noise parameters: **38**

Appendix. 1

Xilinx pins speciffications **41**

ACKNOWLEDGMENT

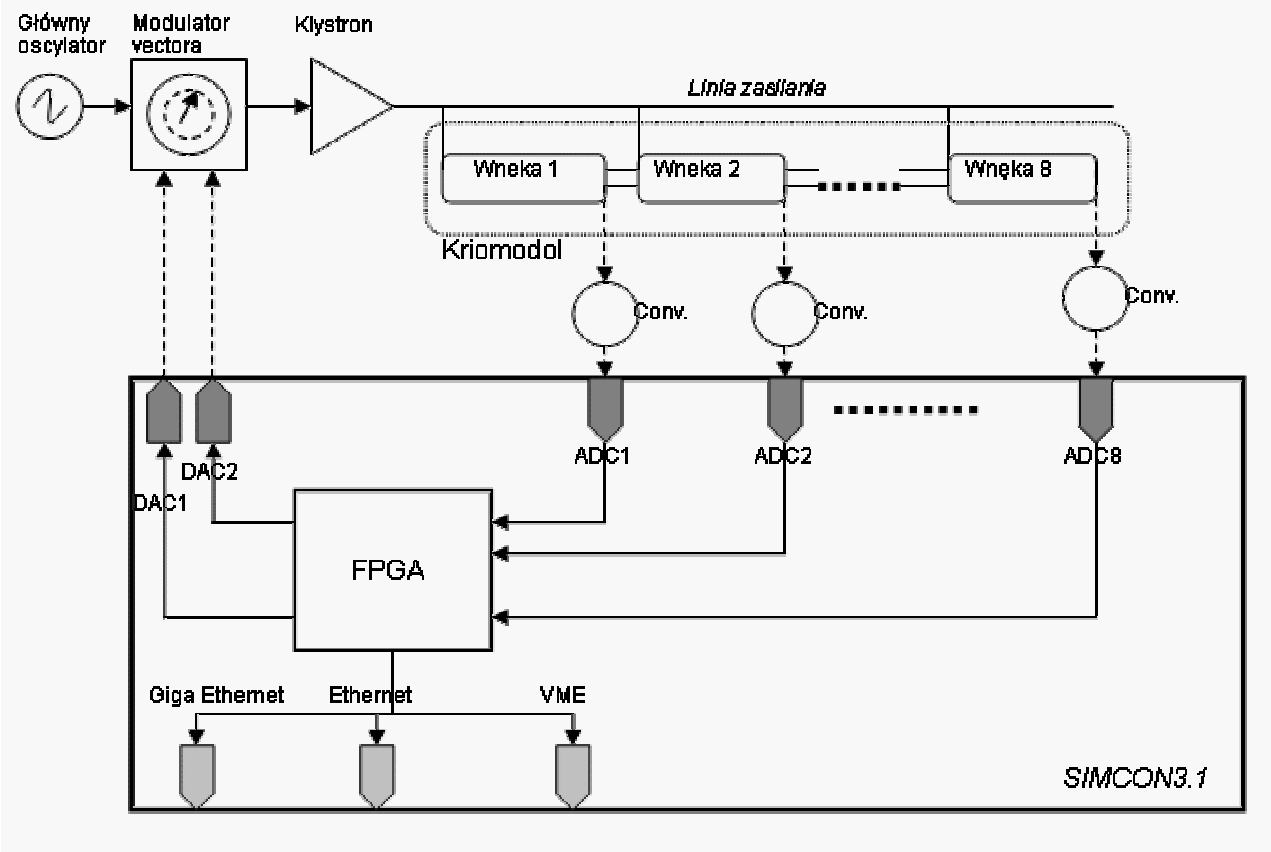
We acknowledge the support of the European Community Research Infrastructure Activity under the FP6 "Structuring the European Research Area" program (CARE, contract number RII3-CT-2003-506395).

1.Short introduction to the SIMCON devices family.

a) Purpose of the SIMCON devices

Family of SIMCON devices were developed with an international cooperation of Warsaw University of Technology (PERG/ELHEP group), Lodz University of Technology and Deutches Elektronen Sunchrotron (DESY). The devices are used in LLRF control system of VUV-FEL accelerator placed in DESY.

Accelerator consists of series of cry modules. Each cry module is build from 8 cavities and maintains ultra low temperatures to allow proper work of the cavities which are build from superconductive materials. The purpose of cavities is to maintain good quality of electron beam (ie. correction of path of beam) by electric field. The cavities are powered and controlled by the clystron device. This situation is shown on the diagram below. The diagram shows also position of the SIMCON device in the LLRF control system.



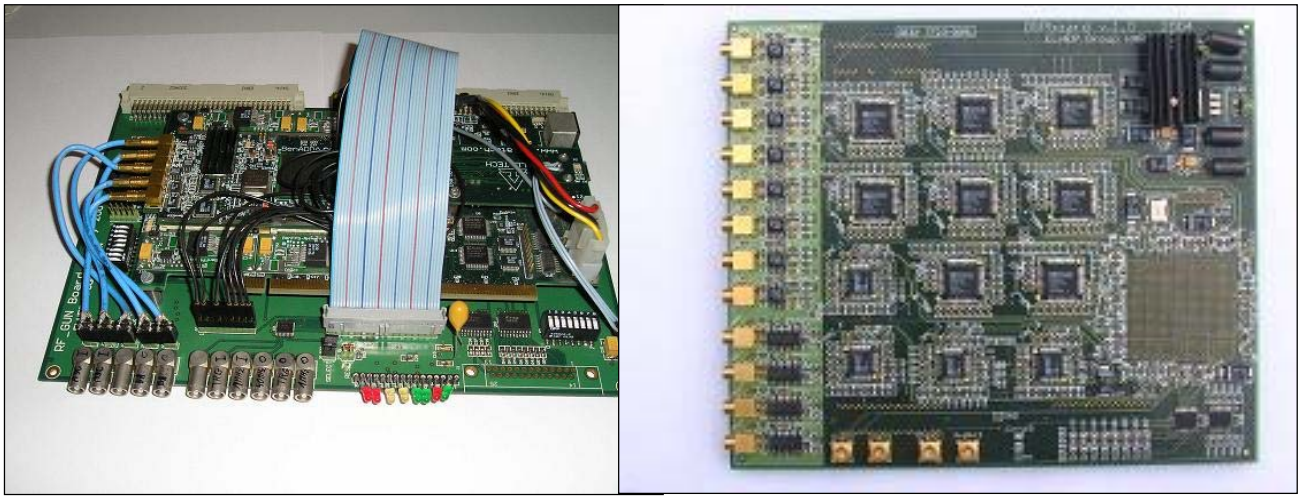
Diag.1
Schema of LLRF control system

LLRF control system is synchronized by the clock signal from master oscillator. Vector modulator controls work of the cystron. Real time measurements from the cavities are downconverted to the 250KHz frequency and carried to the inputs of the SIMCON device. In this device input signals are converters by 14-bits ADC's to the digital format. In FPGA on board is implemented complicated algorithm which calculates vectors from an input data. Those vectors are converted by 14-bits

DAC's and carried as analogue signals to vector modulator. The feedback loop came into begin. Controlling of work of SIMCON devices is done by VME, Ethernet or GigaEthernet interfaces.

b) Short history

SIMCON3.1 is the latest version of the device-the most sophisticated and equipped with newest technologies and features. Before this SIMCON there were developed SIMCON2.1 and SIMCON3.0. Those boards are shown on the pictures below.



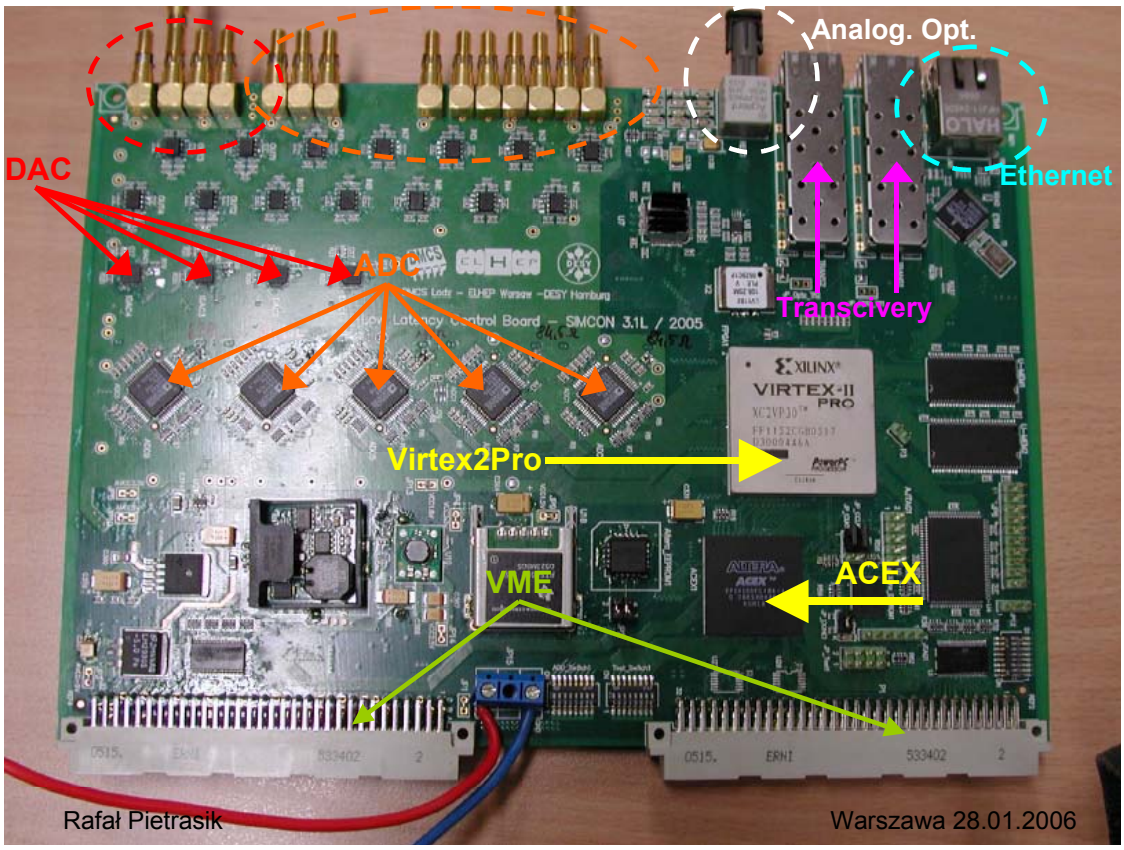
Pic.1

SIMCON2.1 and SIMCON3.0 boards.

The SIMCON2.1 board was build based on Virtex2 3000 FPGA and contained only one input channel. The SIMCON3.1 was equipped with better FPGA-Virtex2Pro and had 8 input channels and 4 output to maintain control of one single cry module (8 cavities). All these boards required a motherboard to work in the VME crate.

c) Characteristic of the SIMCON3.1 device

SIMCON3.1 was designed to work in the EURO crate (VME standard). One is an autonomic and independent device (no motherboards are required). It may work with or without WME controllers. It is however possible to power the board without VME crate by plugging the 5 [V] voltage directly to the proper socket on board.



Pic.3
SIMCON#.1 board with marked commponents

The „heart" of the SIMCON3.1 device is FPGA Virtex2Pro (XCE2VP30-FF1152). It has 30,816 logic cells, 136 multipliers with resolution of 18x18 bits, 2,448 Kbits inbuilt memory (BRAM) and 8 RocketIO transceivers with speed of 3.125 Gbps. To the input of the SIMCON3.1 there are connected only 2 transceivers. They allow to implement very fast GigaEthernets links with other devices.

The most important advantages of the Virtex2Pro device are two inbuilt PowerPC 405 cores. Those processors allow running normal software as Linux and controlling the board from level of this operating system. It is very flexible and comfortable solution cause it simplify handling the rest of the SIMCON's components such as Ethernets I/O or RS232 I/O.

On the SIMCON3.1 device there are two 64 Mbits DRAM's blocks dedicated to use with PowerPC processors. This memory blocks can be synchronized up to 133 MHz clock signal.

Next advantage of the Virtex2Pro is inbuilt PLLs which allows implementing multipliers, dividers and regulators of the clocks phase signals (DCM-Digital Clock Managers). They allow to achieve from the internal 50 MHz on board clock signal frequencies from 1,5 to 270 MHz. This signal can be used to synchronize work of the any of the SIMCON3.1 component.

On the SIMCON3.1 device there is one more FPGA-Altera ACEX (EP1K100FC484-1). It consists of 100,000 gates, 4.992 logic cells and has ability to work with max. Speed of 66 MHz.

Purpose of this FPGA is handling VME interface and Internal Interface (II). Altera ACEX is the bridge between those interfaces and Virtex2Pro.
To program all FPGAs n board JTAG interface is used.

SIMCON3.1 consists of series of the ADC's (10) and DAC's (4) with 14bits resolution. They digitalize input signals with speed up to 105 MSPS and converts digital signal to analogue with speed up to 210 MSPS. Data from the ADC's can be in U2 or natural binary code. All inputs and outputs from the device have 50 [Ohm] resistance and handle signals in range of (-1,+1) [V]. To maxmalize SNR the isolating input amplifiers were used.

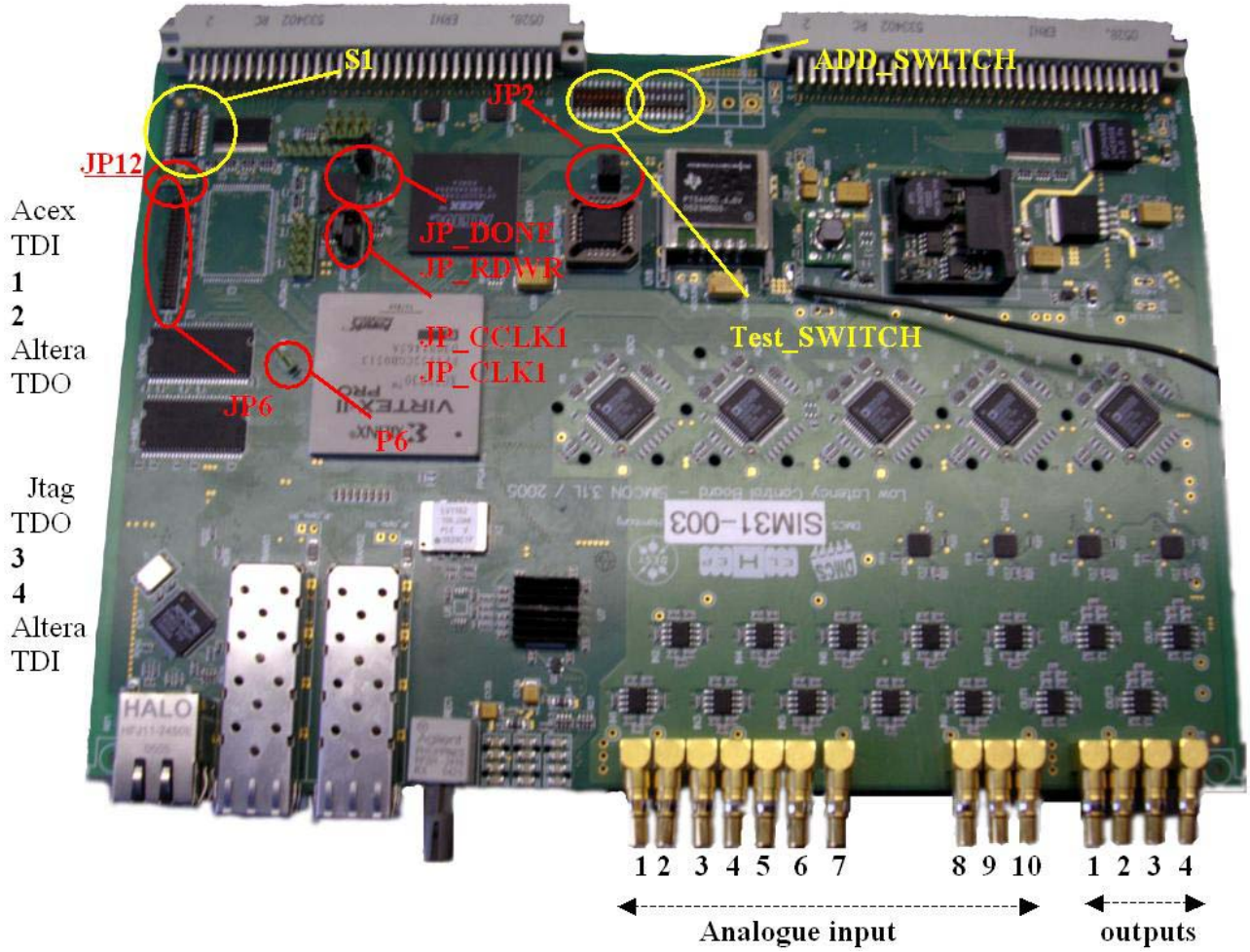
Besides mentioned optical transceivers, SIMCON3.1 is equipped with an analogue optical input. This input is designed for handling very fast optical clock signal. Signal from this input is carried to FPGA.

To communicate with other devices were developed Ethernet socket which allows to handle Fast Ethernet interface through the FPGA and RS232 socket (ie. Handling Linux terminal).

The SIMCON board is equipped with special, very precise input for an external clock signal. Those inputs handle symmetrical LVPECL standard which has better SNR than common non-symmetrical standards. The signal from the inputs can be used to synchronize the work of the ADC's and DAC's directly, without FPGA assist or with FPGA assist. It is big advantage, which allows to disable latencies in an external clock signal produced by the FPGA component.

Beside mentioned I/O, SIMCON3.1 is equipped also with digital input and output which are directly connected with Virtex2Pro (through appropriates buffers)

2.SIMCON3.1 board description and schematics.



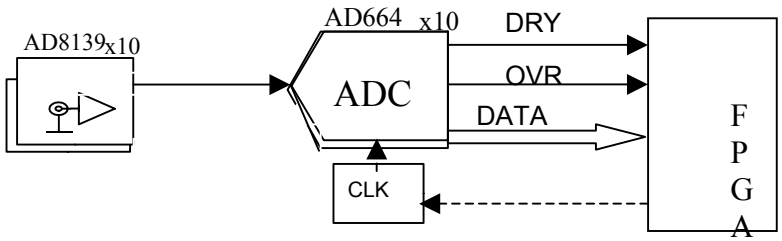
Rys.1

SIMCON3.1 – view from upper side

The picture above shows SIMCON3.1 with marked position of the important jumpers and switches which are described further in the text. Jumpers are marked by red and switches by yellow color. On the picture there are also marked an analogue inputs/outputs of the board;

1.Analogue I/O

The SIMCON3.1 board has analogue input and output path to and from FPGA. Both can be used independent. This paths consist of 10 MCX type sockets (STRAIGHT PCB) with 50 [Ohm] impedance which are used as analogue input and 4 socket of the same type used as analogue output. Analogue input path processes input signal to digital signal, which can be further used in FPGA. Recommended input signal range is from -1 [V] to +1 [V], maximum speed of sampling equals 105 MSPS and resolution is 14 bits. Schema of connection analogue inputs with FPGA is shown on pic no 2.



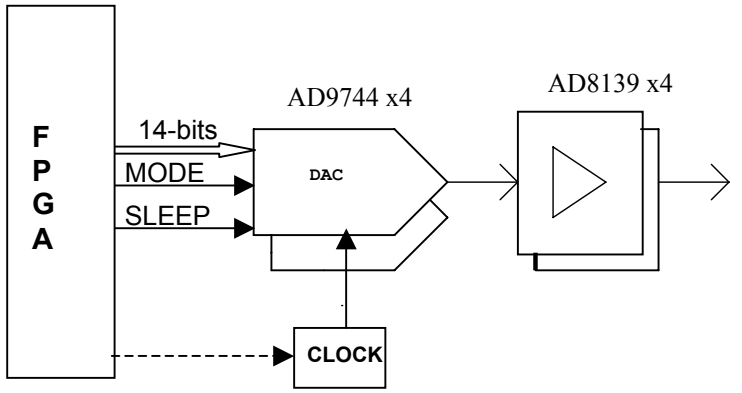
Rys. 2

ma

Asymmetrical input signal is converted to the symmetrical (by AD8139 amplifier), to reduce noises, and then it is carried on ADC input (component AD6645). This analogue converter can be synchronized with clock signal chosen by FPGA in range from 20 Mhz up to 105 Mhz.

Information about data ready to read on output data bus is done by DRY signal. Information about overflow of the input signal range is done by OVR signal. Those signals with an output data from ADC are connected directly with FPGA (Virtex2Pro).

Analogue output path allows to create an analogue signal from the digital data. Maximum resolution of this data is 14 bits and maximum speed of such conversion is 210 MSPS. Schema of the output path is presented below.



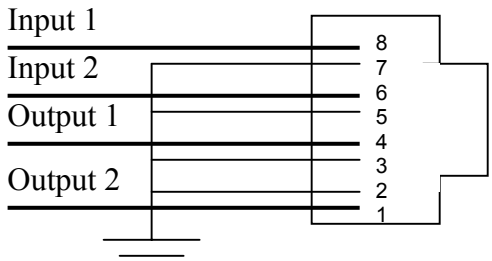
Rys.3
Analogue output path

Output signal, from the SIMCON3.1 board is from differential amplifiers (AD8139) in asymmetrical standard. Amplifiers converts symmetrical signal from 14 bits DACs (component AD9744). DACs are fully configurable by FPGA. They worked with frequency range 20 – 210 [Mhz].

The DACs can be configured to work in U2 or normal binary code. This configuration can be done by the DAC_MOD signal. FPGA can also control work of the DACs by the SLEEP signal.

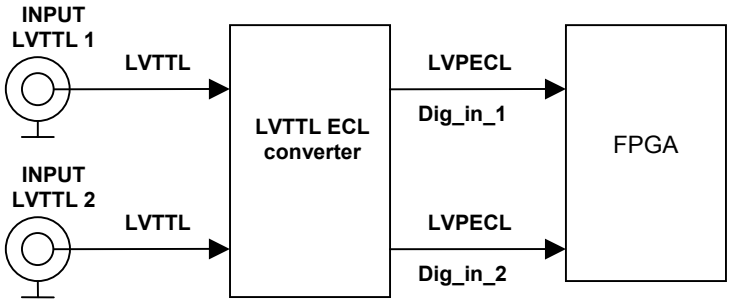
2. Digital I/O

On the SIMCON3.1 there is RJ-45 connector which is connected to digital I/O. Schema of this connection is on the pic below.



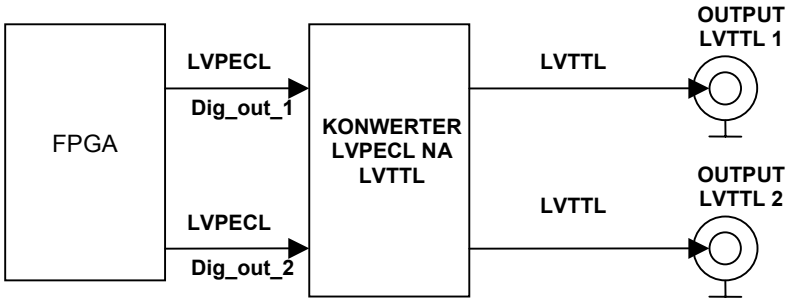
Rys.4
Schema of connection RJ-45 with digitall input/output

All 4 sockets pairs are in LVTTL standards and are connected to FPGA by appropriate converters.



Rys.5
Schema of digital inputs to FPG connection

Asymmetrical signal in LVTTL standard is converted (converter MC100EPT22) to symmetrical signal in LVPECL standard which is required by the Virtex2Pro. After conversion signal is transported directly to FPGA.



Rys.6
Digital outputs connection schema

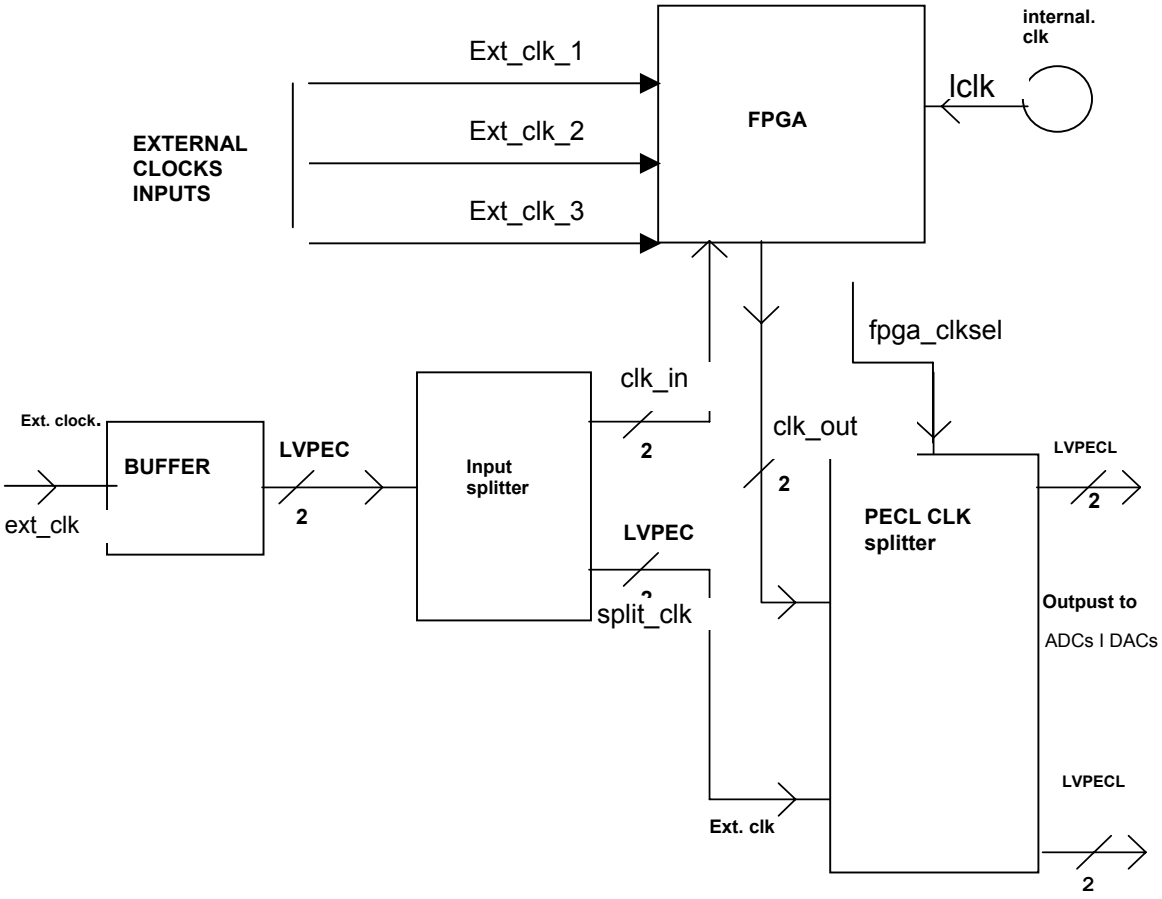
Symmetrical signal (LVPECL standard) from the FPGA is being converted (converter (NB100ELT23L) to asymmetrical signal (LVTTL standard) and then carried directly to the digital

inputs of the SIMCON3.1.

Digital I/Os are very universal. The allows to plug any signals in LVTTL standard to the FPGA. It is very useful ie. in debugging VHDL code. They could be also used as additional clock triggers, but for such triggers there are dedicated special inputs. Digital outputs could be used to synchronize external devices such as generators..

3.Clock system

The SIMCON3.1 board is equipped with very sophisticated clock distribution system. This system allows using an external and internal 50 Mhz clock signal. System is shown on the picture below.



Rys.7
Clock distribution schema

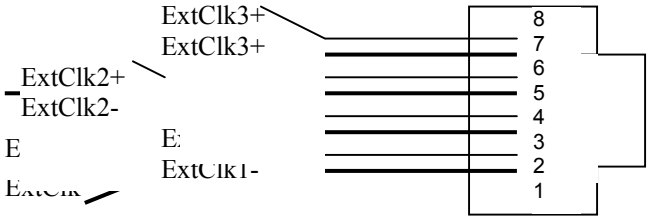
The source of the internal clock signal is inbuilt 50 Mhz resonator. Signal from this resonator is carried directly to the FPGA (lclk signal). The FPGA is equipped with DCM and allows to multiply or divide the internal clock frequency (in range 1,5 – 270 Mhz).

Next source of the clock signal is an external input (ext_clk). Signal from this input is carried by the fanout buffer to the splitter and then carried to the FPGA, ADCs and DACs. To avoid additional delays caused by the FPGA there is possible to synchronize analogue and digital

converters directly by an external clock signal (without FPGA in the path). This is done by the clock splitter and FPGA_CLKSEL(split_clk) signal from FPGA. The FPGA_CLKSEL signal chooses between an external clock and clock form the FPGA.

There are also three more clock inputs (EXT_CLK_1.. EXT_CLK_3). The signal from this inputs is carried directly to the FPGA. However it could be further distributed to all components like an internal clock.

Clocks inputs are available by the RJ-45 connector. Schema on the picture below shows method of the link of this signals.



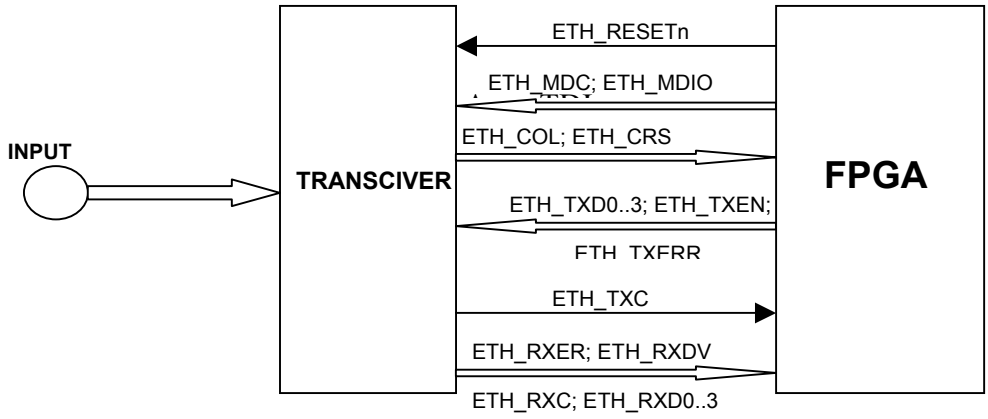
Rys.8

External clock input connection schema

Inputs of an external clock signals supports symmetrical LVPECL standard. All, but without ExtCLk, are connected with FPGA through MC100EP11 buffers. Average propagation time for this components equals 220 [ps].

4.Ethernet link

An Ethernet link is available by standard web socket HF11-2450E (common LAN socket).The Ethernet path consist of the BCM5221KPT one channel transceiver which supports physical net layer. The other layers have to be implemented in the FPGA. Running Linux system on the PowerPC processors enables an easy way to handle Ethernet connections by inbuilt system drivers (VHDL code is not needed in this case).



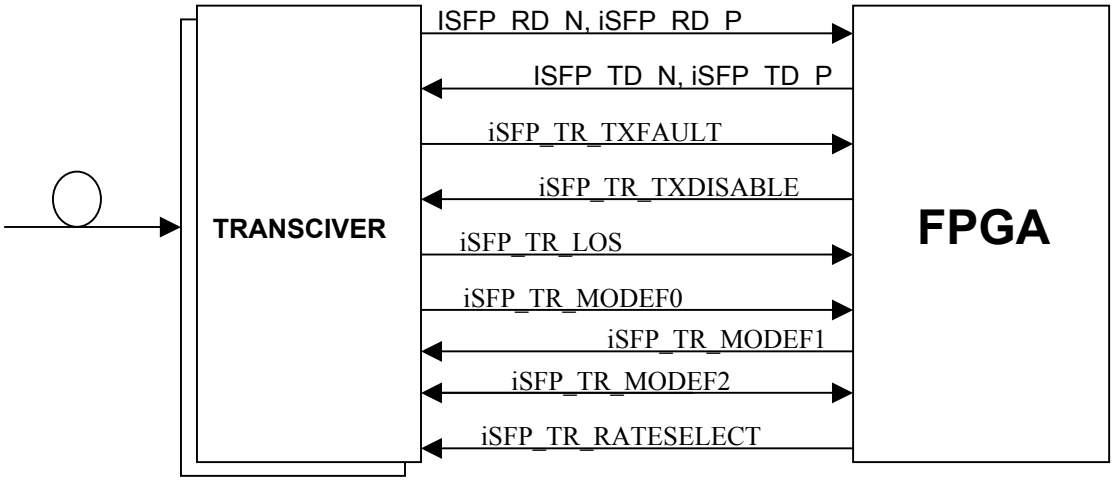
Rys.9 Schema of connection ethernet with FPGA

Schema above illustrates method of link between components in Ethernet path on the SIMCON3.1 board. All shown signals are enough to handle Ethernet standard.

5.Digital optical input

Virtex2Pro has 8 optical RocketIO transceivers dedicated to handle GigaEthernet links. On the SIMCON3.1 board used are only two of them. The link with the transceiver is possible by the pair cases of the V23838-S5-N1 type.

Maximum frequency of the digital optical link is 3.125 Gbps for the channel.



Rys.10
Optical transcivers connection. Schema

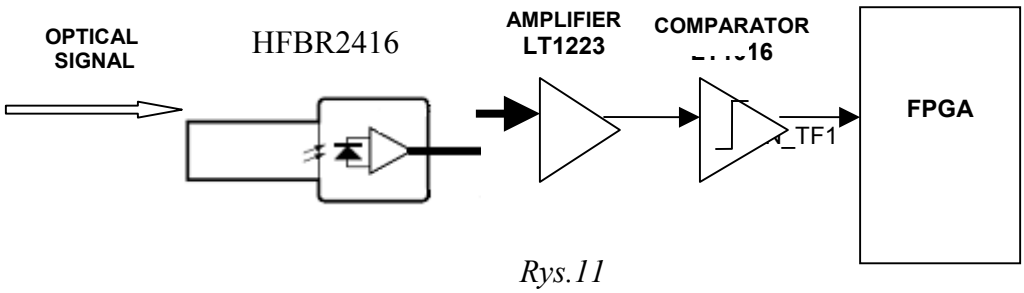
The FPGA handles all signals shown on the pictures above. Each shown signal has an index (because there are two transceivers) ie. iSFP_TR_LOS1 is for the first transceiver and iSFP_TR_LOS2 is for the second transceiver.

iSFP_TR_RATESELECT signal can be handling only when jumper JP_OPTO1 is crossed. (jumper is described further in text)

6.Analogue optical input

SIMCON3.1 allows to receive analogue optical signal by HFBR2416 receiver. This receiver handles following sizes of the fibers: 50/125um, 62,5/125um, 100/140um. 200Um HCS.

Maximum frequency of an input optical signal equals 175MBd and maximum frequency of an electrical signal from receiver is 125 Mhz. The picture below shows the way of the link in the optical input.



Rys.11
Analogue optical input connection-schema

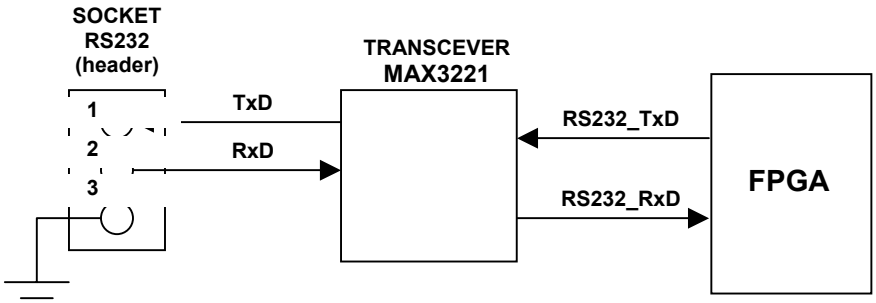
Analogue optical signal is converted to the analogue electrical signal in the receiver. Next it is carried on the amplifier input and then to the comparator. Comparator gives zero-ones signal (digital) to the FPGA (IN_TF1 signal).

An analogue input can be used as another source of the very fast external clock signal.

7. RS232 Interface

RS232 interface provide communication in this standard with use of MAX3221 transceiver (powered by 3.3[V]).Idea of RS232 interface is shown below.

z

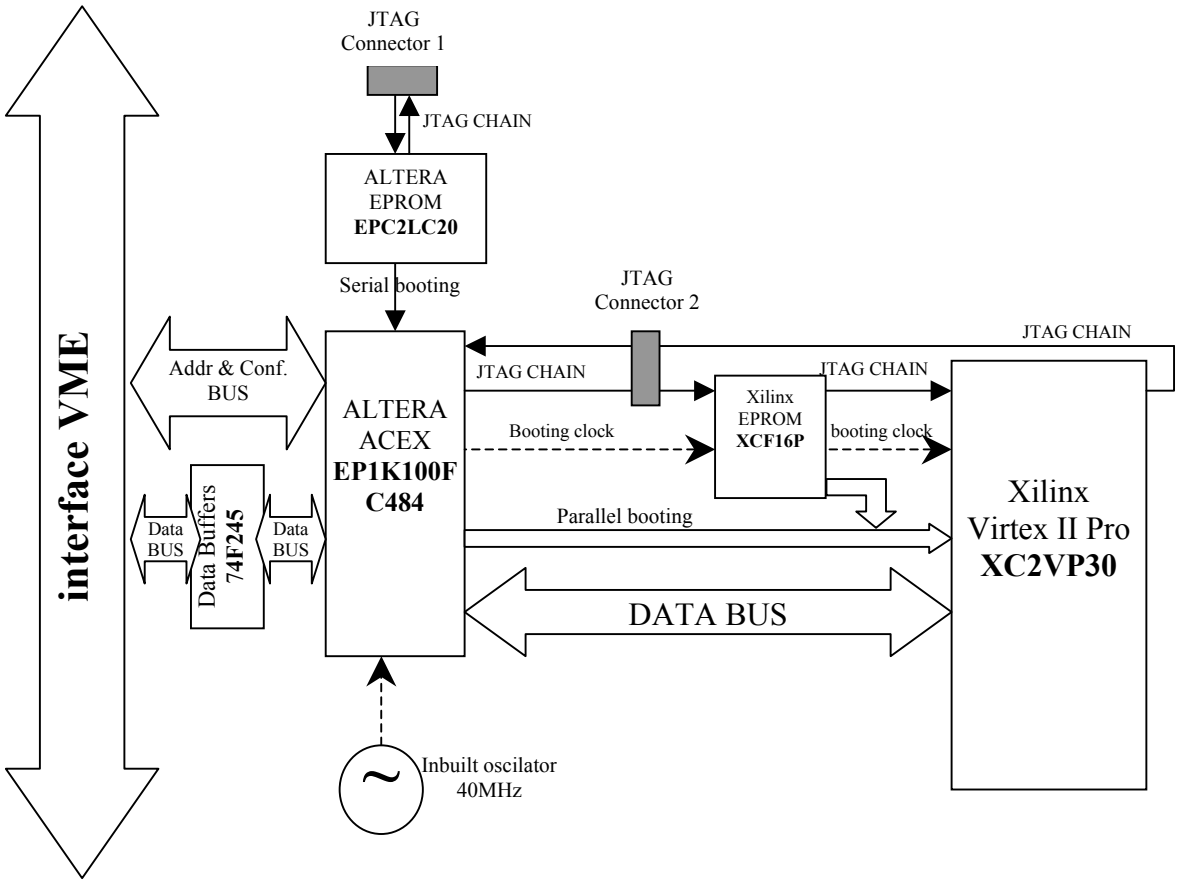


Rys.12
RS232connection schema

Socket which allows to connect RS232 cable is 3 pins header – JP12. One of the pins is connected to the ground, the others lead out TxD and RxD signals from/to the transceiver. This component carried the signal next to the FPGA (RS232_TxD, RS232_RxD signals)

8.VME section and booting FPGA devices.

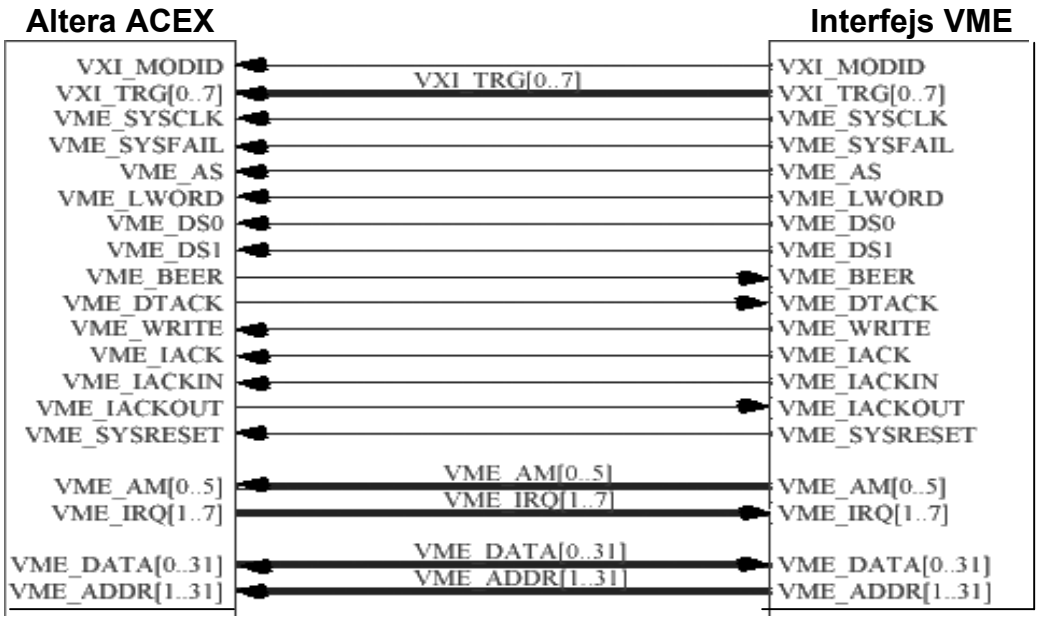
The SIMCON3.1 board is dedicated to work in EURO VME crate. For this reason there the board is equipped with very sophisticated section to handle communication in this standard. The VME section is linked with booting section. These sections are shown on the schema below.



Rys.13

VME sectionof SIMCON3.1 board

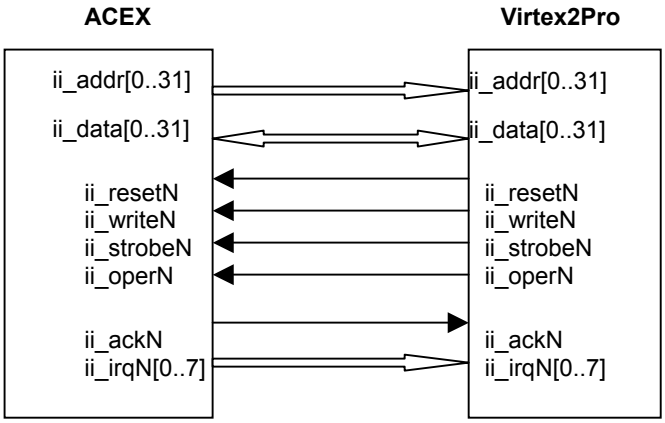
As it can be seen, communication with the VME interface is available by the Altera ACEX (EP1K100FC484). This FPGA with proper program is the bridge between VME interface and a Virtex2Pro device. The way of connection between VME and the Altera Acex is shown on the pic 14.



Rys.14

ACEX connection with VME interface

ACEX device besides data and adress bus handling, handles also all signals needed to proper, bidirectional communication with VME. VME interface requires that all device in crate to have a unique address. SIMCON3.1 board allows to change its address by ADD_Switch switch (picture 14 described further in text)

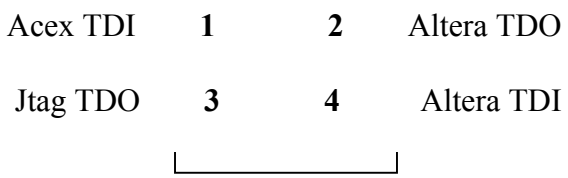


Rys.15
Connection between ACEX and Virtex2Pro

Since ACEX is the bridge between Virtex2Pro and VME, there are series of lines between these two components. This lines are Internall Interface lines shown on the picture 12. Internal Interface allows to read and to write data from address space in FPGA during its work and to send this data to PC with use of proper controller.

ACEX device can be started directly by the programator connected to AJTAG socket or from previously programmed ALTERA EPROM (EPC2LC2) memory. During its work it uses clock signal with frequency 40 [Mhz] from inbuilt oscillator.

Chose of the way of programming is done by proper set of jumper JP2 shown on the pic.13.



Rys.16
JP-2 jumper connection schema

Connecting pins 3 with 4 and 1 with 2 in JP2 turns on AJTAG header. Without such pins connections AJTAG is inactive and ACEX can be booted only from EPROM (crossing pins 2 and 4).

Second FPGA, Virtex2Pro can be also programmed in few ways.

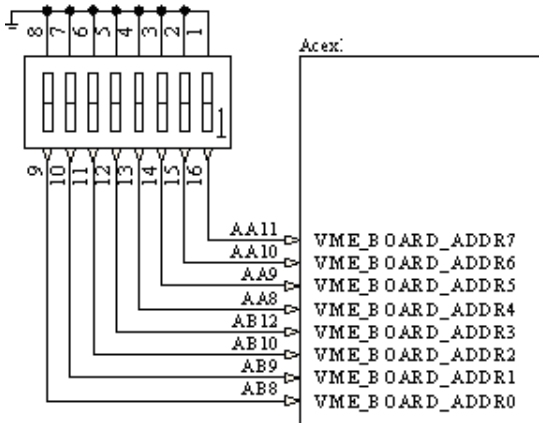
First way is programming directly by programator through XJTAG1 header. This header allows

also to program Xilinx EPROM (XCF16P).

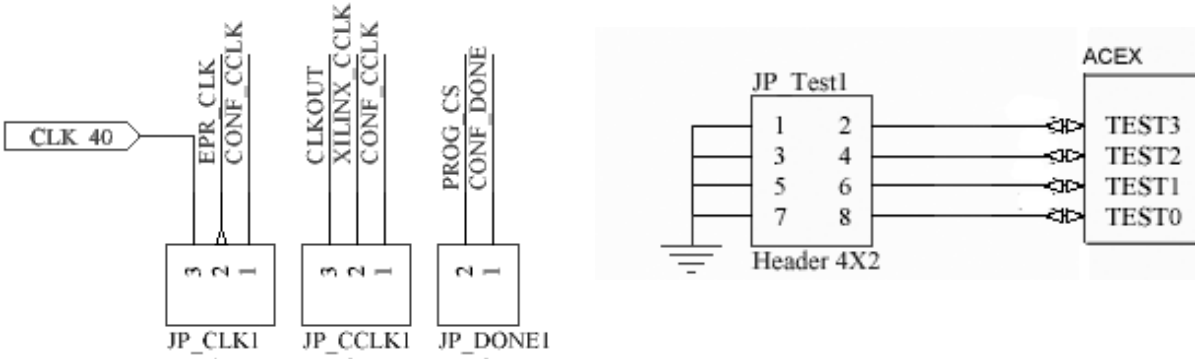
The second way is to boot from previous programmed Xilinx EPROM. This memory can by write by JTAG programator or across Altera ACEX device.

The last method to program Virtex2Pro is using VME interface across ACEX.

To synchronize work of devices in VME section 40 Mhz clock signal is used. There is however possibility to synchronize Xilinx EPROM and Virtex2Pro by clock signal generated from Altera ACEX.



Rys.17
ADD_Switch connection with ACEX - electrical schema



Rys.18
configuration jumpers fromVME section

In SIMCON's3.1VME section there are many jumpers and switches. Some of them are shown on Pic.18. Method of settings all these elements is precisely describe in the table below.

Name and elements type	Settings	Description
JP2 – 4 pins jumper 2x2	Crossing pins 3 with 4 and 1 with 2	Turn on JTAG socket (AJTAG1). It is then possible to program across this socket Aletra ACEX and Altera EPROM.
	Crossing pins 2 with 4	Turns off JTAG sockets (AJTAG1). Altera device is then booted only by EPROM.
JP_Test1 – 8 pins jumper 4x2	Any-depends of program in ACEX	Test jumper-used with VHDL programming and debugging. Connected to Altera ACEX. Its use depends of the program in FPGA. The jumper is shown on pic.13
Test_Switch1 – 8 elements switch	Any-depends from ACEX program	Another test switch connected to Altera ACEX. Switch no.1 corresponds to signal SWITCH7 in ACEX and so on till switch 8 which corresponds to SWITCH0 signal.
JPRDWR – 3 pins jumper 1x3	Crossing pins 2 with 1	Connects conf_rdwr signal from Virtex2Pro to GND. The FPGA can be then booted from Xilinx EPROM.
	Crossing 2 with 3	Connects signals conf_rdwr from Virtex2Pro with conf_write from Altera ACEX. This allows to boot Virtex2Pro across Altera ACEX or JTAG socket.
JP-DONE - 2 pins jumper 1x1	Crossing pins 1 with 2	Connects signals PROG_CS and CONF_DONE in Altera ACEX. This FPGA doesn't waits then for Virtex2Pro to program.
	Span pins 1 and 2	Altera ACEX waits for Virtex2Pro to program.
JP_CLK1 - 3 pins jumper 1x3	Crossing pins 3 with 2	This jumpers decides, which clocks will be use to synchronize Xilinx EPROM. With such setting it will be clock signal from inbuilt oscillator. Jumper is shown on pic.13
	Crossing pins 1 with 2	Clock signal for Xilinx EPROM will be signal (conf_clk) from Altera ACEX.
JP_CCLK1 – 3 pins jumper 1x3	Crossing pins 3 with 2	Jumper decides which clock signal will be use to synchronize Virtex2Pro during booting. Such setting sets signal CLKOUT from Xilinx EPROM as clock.
	Crossing pins 1 with 2	Clock signal for Virtex2Pro will be conf_cclk signal from Altera ACEX.
ADD_Switch – 8 elements switch	Any combination	This switch sets 8bits SIMCON3.1 address. Switch no.1 corresponds to signal VME_BOARD_ADDR_7 and than in sequence till switch no.8 which corresponds to VME_BOARD_ADDR_0.Switch is shown on pic.12
DONE_LED-diode	-	Diode is connected to signals conf_done from Altera ACEX and lights when FPGA is programmed.

Table 1

VME and booting jumpers description

9. Virtex2Pro section

Besides described components and switches connected with Virtex2Pro on the SIMCON3.1 board, there are some additional elements linked directly and only with Virtex2Pro. There are SRAM, DRAM, temperature sensors, diodes and configuration switches and jumpers. These last ones are described in table below.

a) Jumpers and switches connected with Virtex2Pro FPGA

Name and type	Settings	Description
P6 - 2 pins jumper	Unbound pins 1 and 2	Jumpers are used to an external reset of Virtex2Pro. When is unbound the FPGA works normally.
	Crossing pins 1 z 2	Resets the Virtex2Pro.
S1 - 8 elements switch	Test switches 4, 5, 6,7,8 any combination	Switches are connected to the test signals Virtex2Pro in the way below: switch 4 – in_test1 switch 5 – in_test2 switch 6 – in_test3 switch 7 – in_test4 switch 8 – in_test5
	Switches 1,2,3	Switches are connected to the signals below: switch 1 – M1 switch 2 – M0 switch 3 – M2 Switches above decided of the way of Virtex2Pro booting. There are 3 modes: Slave/master/SelectIMAP/Boundary (more in Virtex2Pro datasheet)
JP6 – 40 elements header 2x20	Pins from 1 to 40	Pins are connected in order to signals in_tes0 to in_test39 of the Virtex2Pro.
	Pins 1,2,19,20	Connected to GND
LED0..LED7 - diodes	-	Diodes connected in order to signals LED<0>..LED<7> in Virtex2Pro.

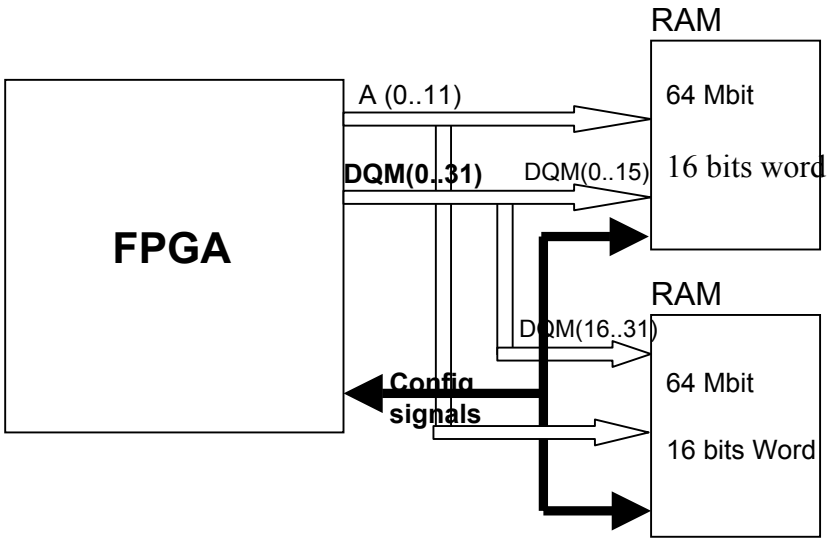
Table 2

jumpers and switches connected with Virtex2Pro

b) Schema of connection SRAM and DRAM to Virtex2Pro FPGA.

SIMCON3.1 has 2 RAMs MT48LC4M16A2TG-7E. This memory has size of 64MBits and is divided into four 1024 Mbits banks (with maximum word length equal 16 bits). The memory

handles also 4 and 8 bits words. Maximum memory speed is 133 Mhz with access time 5.4 ns. Memory is dedicated to work with PowerPC processors and besides stream/burst reading/writing it allows read and write separated bits. One RAM has maximum word length equals16 bits but on SIMCON3.1 board memories are connected in the way which allows to achieve 32 bits words. It is shown on schema below:



Rys.19

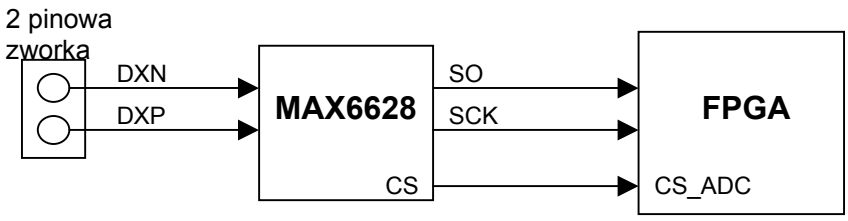
VME configuration jumpers

As it can be seen on pic.16 address bus for both memories is common and consist of 12 signals A(0..11). Data bus is different for each memory. 16 most important bits DQM(31..16) is connected to one memory, and others bits DQM(15..0) is connected to the other memory. The choice of bank, clock frequency and another config signals are common to both memories. From FPGA point of view, SIMCON3.1 has one memory with maximum word width equal 32 bits and size of 128 Mbits.

Second sort of memory handled by Virtex2Pro on SIMCON's3.1 board is very fast SRAM. This memory is dedicated to acquisition not very large amount of data and can be used as ie. Very fast data buffer. Its size is 512Kbits with word length equal 36 bits. Similar to DRAMs there are 2 SRAMs blocks. From the FPGA point of view it is memory with size of 1Mbits and 19 bits address bus (VME_A(0_19) and 36 data bus (VMEDQ(0..35)).

c)Schema of connection temperature sensors with Virtex2Pro

The last elements from SIMCON's3.1 is temperature sensor. It is MAX6628 device. The diagram below shows the way of connection.



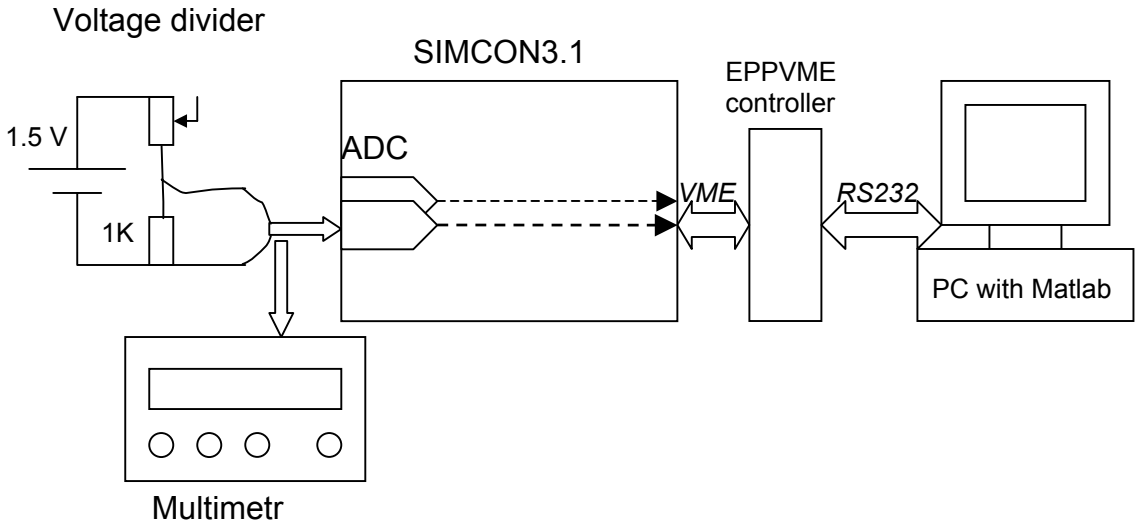
Rys.20
Temperature sensor connection schema

An output to connect termopairs is two pins jumper, which is connected with MAX6628 pins. This device carries signals to FPGA.

3. Measurements

a) Measurement of range of the input voltage and quantization level of analog to digital converters input.

Schema of the measurement configuration.



Rys.1

Schema of configuration to measure input voltage ranges

Constant voltage needed to measurement was taken from 1.5 [V] battery instead of the DC power supply generator to avoid additional noise and disturbances from the 50 MHz 230 Volts grid. Regulation of level of this voltage was implemented by the voltage divider constructed from 1 [Kohm] resistor and linear potentiometer. Such solution allows fully change the voltage in range from -1.5 to 1.5 [V].

Signal from the divider was carried to the input of the SIMCON board and to millimeter. Data from ADCs were taken through Internal Interface to the PC computer with MatLab.

Measurement of the input voltage range was done by changing the level of the input voltage and effects observation of these changes in the output data from ADCs

ADCs have an output resolution equal 14 bits so their output data is in range from 0 to 16384. Graph was plotted from ADCs data. If the graph was near end of range of the ADCs (near 1 or near 16384) its maenad that input voltage reached its maximum or minimum. The value of voltage could be read from the multimeter.

List of devices used to the measurement:

Multimeter: *HP 3440A*

Devices settings:

Multimeter was set for DC measurement In range 10.000000 [V]. SIMCON3.1's ADC's were working with 50 [MHz] frequency.

Results of the measurements:

Value of the input voltage was read directly from the Multimeter. Estimation of quantization level was possible after measurement of the input voltages. The sum of minimum and maximum input voltage divided by 2^14 (16384) is searched quantization level (ql). This ql defines a micro volts needed to change from one output code to next one in ADC outputs. The results are shown below.

Input number	**Lower range**		**Higher range**		**Quantization level**		
	Value *[V]*	**Error *[V]***	**Value *[V]***	**Error *[V]***	***Value [uV]***	**Error**	
						[uV]	***[%]***
ADC 1	-1,1589	0,0047	1,2621	0,0049	147,766	0,586	0,4
ADC 2	-1,1632	0,0047	1,2583	0,0049	147,797	0,586	0,4
ADC 3	-1,1723	0,0047	1,2718	0,0049	149,179	0,586	0,4
ADC 4	-1,1934	0,0048	1,2701	0,0049	150,361	0,592	0,4
ADC 5	-1,1831	0,0047	1,2431	0,0049	148,092	0,586	0,4
ADC 6	-1,1821	0,0048	1,2632	0,0049	149,133	0,592	0,4
ADC 7	-1,1252	0,0046	1,2613	0,0049	145,621	0,580	0,4
ADC 8	-1,1351	0,0047	1,2445	0,0049	145,212	0,586	0,4
ADC 9	-1,1783	0,0047	1,2415	0,0048	147,693	0,580	0,4
ADC 10	-1,1543	0,0047	1,2432	0,0048	146,332	0,580	0,4

Table 1

Results of the input range measurements

Average input range equals (-1,16459;1,25578) [*V*]. It is 165 [*mV*] more from the lower side and 255 [*mV*] from the upper side than range needed to proper work of the SIMCON3.1 board. Also, it can be seen that input voltage range is shifted by the 50 [mV].
This shift is fully acceptable and can be reduced in FPGA.

Estimation of measurement error:

A measured error occurred during reading a graph plotted from an output data from ADCs and during reading a multimedia measurements.

The line which defines level of voltage on the graph had some thickness. It was the result of

overlapping the noise on the input signal. Thickness of this line was maximum 20 output codes from ADC. With assuming (with high reserve) that input range of each ADC is -1,2 [V] to 1,3 [V] quantization level can be estimated. This quantization level multiplied by the thick of the graph line gives a maximum error.

$$|-1,2|\ [V] + 1.3\ [V] = 2.5\ Vpp \Rightarrow 1\ [Code] = 2^{14}/2.5\ [V] = 16384 / 2,5\ [V] = 0,15\ [mV]$$

$$\delta max < 20*0,15\ [mV] = 3\ [mV]$$

Error caused by the thickness of the graph line: $\delta max = 3\ [mV]$

Multimeter error on 10 [V] range: $\delta m = 0,0015\%$

Summary maximum error: $\delta z = 0,0015\% + 3\ [mV]$

Quantization level was estimated from the measured input voltage level of each ADC. It has an error calculated from the sum of input voltage errors divided by 2^14.

Quantization level error: $\delta pk = (\delta zmin + \delta zmax)/16384$

b) Measurements of the input offsets

Measurement configuration schema:

Measurements configuration schema was the same as in the input voltage ranges measurements (last subsection).

The measurements of the offset was done by regulation of an input voltage and observation output data from ADCs till this data succeeded average level of 8192. Than a multimeter readings were searched offset level.

Results of the offset measurements:

Results of the offset measurements were shown in the table below. Column “Value in code” shows a average value of the output code read from ADC. Precisely set this level to 8192 was not possible due to potentiometer granularity.

Input	Offset [*mV*]	Value in code [*bit*]	Error [*uV*]
ADC1	51,632	8191	148,4294
ADC2	48,101	8200	1186,99
ADC3	49,125	8194	299,456
ADC4	53,705	8197	754,684
ADC5	51,856	8186	892,1458
ADC6	50,292	8185	1048,15
ADC7	50,231	8201	1315,73
ADC8	54,005	8188	583,273
ADC9	48,082	8190	296,6181
ADC10	50,733	8202	1469,04

Table 2

Results of offsets measurements

Table 2 shows that an average offset is equal 50 [mV] and it is the same for all the inputs. This situation allows using all inputs for the same purposes-they have identical offset and input ranges parameters.

Error estimation methods:

Multimeter error on 10 [V] range: $\delta m = 0{,}0015\%$

Error of the average output code level from ADC was calculated from the difference of set level and an average level 8192 multiplied by the quantization level. Also this error includes quantization error.

δoff=(8192 value in codes)*(QuantizationLevel+ δpk)

δoff_{ADC1}= (8192-8191)*(147,766+0,586)=148,352 [*uV*]

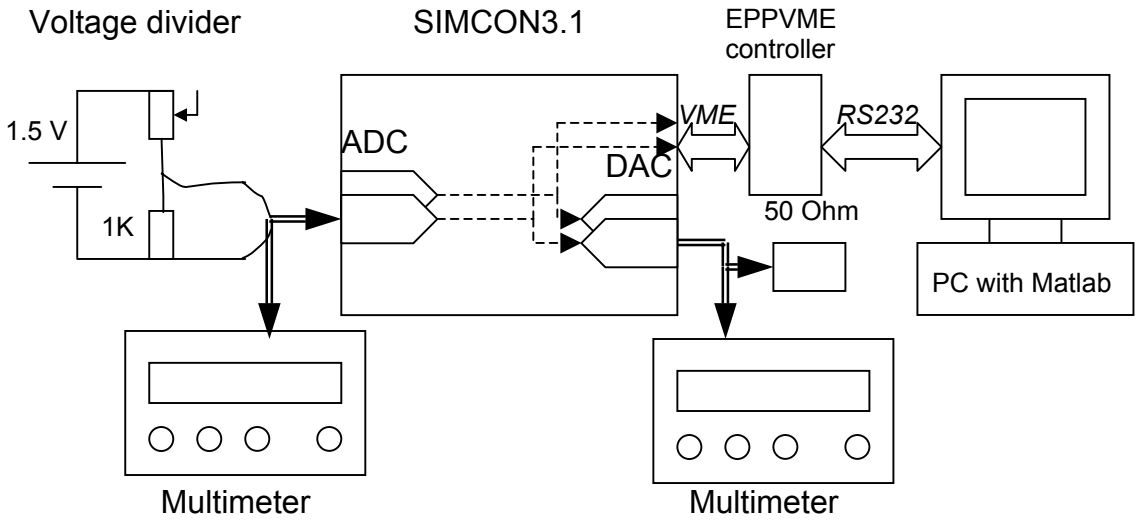
Total, maximum error is the sum of above errors and it equals:

$\delta offset = \delta off + 0{,}0015 * Offset$

Where Offset is measured voltage level read from multimeter.

c)Measurement of the voltage Ganges with and without 50 [*Ohm*] load

Measurement configuration schema:



Rys.3

Schema of measurement configuration

Schema from pic.3 is similar to the schema from a) subsection. The difference is in plugging a load to an analogue output of the SIMCON3.1 board. The measurements were done by multimeteres. Set of maximum and minimum value of the output voltage was done by FPGA by setting the minimum and maximum input codes.

Shown schema was also used to take a input/output characteristic with 50 [Ohm] loaded output.

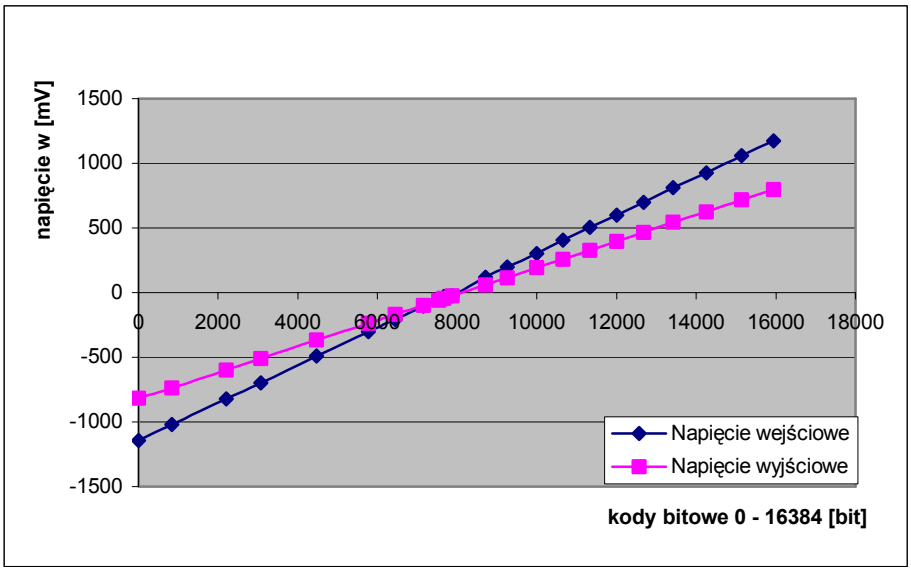
Results of the measurements:

Output	*Measurements without any load*		**Measurements with 50 [*Ohm*] load**	
	Min. Voltage [V]	**Max. Voltage [V]**	**Min. Voltage [V]**	**Max Voltage [V]**
DAC1	-1,6766	1,6542	-0,8371	0,8282
DAC2	-1,6777	1,6521	-0,8365	0,8301
DAC3	-1,6600	1,6621	-0,8401	0,8312
DAC4	-1,6790	1,6586	-0,8387	0,8321

Tabela.3

Results of output voltage measurements

Output voltages from the SIMCON3.1 board depend from the load and are linearly depended from an input voltage. It is required to the proper work of the board in LLRF control system. The graph below shows this situation.



Rys.4
ADC9 to DAC3characteristic

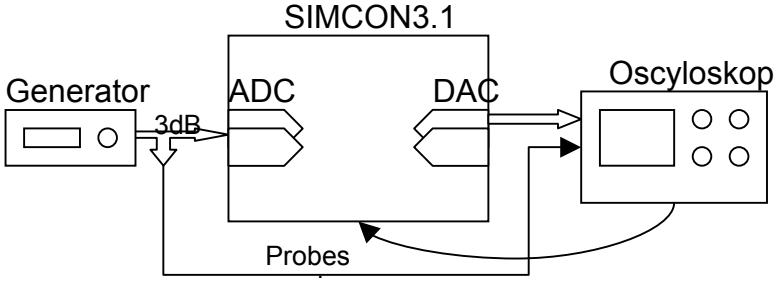
On the graph some amplifying can be seen. It also depends of the output load and should be included.

The way of an error estimation

Multimeter error on 10 [V] and 1 [V] range: $\delta m=0,0015\%$

d)Offset and delays measurement of an analogue SIMCON3.1 path

Measurement configuration:

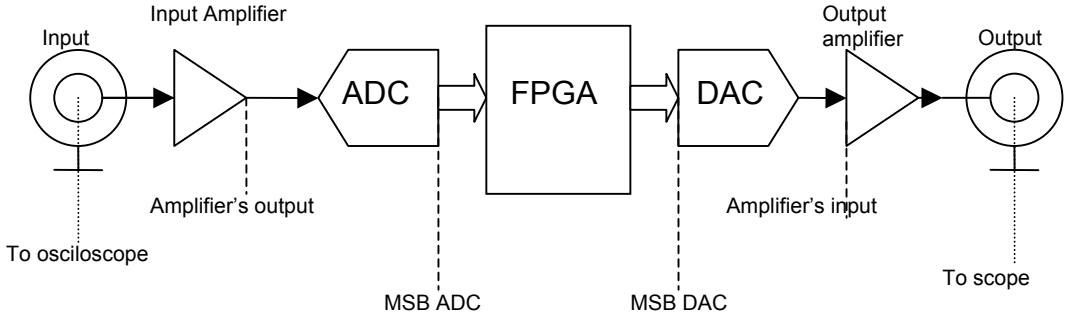


Rys.5

Schema of time parameters measure

Picture pic.5 shows schema of the measurement configuration. Square signal generator gives a signal to the ADCs inputs. This signal had a small frequency to eliminate error connected with to fast signal switching. Signal switch had to be much slower than summary delay. Amplitude of the input signal was nearly equal to the full scale input range of ADCs.

Analogue signals from DACs were observed on the oscilloscope. Probes were also used to observe a signals directly on the measurements points on the board. This points are shown on the picture below.



Rys.6

Measure points in delay measurements

Output from an amplifier its out pin OUTp and OUTn from this component. MSB ADC it's the Most Significant Bit of an output from ADC and input to DAC. Both, ADC and DAC were working in U2 system so MSB is changed when signal goes through zero level (changes its mark from minus to plus).

Devices used in measurement:

Generator: *function generator TG120 firm Thurlby Standard Instruments (Tti)*

Oscilloscope: *WaveRunner 6100 firmy LeCroy 5Gs/s do 1GHz*

Probes: *LeCroy PP007-WR 500MHz 10MOhm 9,5pF*

(for WaveRunner series)

Configuration of the devices:

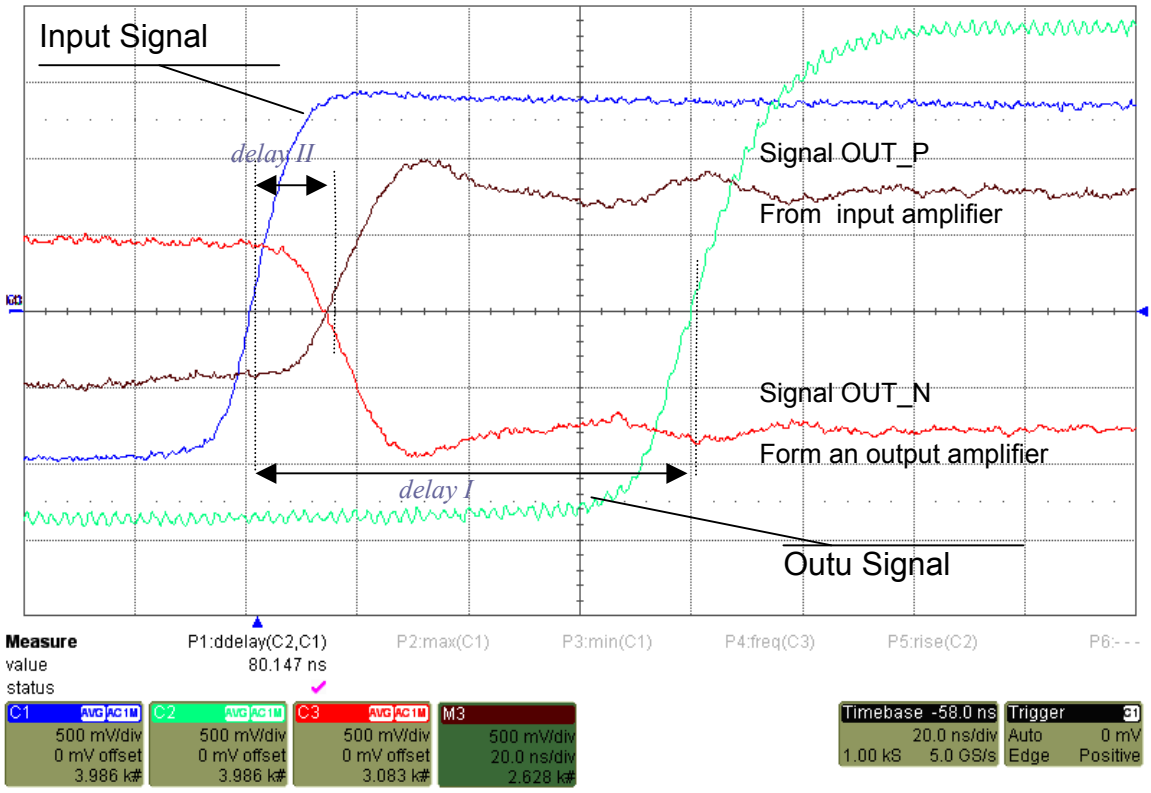
Function generator generates square signal with frequency about 250 [kHz] and with an amplitude 2 [Vpp] in range -1;1 [V]. Mean value was set to zero.

Oscilloscope worked in 5Gs/s mode with four input active. In each channel averaging in 10 periods was set. Triggering was set on rising edge of the SIMCON3.1 signal. All inputs worked in AC mode to eliminate offsets. Time delay was set to 20 [ns] and input signal range was 500 [mV/div].

ADC and DAC worked with 105 [MHz] speed.

Measurement results:

Pictures below are screen shots from oscilloscope. Only few were shown to explain the measurements.

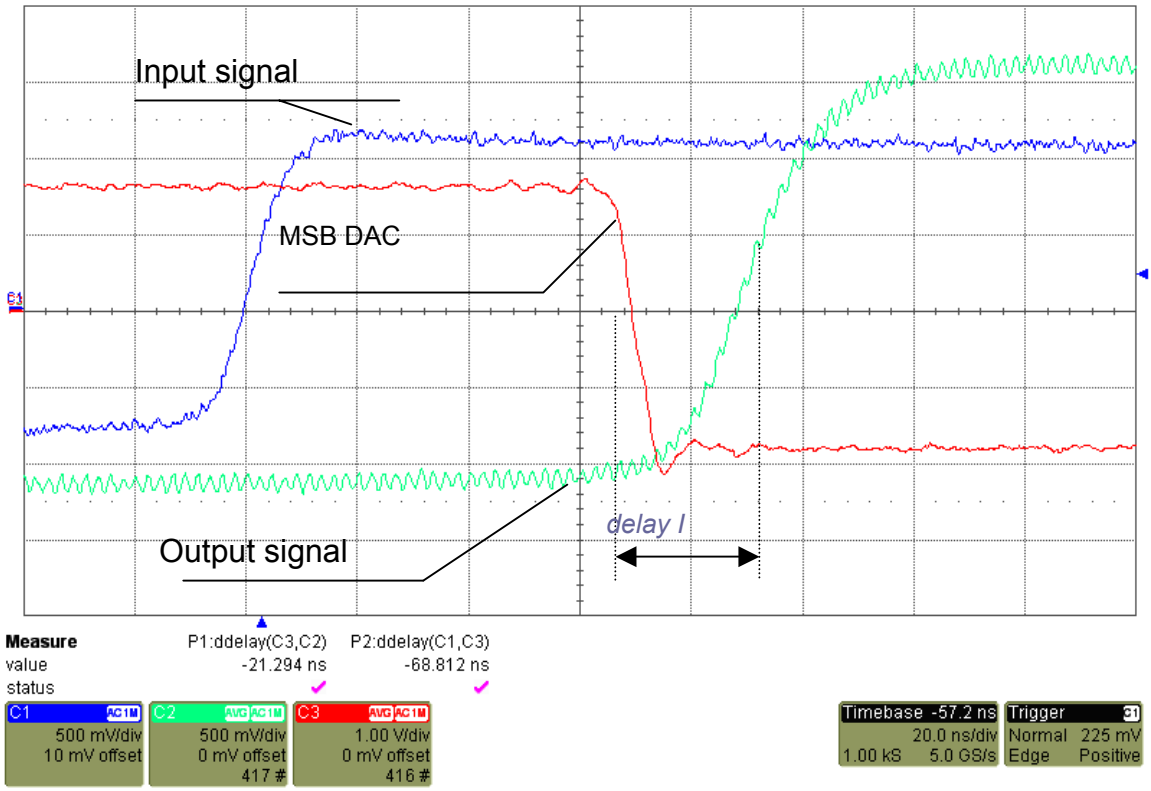


Rys.7

Screen from the scope - ADC-DAC measure

Picture Pic7 illustrates the way of measure delay between input and output of the SIMCON3.1 board (delay I) and delay caused by an input amplifier (delay II).

Cause signal form an amplifier output is symmetrical there are two output signals from this component. One is positive (OUT_P) and one are negative (OUT_N). As it can be seen on the picture both signals have the same amplitude and are mirrored to each other. Delays between measured signals were read in 50% of its rising/falling times. On pic7 it can be seen that delay between input-output is equal about 80,147 [ns], and input amplifier latency is equal 13.6 [ns]



Rys.8

ADC-DAC measure example (screen from the osciloscope)

Picture pic.8 shows measurement of delay on DAC and on output amplifier. This is the "delay I" between "MSB DAC" and output signal (green color).

Signal marked "MSB DAC" is most significant bit of DAC input data. It's switch from 1 to 0 shows corresponding point of sign change of an input signal.

The total delay on the picture equals abort 89.169 [ns][and latency of DAC and it's output amplifier is 21.2 [ns].

Result of delay measurements on each component in analogue path in SIMCON3.1 board are listed in tables below:

Input	**Latency of input amplifier [*ns*]**	**ADC delay**	**Summary input delay (ADC and amplifier)**
ADC 1	12,4	37,3	49,7
ADC 2	12,6	38,1	50,7
ADC 3	12,8	37,2	50
ADC 4	12,4	36,2	48,6
ADC 5	12,5	37,2	49,7

ADC 6	13,0	37,1	50,1
ADC 7	12,6	34,7	47,3
ADC 8	13,2	35,1	48,3
ADC 9	12,5	35,8	48,3
ADC 10	12,8	36,3	49,1
Average	*12,68*	*36,5*	*49,18*

Table 4

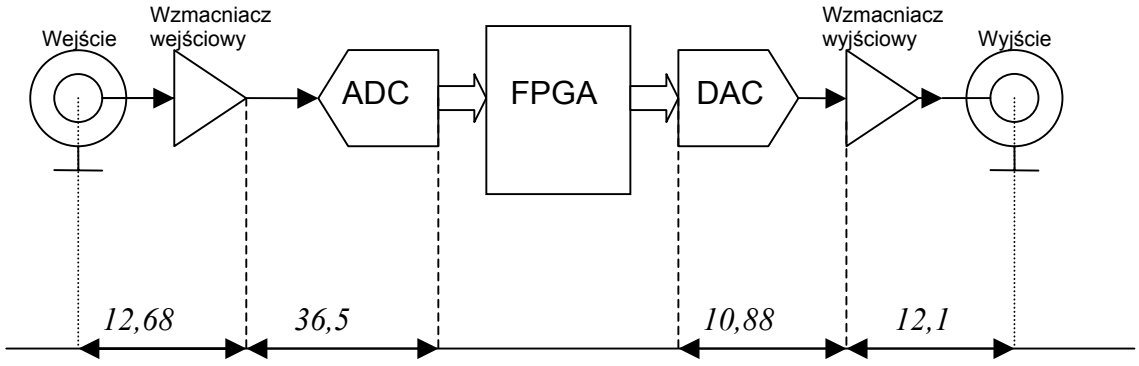
Measure results of ADC latancies

Output number	**DAC delay**	**Output amplifier latency**	**Summary delay (DAC and amplifier)**
DAC 1	10,8	12,3	22,1
DAC 2	10,3	13,1	25,4
DAC 3	11,3	12,0	23,3
DAC 4	11,1	11,0	22,1
Average	*10,88*	*12,1*	*23,23*

Table 5

Measure results of DAC latancies

Summary results of delay measurements on each component in analogue path in SIMCON3.1 board are shown on picture below. The results are average values for all inputs and outputs.



Rys.9

Measurement results presentation

As it can be seen, average delays are very small and much lower than in SIMCON3.0 board. Delays are nearly equal to corresponding data in datasheets.

Total delay is enough to use SIMCON3.1 board in LLRF system.

Estimation of a measurement error:

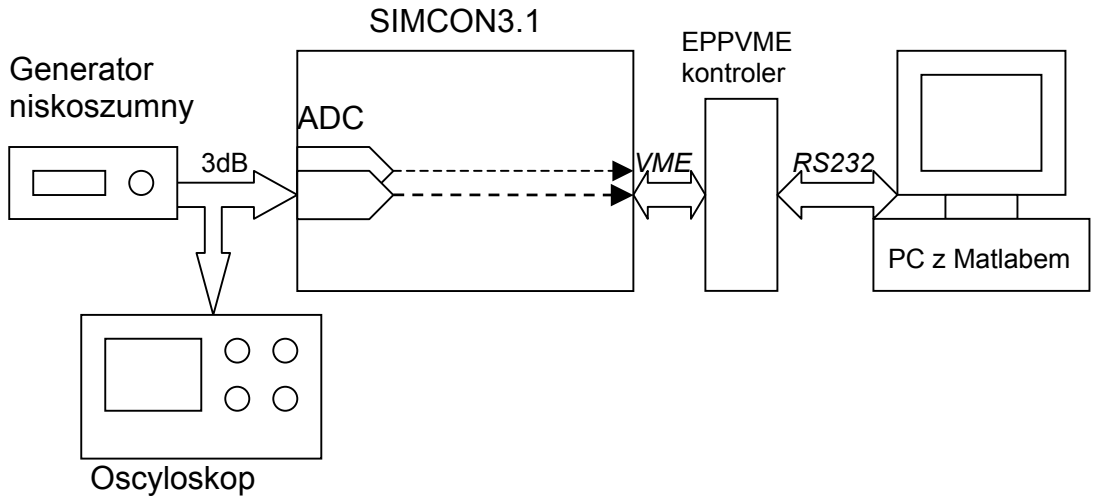
Error of readings from oscilloscope: $\delta t1=0,1*$ div $= 0,1*20$ [ns] = 2 [ns]

Oscilloscope resolution error: $\delta t2=200$ [ps] = 0,2 [ns]

Summary error: $\delta t1+\delta t2=2$ [ns] + 0,2 [ns] = 2,2 [ns]

f) Measurement of SNR, SFDR, SINAD and effective number of bits (ENOB)

Measurement configuration schema:



Rys.10

Measure configuration to estimate ENOB

Picture Pic.10 shows schema of measurement configuration. Sine wave signal was generated by low noise generator and taken to the SIMOCN3.11 inputs. Oscilloscope allows to control frequency and amplitude of the signal. Data from ADCs were taken through Internal Interface to PC with Matlab.

Devices used to measurement:

LN Generator: *Marconi instruments 10k-2.7GHz Low Noise Signal generator 2041*

Osciloscope: *WaveRunner 6100 firmy LeCroy 10Gs/s do 1GHz*

Software: *Matlab ver 6.5*

Devices configuration:

Generator worked in "Low noise Mode I" with "Low intermodulation" mode. The frequency of signal was set on 1MHz and 1,5MHz and it's power was set to +10,7 [dB] (RF Level) which was equal nearly full scale of SIMCON3.1 input.

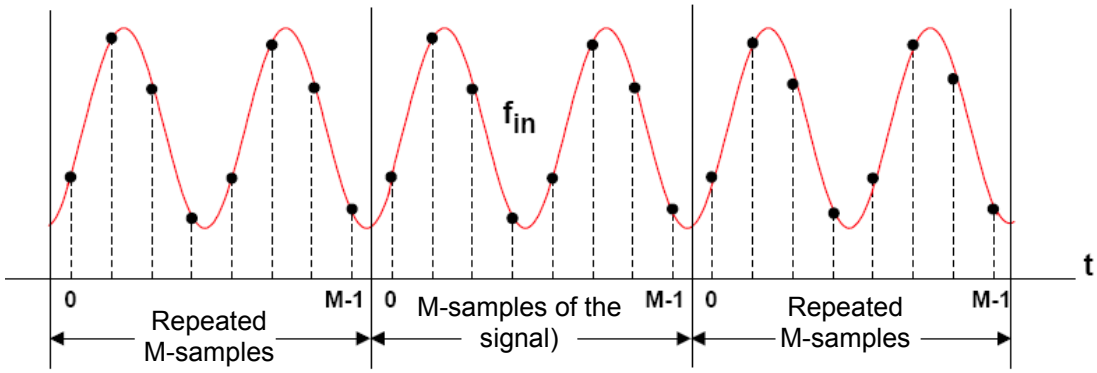
All kind of modulation were off. Generated signal was "pure" sine wave.

Cause oscilloscope was used only to control signal parameter, it's settings won't be described.

Measurements results:

With use of properly programmed FPGA and II the 8192 signal samples were taken for each input. This samples were collected with input signal freq. equal 1.5 [MHz].

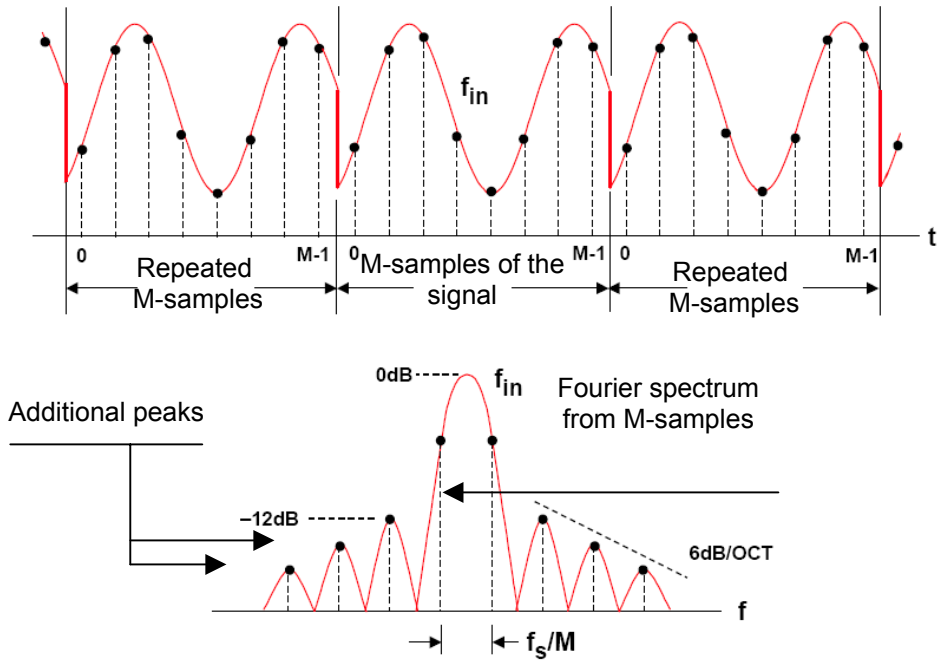
After acquisition, data had to be specially processed. To calculate ENOB the FFT algorithm was used. This algorithm estimates shape of the signal from M-samples at it has been shown below. M-samples are repeated in time domain.



Rys.11

Explanation of the FFT algorithm

Fourier spectrum of signal above is ideal sine wave signal spectrum without any distortion cause M-samples have N-whole signal periods. When M-sample doesn't have N-whole signal periods, the distortion occurs in FFT spectrum. It is shown on the picture below.



Rys.12

Explanation of FFT errors

It can be seen that repeated M-samples don't have N-whole signal periods. This produces disturbances in the spectrum-spectrum leakage.

First additional peak is only 12 [dB] below main signal peak and the next one are falling 6 [dB/oct]. This situation makes impossible to estimate SNR, ENOB etc. parameters from a spectrum.

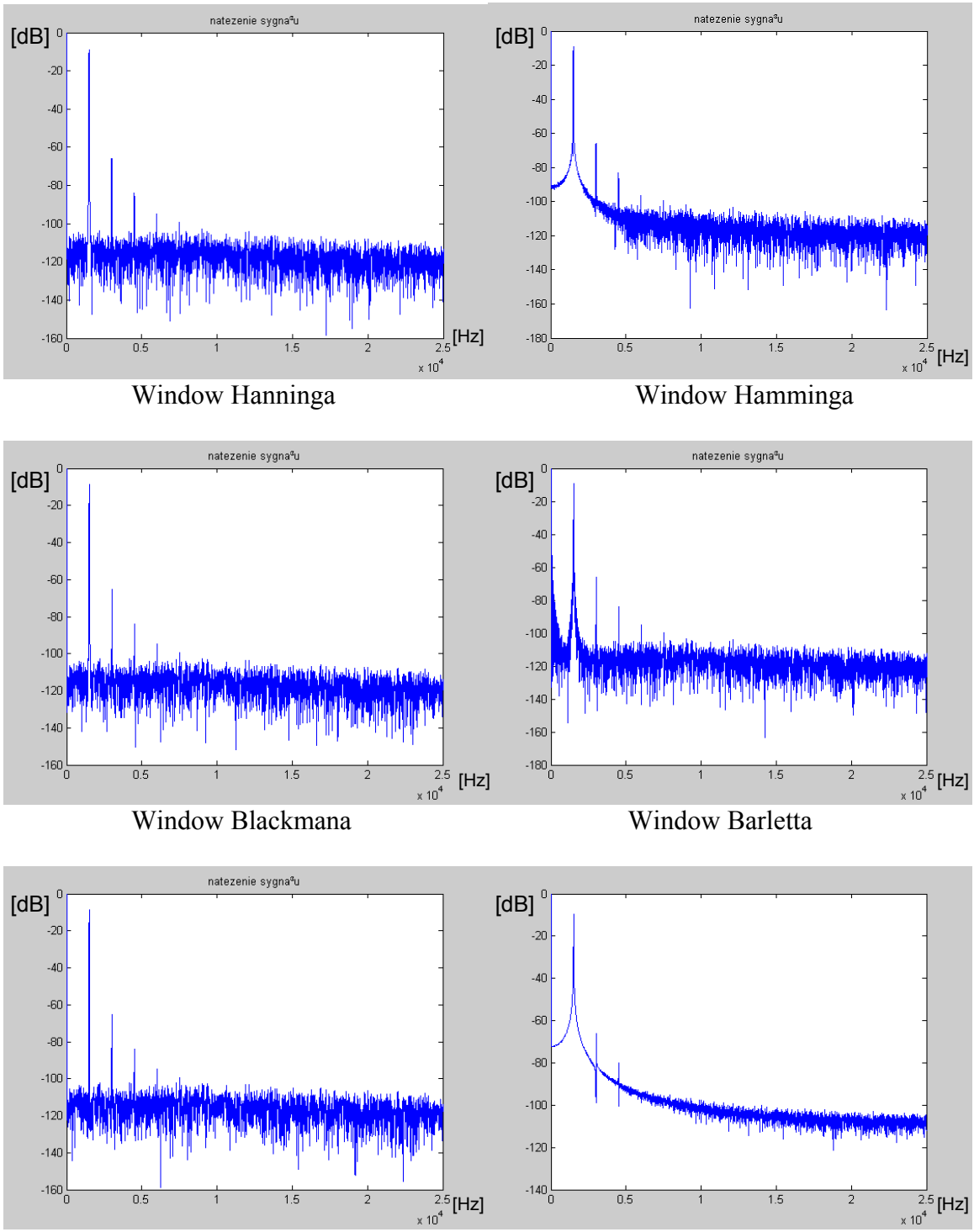
However there is a way to improve a spectrum using a special sampling window instead of square window. The window was chosen by comparison of effect and parameters of popular windows. This windows are introduced in table below.

Window name	**Formula:** **n-sample** **M-number of samples**	**Level of first additional pick [dBc]**	**Fall on octave [dBc]**
square	N	-12	6
Hanninga	0,5-0,5COS(2*pi*n/M)	-32	18
Blackmana	0,42-0,5COS(2*pi*n/M)+0,08COS(4*pi*n/M)	-58	18
Hamminga	0,54-0,46COS[2*pi*n/M]	-43	6
Blackmana-Harrisa	0,35875-0,48829COS(2*pi*n/M)+0,14128COS(4*pi*n/M)-0,01168COS(6*pi*n/M)	-92	6

Tabela.5

Windows parameters

The lowest level of additional to the main peak was in Blackam-Harris window (-92 [dB]). However it wasn't main criteria of window chose. In selection it was also important to the window to have a very fast falling level of additional peaks. So the Blackman window was selected. His window Has low level of additional peak and very Fast falling level. The results of testing all window are shown on the graphs below.



Window Hanninga | Window Hamminga

Window Blackmana | Window Barletta

Window Blackmana-Harrisa | Window prostokątne

Rys.13
Examples of windows results

On a graph sets Pic.12 it can be observed that the worst results gives pure square window and the

best results gives Blackman, Harris and Blackam-Harris windows.

The spectrum graphs were created from specially rescaled ADC's data.

Rescaling in time domain was based on the fact that spectrum is from 0 to fs/2 where fs is sampling frequency (in this case 50 [MHz]).

Rescaling the data to dB was done by this formula:

$$dB=10\log_{10}[Wartość^2/WartośćMax^2]=20\log_{10}[Value/ValueMax]$$

where:

dB – rescaled value used to plot the graph

Value – Value of the n-th sample of the signal

ValueMax – Maximum value from M (8192) samples

To calculate ENOB it is required to estimate SNR, NF, SINAD, THD.

NF-Noise Floor was estimated from the signal spectrum without spectrum leakage around main signal peak and harmonics peaks. From the NF taken from the spectrum the PG was removed.

PG – Processing Gain is an FFT effect which lowers the real NF.

NF is lowered by a value which can be calculated from formula below:

$$PG=10*\log_{10}(M/2)$$

where:

PG – Processing Gain, value of FFT lowering

M – total number of samples

SNR – Signal to Noise Ratio in [dBc] is related to the main signal peak on Fourier spectrum. It is the difference between NF (reduced by PG) and the level of the main signal peak.

The other values :SINAD and THD where calculated from formulas below:

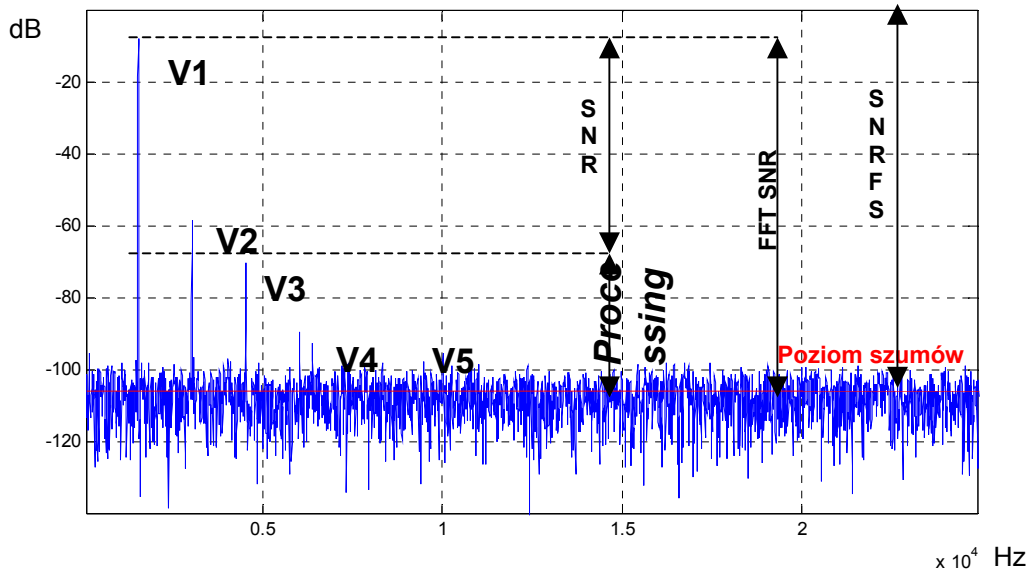
$$THD = 20*\log_{10}Sqrt((10^{-V2/20})^2+(10^{-V3/20})^2+\ldots+(10^{-V6/20})^2)$$

$$SINAD = 20*\log_{10}Sqrt((10^{-SNR/20})^{2+}(10^{-THD/20})^2)$$

where:

V2 … V6 – values in dB of harmonics peaks

An example from spectrum graph, which shows the way of estimation SNR, NF and PG is shown below.



Pic. 14
Explanation of parameters

Amplitude of the input sine wave was a little lower than full scale range of SIMCON3.1 inputs. It had to be taken into a count when ENOB was estimated. This was done by formula:

ENOB= (SINAD-1,76 + |V1|) /6,02 [dB]

Effective number of bits calculated by above formula will be always the same to each SIMCON3.1 input if measurements were be done in the same conditions.

Results of all measurements and calculation are show in table below.

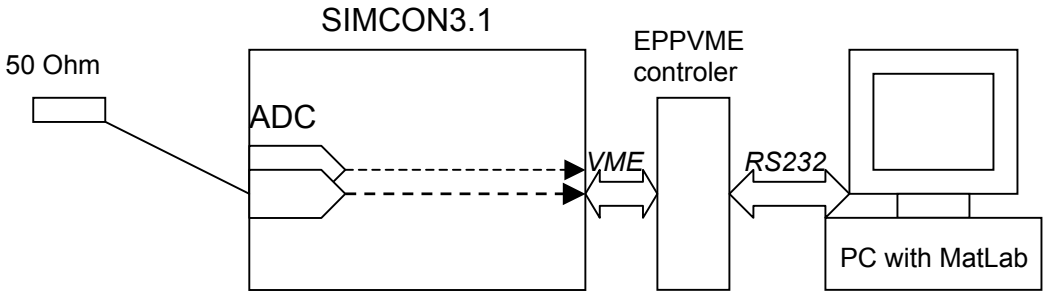
Value	**ADC1**	**ADC2**	**ADC3**	**ADC4**	**ADC5**	**ADC6**	**ADC7**	**ADC8**	**ADC9**	**ADC10**
V1 ***[dB]***	-7,6	-7,5	-7,8	-7,6	-7,6	-7,6	-7,7	-7,7	-7,4	-7,7
V2 ***[dB]***	-58,2	-58,1	-58,2	-58,1	-56,6	-58,2	-58,2	-58,2	-58,2	-58,4
V3 ***[dB]***	-69,8	-69,5	-69,1	-70,0	-72,8	-70,1	-69,9	-70,2	-69,5	-69,5
V4 ***[dB]***	-90,6	-89,3	-90	-89,7	-82,6	-89,4	-90,7	-89,5	-89,5	-87,5
V5 ***[dB]***	-101,5	-101,4	-104,6	-101,8	-86,5	-101,5	-98,8	-100,5	-101,2	-101,0
V6 ***[dB]***	-102,2	-104,8	-102,4	-101,1	-9,5	-106,1	-100,1	-98,9	-101,4	-101,7
NF ***[dB]***	-105,9	-105,9	-106,0	-105,3	-99,75	-105,8	-106,1	-106,3	-105,3	-105,9

SNR ***[dBc]***	62,1	62,1	62,2	61,6	56,0	62,1	62,2	62,4	61,7	62,2
SNRFS ***[dB]***	69,7	69,6	70,0	69,2	63,6	69,7	70,1	70,1	69,1	69,9
SINAD ***[dBc]***	62,18	62,18	62,29	61,6	56,0	62,4	62,2	62,4	61,7	62,28
ENOB	**11,31**	**11,30**	**11,32**	**11,20**	**10,27**	**11,29**	**11,33**	**11,36**	**11,21**	**11,30**

Table.8
ENOB measure results

g) Measurement of the noise parameters:

Measurement configuration schema:



Rys.15 Noise parameters measurement configuration

The SIMCON3.1 board was connected in the same way as in a subsection. In a part of measurements there was 50 Ohms resistance connected to SIMCON3.1's inputs.

Measurement results:

For each of an analogue inputs following parameters were measured:

Average – Average value of noise

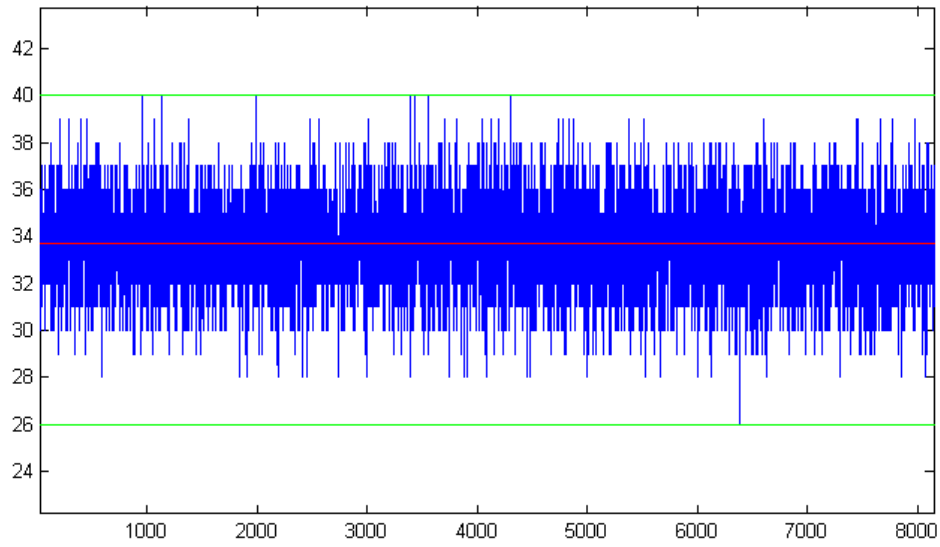
Max; Min – Maximum and minimum noise values

Sigma – standard deviation - rms

Sigma^2 – standard deviation to 2 power

NF – Noise level

Parameters were calculated from data from ADCs. ADCs outputs data is in range from 0 – 16834 (14 bits converters 2^{14}=16384). The 8192 is the 0 Volts level, so the output data were shifted by this value. It is shown below:



Rys.16
Example of noise readings (ADC1, terminated)

Blue lines it is the noise graph shifted by 8192. Red line it is average value and green line shows maximum and minimum noise values. Measure results are shown in table below:

Input	**Average**		**Min**		**Max**		**Sigma**		**Sigma^2**	**NF**
	Kod	***[mV]***	***kod***	***[mV]***	***Kod***	***[mV]***	***Kod***	***[uV]***	***Kod***	***[dB]***
ADC1	418,2	61,8	412	60,9	426	62,9	1,842	272	3,479	73,2
ADC2	393,4	58,1	386	57,0	400	59,1	1,822	269	3,322	73,3
ADC3	442,2	66,0	436	65,0	452	67,4	1,803	269	3,252	73,1
ADC4	407,6	62,1	400	60,1	415	62,4	2,151	324	4,452	71,3
ADC5	409,2	60,9	392	58,0	425	62,9	3,802	563	14,479	66,5
ADC6	404,6	61,0	397	59,3	411	61,3	1,818	271	3,305	73,2
ADC7	416,1	60,6	409	57,6	423	61,6	1,892	275	3,532	73,5
ADC8	429,8	62,4	423	61,4	436	63,3	1,797	260	3,226	72,3
ADC9	384,2	56,7	378	55,8	391	57,7	1,992	285	3,739	73,2
ADC10	450,3	65,9	443	64,8	457	66,9	1,885	276	3,563	73,7
średnia	***415,2***	***61,55***	***40,76***	***59,99***	***423,6***	***62,55***	***2,0804***	***306,4***	***4,6349***	***72,33***

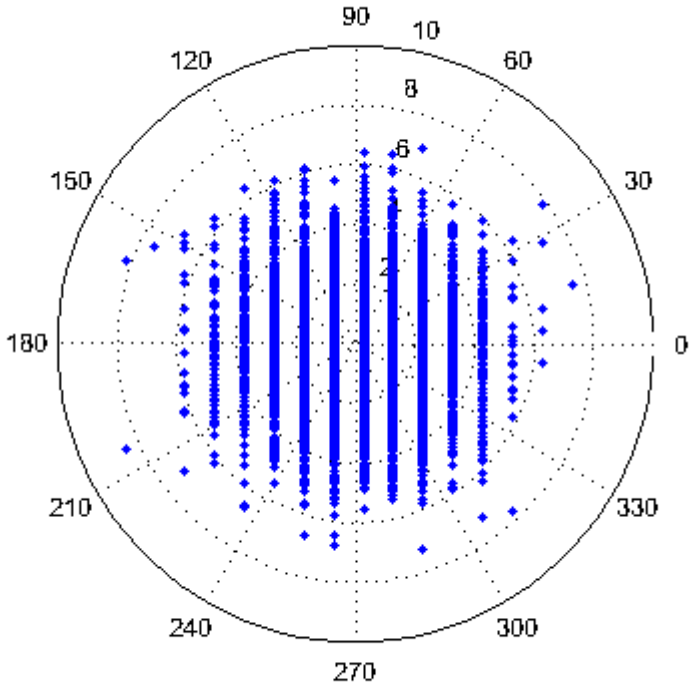
Tabela 9
Noise measurements results for unterminated inputs

Input	**Avergae**		**Min**		**Max**		**Sigma**		**Sigma^2**	**NF**
	Kod	***[mV]***	***kod***	***[mV]***	***kod***	***[mV]***	***Kod***	***[uV]***	***kod***	***[dB]***

ADC1	33,7	4,95	26	3,81	4,0	5,91	1,863	275	3,450	72,1
ADC2	3,76	5,55	-3	-4,42	10	1,52	3,419	273	3,420	72,8
ADC3	54,6	8,23	47	7,03	62	9,24	3,330	272	3,330	72,9
ADC4	19,3	2,93	9	1,41	28	4,21	4,592	322	4,592	71,6
ADC5	21,0	3,10	8	1,26	37	5,57	15,52	583	15,528	66,3
ADC6	14,9	2,23	8	1,26	21	3,14	3,385	274	3,389	72,4
ADC7	31,0	4,86	26	3,81	40	5,81	3,535	273	3,535	72,7
ADC8	42,5	6,13	35	5,16	49	7,18	3,242	261	3,248	73,0
ADC9	-9,42	-1,40	-16	-2,42	-2	-0,29	3,738	294	3,964	72,3
ADC10	60,3	8,9	54	7,90	69	10,16	3,461	272	3,461	72,1
średnia	27,164	4,548	19,4	2,48	31,8	5,245	4,6085	309,9	4,7917	71,82

Tabela 10
Noise results for terminated inputs

To check jitter influence on noise, there was made a graph which shows amplitude and phase of the noise in polar coordinates. This graph was made from noise data after removing average value and transforming data by Hilibert transform (hilibert() function in MatLab).



Rys.17
Phase and module graph from terminated ADC3 noise data

It can be seen that noise is evenly composed around the center. There are no elliptic disturbances which can point on jitter as source of the noise.

Error estimation methods:

Calculating values in Volts were done by multiplying them by quantization level, so all values in Volts from table 10 have $\delta pk=0,4\%$ error.

Appendix. 1

Xilinx pins speciffications

SIGNAL NAME	PIN	STANDARD
"lclk"	"AK17"	LVCMOS25 ;
"lclk_ena"	"AL17"	LVCMOS25 ;
"LED<0>"	"G22"	LVCMOS25 ;
"LED<1>"	"G25"	LVCMOS25 ;
"LED<2>"	"G26"	LVCMOS25 ;
"LED<3>"	"H19"	LVCMOS25 ;
"LED<4>"	"H21"	LVCMOS25 ;
"LED<5>"	"H22"	LVCMOS25 ;
"LED<6>"	"H25"	LVCMOS25 ;
"LED<7>"	"H26"	LVCMOS25 ;
"POWER_EN"	"AJ8"	LVCMOS25 ;
"RXD"	"AJ27"	LVCMOS25 ;
"TXD"	"AJ28"	LVCMOS25 ;
"rs_resetN"	"U30"	LVCMOS25 ;
"FPGA_CLKOUTn"	"D16"	
"FPGA_CLKOUTp"	"E16"	
"FPGA_CLKSEL"	"D15"	LVCMOS25 ;
"trg_in_p"	"G17"	LVCMOS25 ;
"trg_in_n"	"F17"	LVCMOS25 ;
"str_in_p"	"G16"	LVCMOS25 ;
"str_in_n"	"F16"	LVCMOS25 ;
"dig_out0_p"	"J16"	LVCMOS25 ;
"dig_out0_n"	"H16"	LVCMOS25 ;
"dig_out1_p"	"G15"	LVCMOS25 ;
"dig_out1_n"	"F15"	LVCMOS25 ;
"ADC1_D<0>"	"L2"	LVCMOS25 ;
"ADC1_D<1>"	"L1"	LVCMOS25 ;
"ADC1_D<2>"	"K2"	LVCMOS25 ;
"ADC1_D<3>"	"K1"	LVCMOS25 ;
"ADC1_D<4>"	"J2"	LVCMOS25 ;
"ADC1_D<5>"	"H2"	LVCMOS25 ;
"ADC1_D<6>"	"H1"	LVCMOS25 ;
"ADC1_D<7>"	"G2"	LVCMOS25 ;
"ADC1_D<8>"	"G1"	LVCMOS25 ;
"ADC1_D<9>"	"F2"	LVCMOS25 ;
"ADC1_D<10>"	"F1"	LVCMOS25 ;
"ADC1_D<11>"	"E2"	LVCMOS25 ;
"ADC1_D<12>"	"E1"	LVCMOS25 ;
"ADC1_D<13>"	"D2"	LVCMOS25 ;
"ADC1_OVR"	"M1"	LVCMOS25 ;
"ADC1_DRY"	"D1"	LVCMOS25 ;
"ADC2_D<0>"	"H6"	LVCMOS25 ;
"ADC2_D<1>"	"J6"	LVCMOS25 ;
"ADC2_D<2>"	"L6"	LVCMOS25 ;

"ADC2_D<3>"	"M6"	LVCMOS25 ;
"ADC2_D<4>"	"N6"	LVCMOS25 ;
"ADC2_D<5>"	"P6"	LVCMOS25 ;
"ADC2_D<6>"	"R6"	LVCMOS25 ;
"ADC2_D<7>"	"T6"	LVCMOS25 ;
"ADC2_D<8>"	"U6"	LVCMOS25 ;
"ADC2_D<9>"	"V6"	LVCMOS25 ;
"ADC2_D<10>"	"W6"	LVCMOS25 ;
"ADC2_D<11>"	"Y6"	LVCMOS25 ;
"ADC2_D<12>"	"AA6"	LVCMOS25 ;
"ADC2_D<13>"	"AB6"	LVCMOS25 ;
"ADC2_OVR"	"G6"	LVCMOS25 ;
"ADC2_DRY"	"AC6"	LVCMOS25 ;
"ADC3_D<0>"	"AA2"	LVCMOS25 ;
"ADC3_D<1>"	"AA1"	LVCMOS25 ;
"ADC3_D<2>"	"Y2"	LVCMOS25 ;
"ADC3_D<3>"	"Y1"	LVCMOS25 ;
"ADC3_D<4>"	"W2"	LVCMOS25 ;
"ADC3_D<5>"	"V2"	LVCMOS25 ;
"ADC3_D<6>"	"U2"	LVCMOS25 ;
"ADC3_D<7>"	"T2"	LVCMOS25 ;
"ADC3_D<8>"	"R2"	LVCMOS25 ;
"ADC3_D<9>"	"R1"	LVCMOS25 ;
"ADC3_D<10>"	"P2"	LVCMOS25 ;
"ADC3_D<11>"	"P1"	LVCMOS25 ;
"ADC3_D<12>"	"N2"	LVCMOS25 ;
"ADC3_D<13>"	"N1"	LVCMOS25 ;
"ADC3_OVR"	"AB1"	LVCMOS25 ;
"ADC3_DRY"	"M2"	LVCMOS25 ;
"DAC1_D<0>"	"D14"	LVCMOS25 ;
"DAC1_D<10>"	"F8"	LVCMOS25 ;
"DAC1_D<11>"	"E7"	LVCMOS25 ;
"DAC1_D<12>"	"D6"	LVCMOS25 ;
"DAC1_D<13>"	"E6"	LVCMOS25 ;
"DAC1_D<1>"	"C13"	LVCMOS25 ;
"DAC1_D<2>"	"D13"	LVCMOS25 ;
"DAC1_D<3>"	"J14"	LVCMOS25 ;
"DAC1_D<4>"	"C11"	LVCMOS25 ;
"DAC1_D<5>"	"E10"	LVCMOS25 ;
"DAC1_D<6>"	"D10"	LVCMOS25 ;
"DAC1_D<7>"	"D9"	LVCMOS25 ;
"DAC1_D<8>"	"C9"	LVCMOS25 ;
"DAC1_D<9>"	"E9"	LVCMOS25 ;
"DAC2_D<0>"	"G14"	LVCMOS25 ;
"DAC2_D<10>"	"G10"	LVCMOS25 ;
"DAC2_D<11>"	"G13"	LVCMOS25 ;
"DAC2_D<12>"	"G9"	LVCMOS25 ;
"DAC2_D<13>"	"H10"	LVCMOS25 ;
"DAC2_D<1>"	"H13"	LVCMOS25 ;
"DAC2_D<2>"	"H9"	LVCMOS25 ;
"DAC2_D<3>"	"F9"	LVCMOS25 ;
"DAC2_D<4>"	"E14"	LVCMOS25 ;
"DAC2_D<5>"	"E13"	LVCMOS25 ;
"DAC2_D<6>"	"F14"	LVCMOS25 ;
"DAC2_D<7>"	"H14"	LVCMOS25 ;
"DAC2_D<8>"	"J15"	LVCMOS25 ;
"DAC2_D<9>"	"D11"	LVCMOS25 ;
"DAC3_D<0>"	"J11"	LVCMOS25 ;
"DAC3_D<10>"	"L8"	LVCMOS25 ;

```
"DAC3_D<11>"        "N8"        LVCMOS25 ;
"DAC3_D<12>"        "P8"        LVCMOS25 ;
"DAC3_D<13>"        "T8"        LVCMOS25 ;
"DAC3_D<1>"         "J10"       LVCMOS25 ;
"DAC3_D<2>"         "J12"       LVCMOS25 ;
"DAC3_D<3>"         "D12"       LVCMOS25 ;
"DAC3_D<4>"         "D5"        LVCMOS25 ;
"DAC3_D<5>"         "F7"        LVCMOS25 ;
"DAC3_D<6>"         "E4"        LVCMOS25 ;
"DAC3_D<7>"         "E3"        LVCMOS25 ;
"DAC3_D<8>"         "J8"        LVCMOS25 ;
"DAC3_D<9>"         "K8"        LVCMOS25 ;
"DAC4_D<0>"         "U8"        LVCMOS25 ;
"DAC4_D<10>"        "T9"        LVCMOS25 ;
"DAC4_D<11>"        "U9"        LVCMOS25 ;
"DAC4_D<12>"        "V9"        LVCMOS25 ;
"DAC4_D<13>"        "W9"        LVCMOS25 ;
"DAC4_D<1>"         "V8"        LVCMOS25 ;
"DAC4_D<2>"         "W8"        LVCMOS25 ;
"DAC4_D<3>"         "AA8"       LVCMOS25 ;
"DAC4_D<4>"         "AB8"       LVCMOS25 ;
"DAC4_D<5>"         "L9"        LVCMOS25 ;
"DAC4_D<6>"         "M9"        LVCMOS25 ;
"DAC4_D<7>"         "N9"        LVCMOS25 ;
"DAC4_D<8>"         "P9"        LVCMOS25 ;
"DAC4_D<9>"         "R9"        LVCMOS25 ;
"DAC_MOD"           "C14"       LVCMOS25 ;
"DAC_SLEEP"         "F13"       LVCMOS25 ;


"ADC4_D<0>"         "K7"        LVCMOS25 ;
"ADC4_D<1>"         "L7"        LVCMOS25 ;
"ADC4_D<2>"         "M7"        LVCMOS25 ;
"ADC4_D<3>"         "N7"        LVCMOS25 ;
"ADC4_D<4>"         "P7"        LVCMOS25 ;
"ADC4_D<5>"         "R7"        LVCMOS25 ;
"ADC4_D<6>"         "T7"        LVCMOS25 ;
"ADC4_D<7>"         "U7"        LVCMOS25 ;
"ADC4_D<8>"         "V7"        LVCMOS25 ;
"ADC4_D<9>"         "W7"        LVCMOS25 ;
"ADC4_D<10>"        "Y7"        LVCMOS25 ;
"ADC4_D<11>"        "AA7"       LVCMOS25 ;
"ADC4_D<12>"        "AB7"       LVCMOS25 ;
"ADC4_D<13>"        "AC7"       LVCMOS25 ;

"ADC5_D<0>"         "AK1"       LVCMOS25 ;
"ADC5_D<1>"         "AJ2"       LVCMOS25 ;
"ADC5_D<2>"         "AJ1"       LVCMOS25 ;
"ADC5_D<3>"         "AH2"       LVCMOS25 ;
"ADC5_D<4>"         "AH1"       LVCMOS25 ;
"ADC5_D<5>"         "AG2"       LVCMOS25 ;
"ADC5_D<6>"         "AG1"       LVCMOS25 ;
"ADC5_D<7>"         "AF2"       LVCMOS25 ;
"ADC5_D<8>"         "AE2"       LVCMOS25 ;
"ADC5_D<9>"         "AE1"       LVCMOS25 ;
"ADC5_D<10>"        "AD2"       LVCMOS25 ;
"ADC5_D<11>"        "AD1"       LVCMOS25 ;
"ADC5_D<12>"        "AC2"       LVCMOS25 ;
"ADC5_D<13>"        "AC1"       LVCMOS25 ;
```

"ii_addr<0>"	"AG25"	LVCMOS25 ;
"ii_addr<10>"	"AJ24"	LVCMOS25 ;
"ii_addr<11>"	"AJ22"	LVCMOS25 ;
"ii_addr<12>"	"AJ21"	LVCMOS25 ;
"ii_addr<13>"	"AJ20"	LVCMOS25 ;
"ii_addr<14>"	"AJ19"	LVCMOS25 ;
"ii_addr<15>"	"AK27"	LVCMOS25 ;
"ii_addr<16>"	"AK24"	LVCMOS25 ;
"ii_addr<17>"	"AK22"	LVCMOS25 ;
"ii_addr<18>"	"AK21"	LVCMOS25 ;
"ii_addr<19>"	"AK19"	LVCMOS25 ;
"ii_addr<1>"	"AG22"	LVCMOS25 ;
"ii_addr<20>"	"AL28"	LVCMOS25 ;
"ii_addr<21>"	"AL27"	LVCMOS25 ;
"ii_addr<22>"	"AL24"	LVCMOS25 ;
"ii_addr<23>"	"AL23"	LVCMOS25 ;
"ii_addr<24>"	"AL22"	LVCMOS25 ;
"ii_addr<25>"	"AL21"	LVCMOS25 ;
"ii_addr<26>"	"AL20"	LVCMOS25 ;
"ii_addr<27>"	"AL19"	LVCMOS25 ;
"ii_addr<28>"	"AM28"	LVCMOS25 ;
"ii_addr<29>"	"AM24"	LVCMOS25 ;
"ii_addr<2>"	"AG21"	LVCMOS25 ;
"ii_addr<30>"	"AM22"	LVCMOS25 ;
"ii_addr<31>"	"AM21"	LVCMOS25 ;
"ii_addr<3>"	"AG19"	LVCMOS25 ;
"ii_addr<4>"	"AG18"	LVCMOS25 ;
"ii_addr<5>"	"AH25"	LVCMOS25 ;
"ii_addr<6>"	"AH22"	LVCMOS25 ;
"ii_addr<7>"	"AH21"	LVCMOS25 ;
"ii_addr<8>"	"AH20"	LVCMOS25 ;
"ii_addr<9>"	"AH19"	LVCMOS25 ;
"ii_data<0>"	"AG17"	LVCMOS25 ;
"ii_data<10>"	"AF18"	LVCMOS25 ;
"ii_data<11>"	"AJ15"	LVCMOS25 ;
"ii_data<12>"	"AJ14"	LVCMOS25 ;
"ii_data<13>"	"AJ13"	LVCMOS25 ;
"ii_data<14>"	"AJ11"	LVCMOS25 ;
"ii_data<15>"	"AK16"	LVCMOS25 ;
"ii_data<16>"	"AK14"	LVCMOS25 ;
"ii_data<17>"	"AK13"	LVCMOS25 ;
"ii_data<18>"	"AK11"	LVCMOS25 ;
"ii_data<19>"	"AK8"	LVCMOS25 ;
"ii_data<1>"	"AG16"	LVCMOS25 ;
"ii_data<20>"	"AL16"	LVCMOS25 ;
"ii_data<21>"	"AL15"	LVCMOS25 ;
"ii_data<22>"	"AL14"	LVCMOS25 ;
"ii_data<23>"	"AL13"	LVCMOS25 ;
"ii_data<24>"	"AL12"	LVCMOS25 ;
"ii_data<25>"	"AL11"	LVCMOS25 ;
"ii_data<26>"	"AL8"	LVCMOS25 ;
"ii_data<27>"	"AL7"	LVCMOS25 ;
"ii_data<28>"	"AM14"	LVCMOS25 ;
"ii_data<29>"	"AM13"	LVCMOS25 ;
"ii_data<2>"	"AG14"	LVCMOS25 ;
"ii_data<30>"	"AM11"	LVCMOS25 ;
"ii_data<31>"	"AM7"	LVCMOS25 ;
"ii_data<3>"	"AG13"	LVCMOS25 ;
"ii_data<4>"	"AG10"	LVCMOS25 ;
"ii_data<5>"	"AH16"	LVCMOS25 ;

"ii_data<6>"	"AH15"	LVCMOS25 ;
"ii_data<7>"	"AH14"	LVCMOS25 ;
"ii_data<8>"	"AH13"	LVCMOS25 ;
"ii_data<9>"	"AH10"	LVCMOS25 ;
"ii_irqN<0>"	"AF10"	LVCMOS25 ;
"ii_irqN<1>"	"AF11"	LVCMOS25 ;
"ii_irqN<2>"	"AF13"	LVCMOS25 ;
"ii_irqN<3>"	"AF14"	LVCMOS25 ;
"ii_irqN<4>"	"AF15"	LVCMOS25 ;
"ii_irqN<5>"	"AF16"	LVCMOS25 ;
"ii_irqN<6>"	"AF17"	LVCMOS25 ;
"ii_irqN<7>"	"AJ16"	LVCMOS25 ;
"ii_operN"	"AF20"	LVCMOS25 ;
"ii_resetN"	"AF21"	LVCMOS25 ;
"ii_strobeN"	"AF22"	LVCMOS25 ;
"ii_writeN"	"AF24"	LVCMOS25 ;