WP2.4 Piezo Compenstation Status of Subtasks

Konrad Przygoda

Technical University of Lódź, Poland
Dep. of Microelectronics and Computer Science

WP2.4 – Piezo Compensation

- **WP2.4 - 10 subtasks (20 deliverables)**
  - 2.4.1 Preparation of requirements in EA (1)
  - 2.4.2 Piezo controller development with SimconDSP board (3)
  - 2.4.3 Piezo controller development with PowerQUICC III processor and Virtex 5 FPGA and 32-channel DAC board (4)
  - 2.4.4 Design and laboratory tests of piezo driver prototypes (1)
  - 2.4.5 Design and development of 8-channel piezo drivers for permanent installation in FLASH accelerator (4)
  - 2.4.6 Design and development of 8-channel piezo drivers (1)
  - 2.4.7 Design and development of 32-channel control system with gigalinks (2)
  - 2.4.8 The preliminary tests of whole system before FLASH tests (1)
  - 2.4.9 Tests of piezo driver and prototype controller in FLASH (1)
  - 2.4.10 Detuning computation algorithm implemented in PowerQUICC III processor (2)

2.4.1 Preparation of requirements in EA

1) Short description of deliverables:

The requirements diagrams for integration of piezo control system to ATCA architecture were defined. The block definition diagrams for piezo control system were designed. The activity diagrams for control algorithms were designed.

2) Current status:

/RF_Station_Model/RF_Station_Structure/LLRF_Structure/CavityResonanceControl
Requirements diagrams (50%),
Block definition diagrams (50%),
Activity diagrams (50%),

3) Important achievements:

Elementary knowledge of Enterprise Architect and its functional blocks

4) Delivery date, reviewer name, the date when you plan to contact the reviewer:

30.11.2008, M. Grecki, not specified yet

2.4.1 Preparation of requirements in EA

2.4.2 Piezo controller development with SimconDSP board

1) Short description of deliverables:

The prototype piezo control system was designed. It consistes of SimconDSP boards and 8-channel piezo drivers. The 8-channel detuning algorithm was implemented using Virtex II Pro FPGA for monitoring the Lorentz force detuning. The tool can be also used for microphonics identyfication. The prototype system was also used to measure the compensation characteristics and to develope the algorithm for automatic computation of piezo pulse amplitude.

2) Current subtask status:

8-channel scope for LFD (80%) – VHDL code, Matlab code
Piezo excitation paramters automatic computation (80%) – VHDL code, Matlab code

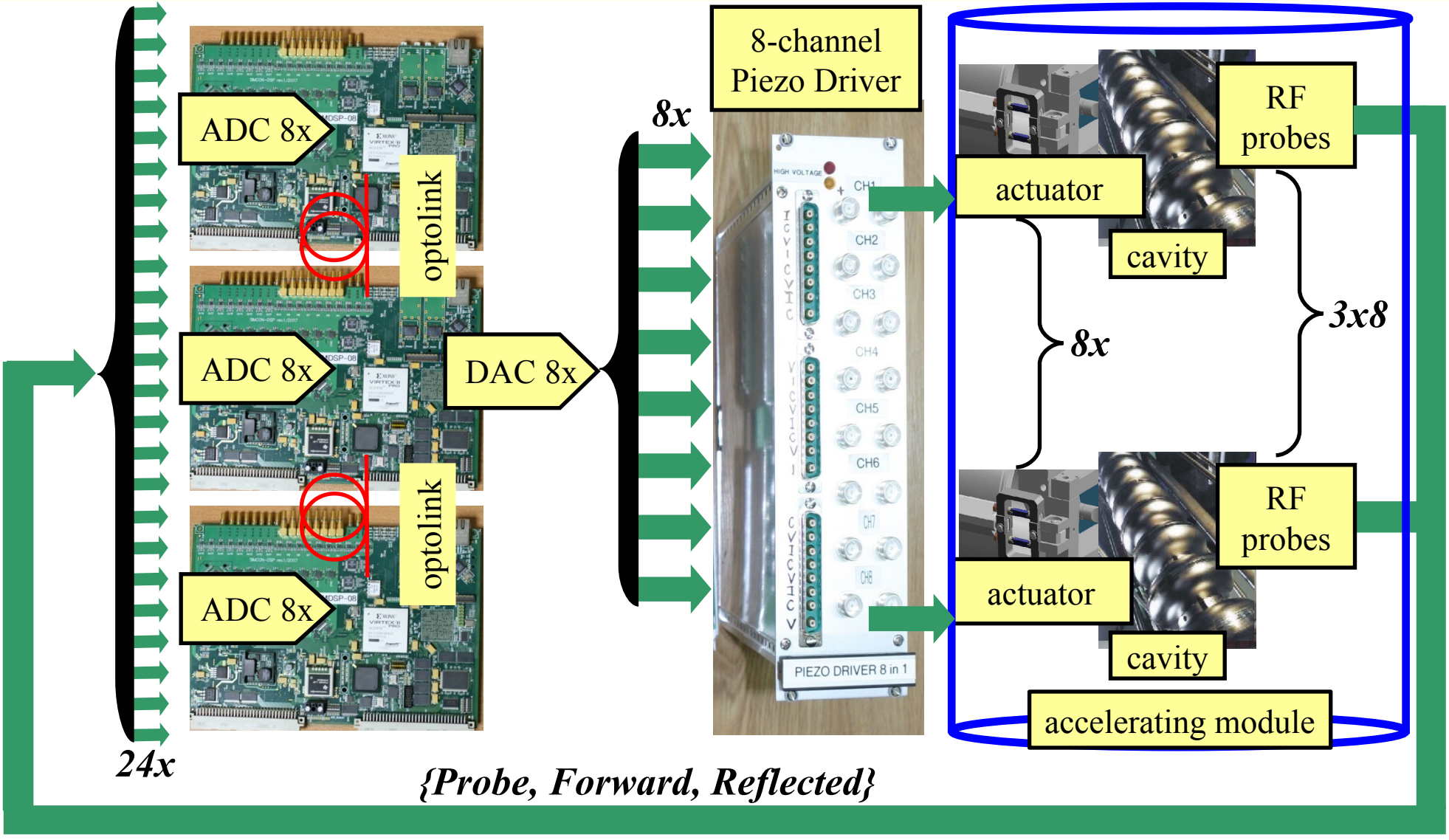
3) Important achievements:

Laboratory and FLASH tests

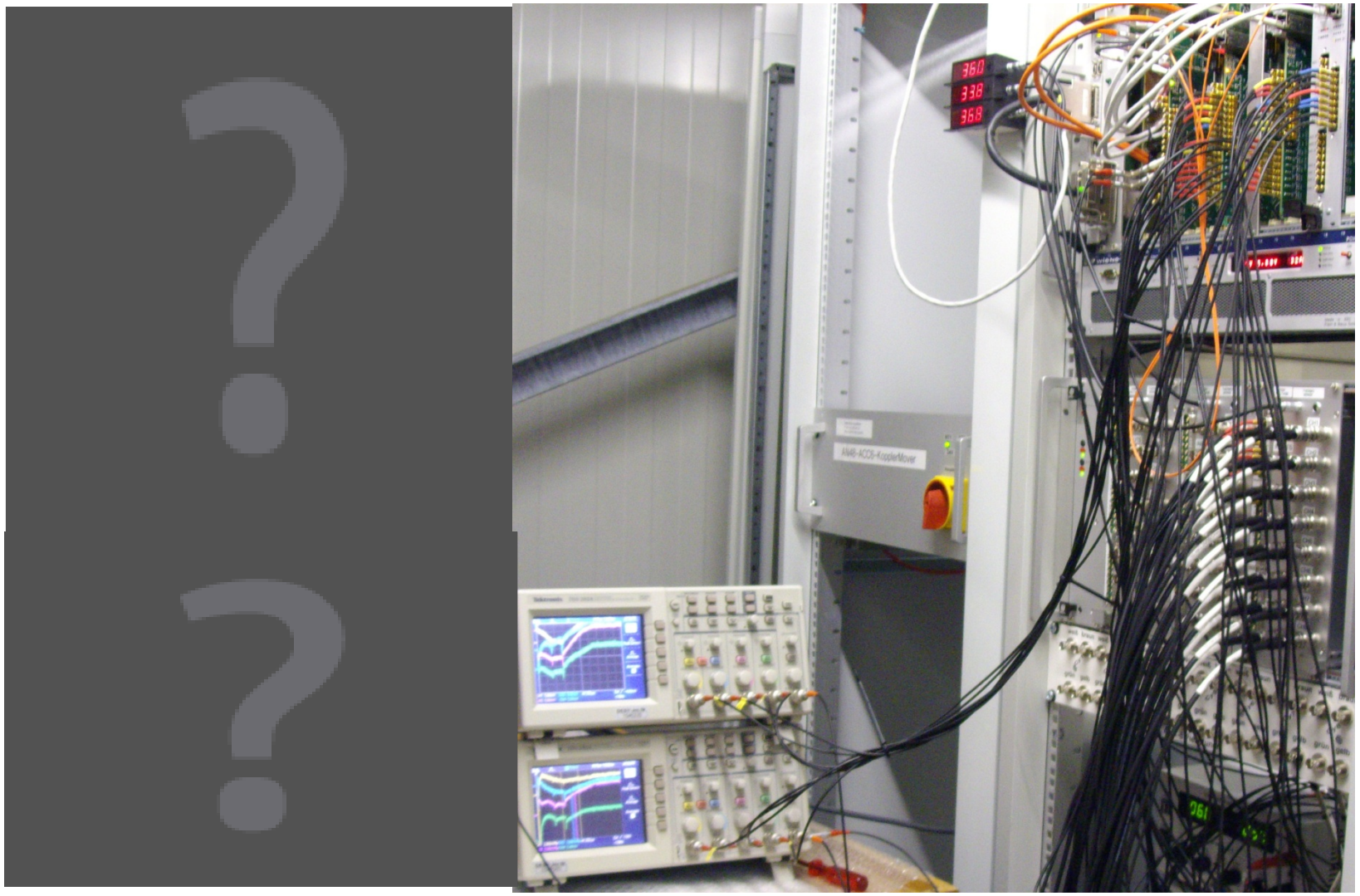
4) Delivery date, reviewer name, the date when you plan to contact the reviewer:

30.09.2008, M. Grecki, not specified yet

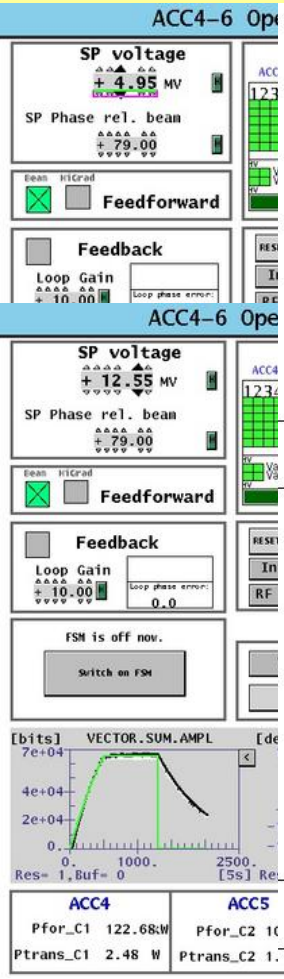
2.4.2 Piezo controller development with SimconDSP board

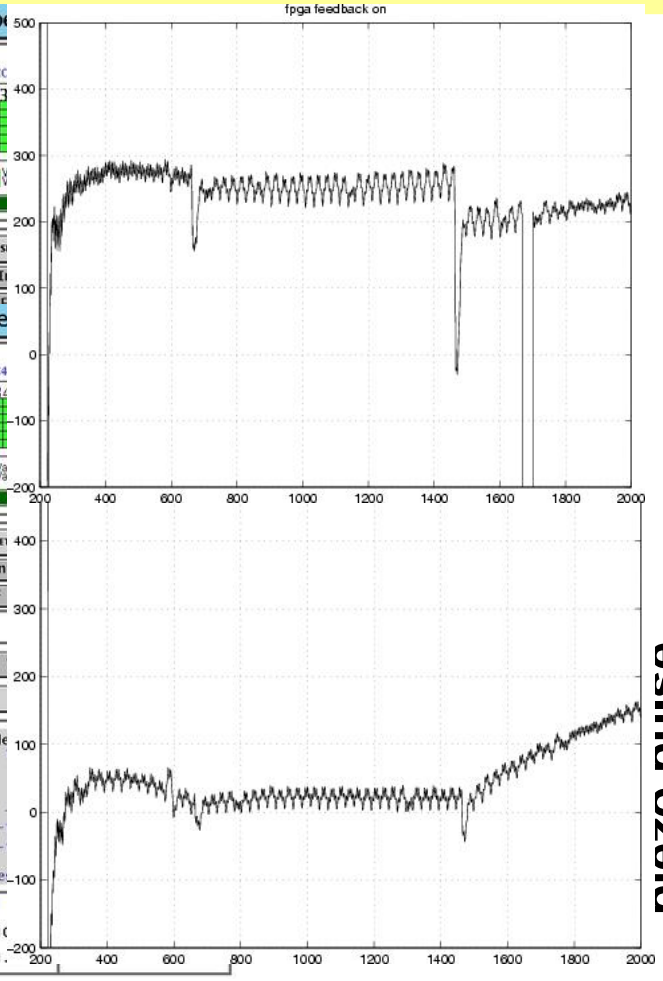


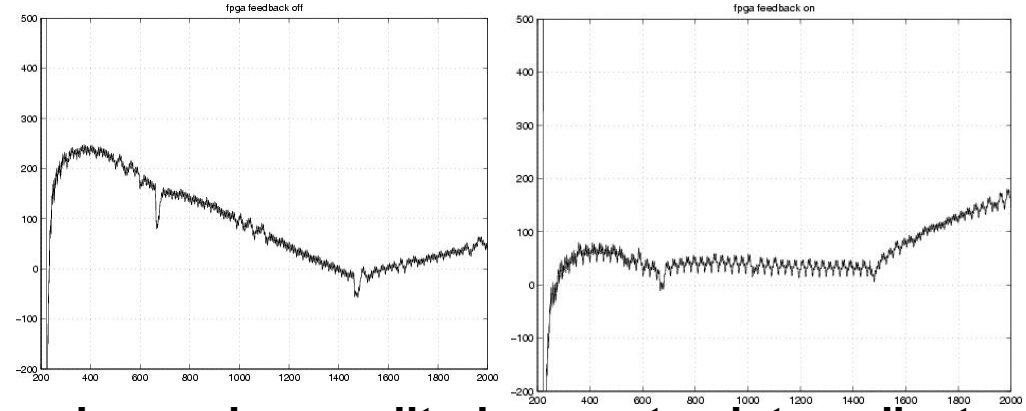
2.4.2 Piezo controller development with SimconDSP board



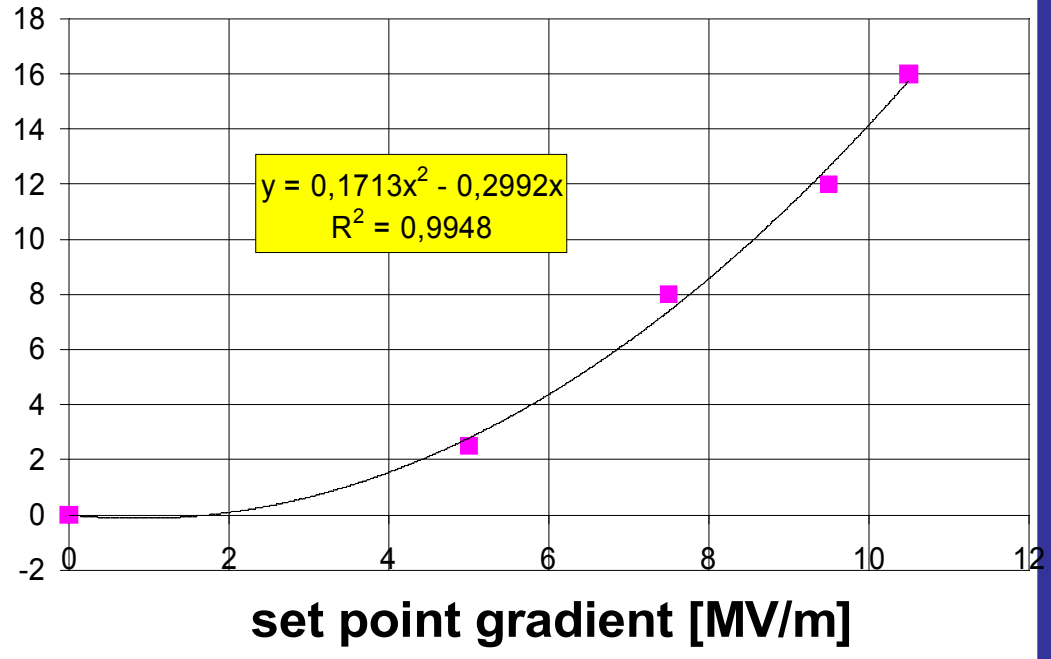
2.4.2 Piezo controller development with SimconDSP board







piezo pulse amplitude vs. set point gradient



2.4.3 Piezo controller development with PowerQUICC III processor and Virtex 5 FPGA and 32-channel DAC board

1) Short description of deliverables:

The piezo control system was designed. It consists of evaluation board from Freescale (based on PowerQUICC III processor) , evaluation board from Xilinx (based on Virtex 5) and 32-channel DAC board (based on integrated DAC device). The DAC board is controlled by client application thu matlab scripts. The server application and linux device drivers are used to communicate with PCIe device (FPGA). The 32-channel DAC control core was implemented to control the external DAC board using SPI interface.

2) Current subtask status:

32-channel prototype board (80%) – pcb board
Client application and run scripts (80%) – C/C++ code, Matlab code
32-channel DAC control core (80%) – VHDL code

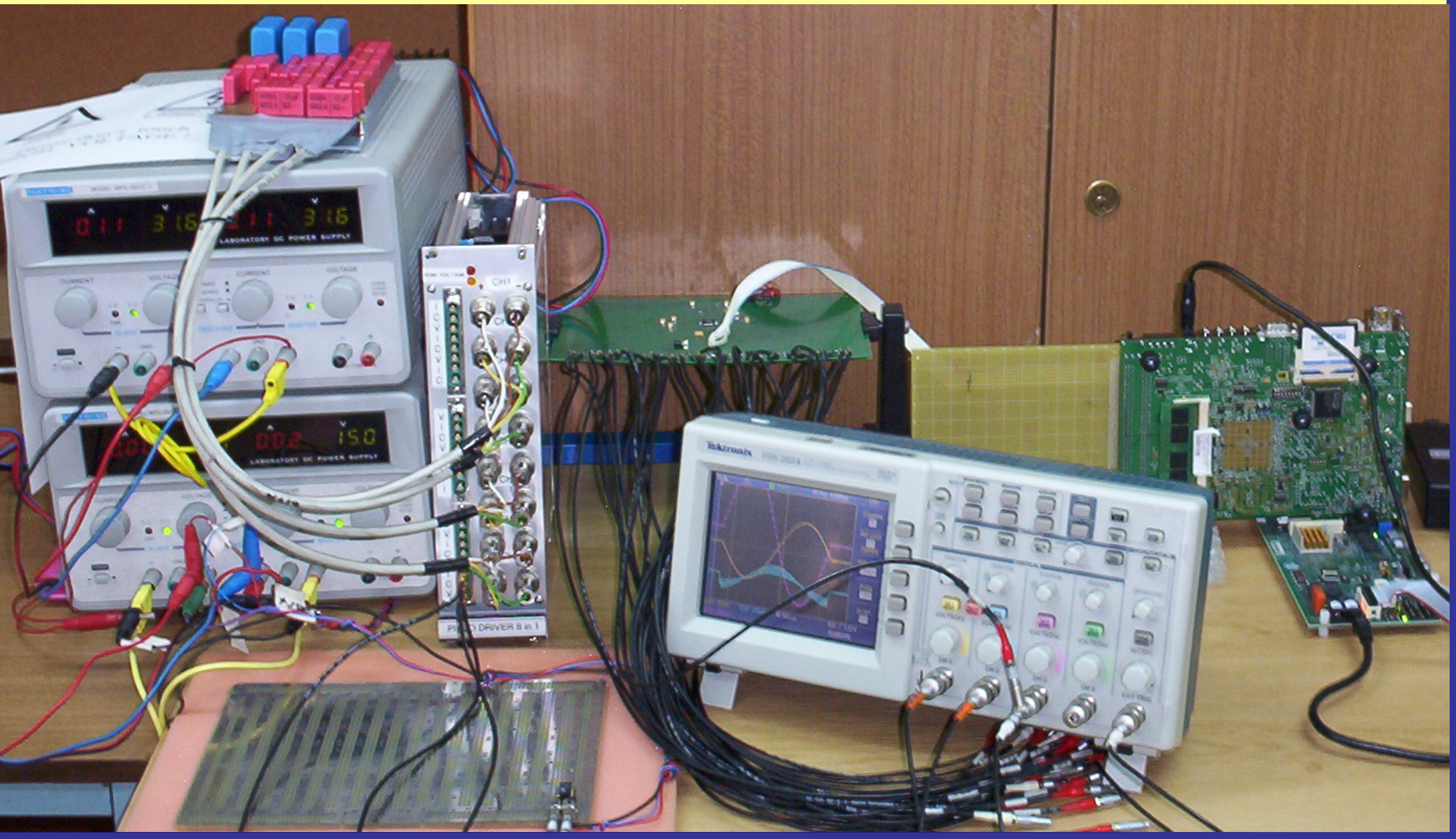
3) Important achievements:

Laboratory and FLASH tests

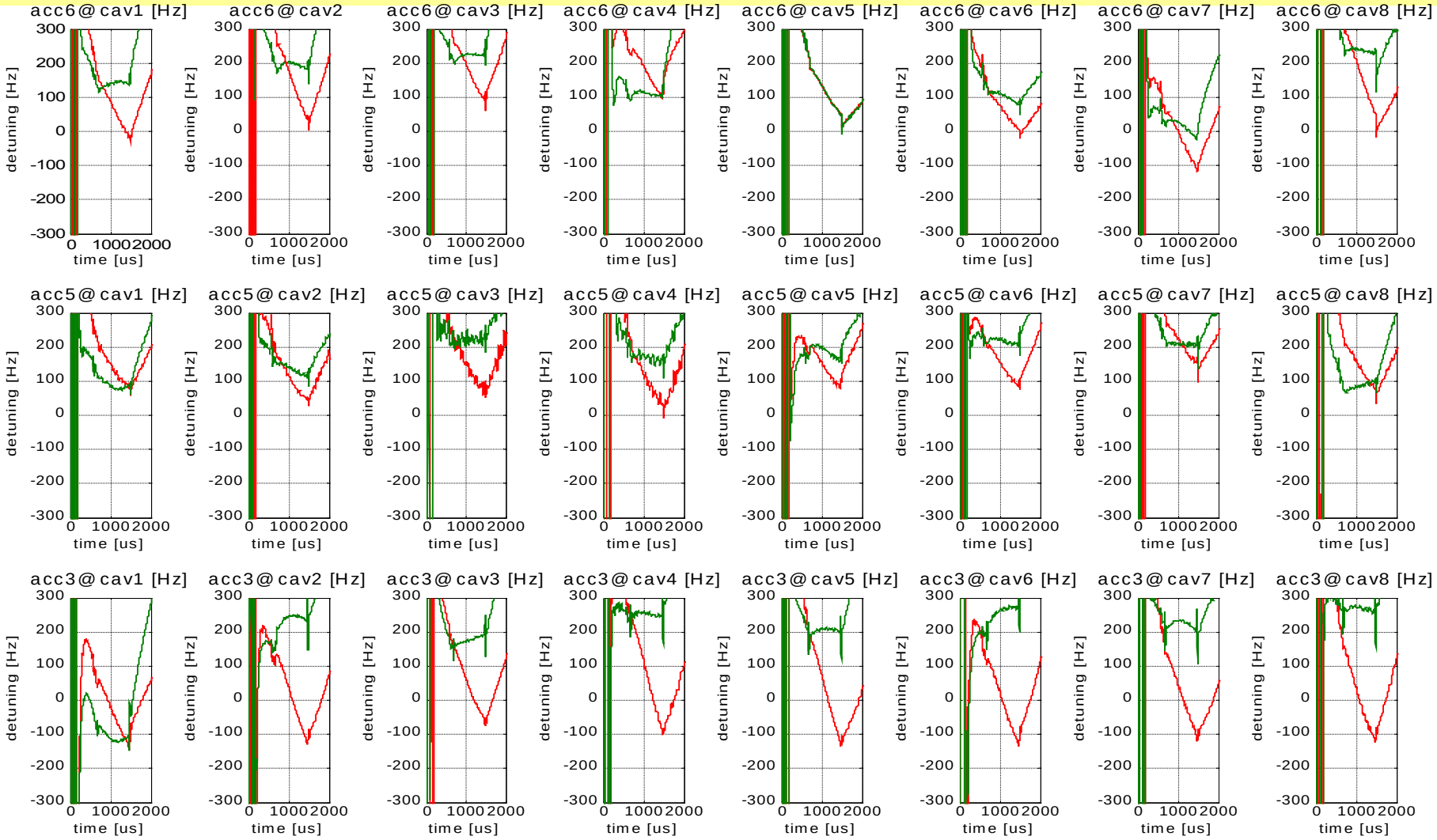
4) Delivery date, reviewer name, the date when you plan to contact the reviewer:

30.08.2008, M. Grecki, not specified yet

2.4.3 Piezo controller development with PowerQUICC III processor and Virtex 5 FPGA and 32-channel DAC board



2.4.3 Piezo controller development with PowerQUICC III processor and Virtex 5 FPGA and 32-channel DAC board



2.4.4 Design and laboratory tests of piezo drivers prototypes

1) Short description of deliverables:

Two prototypes of piezo drivers were designed and tested. The piezo driver ver. 1 is capable of driving single piezo. The piezo driver ver. 2 is capable of driving 8 piezos.

2) Current subtask status:

Piezo driver ver. 1 (80%) – stand alone box
Piezo driver ver. 2 (80%) – stand alone box

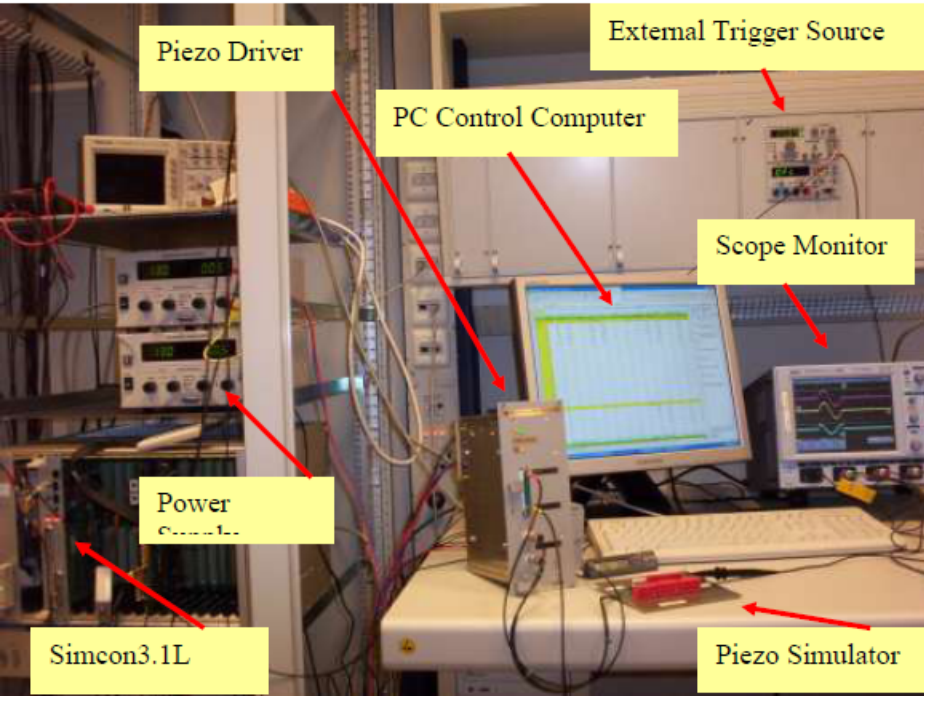
3) Important achievements:

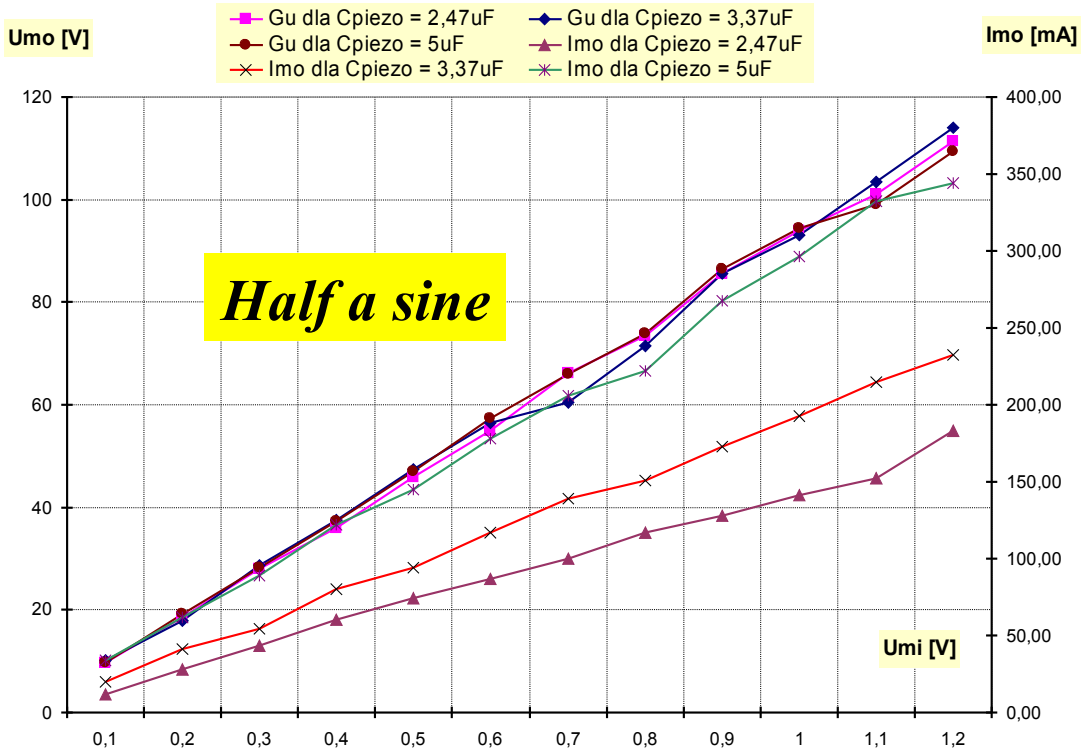
Laboratory and FLASH tests

4) Delivery date, reviewer name, the date when you plan to contact the reviewer:

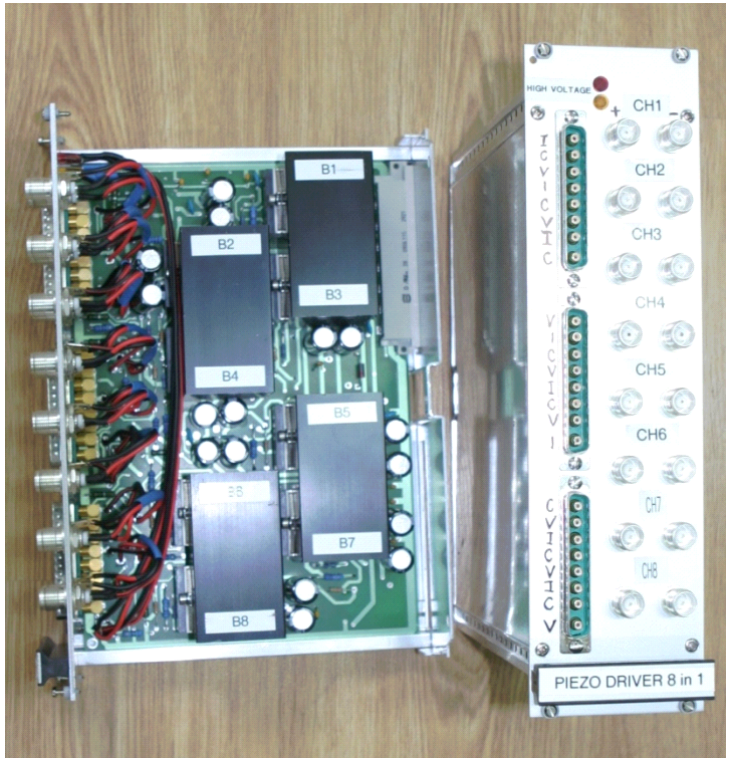
30.11.2008, M. Grecki, not specified yet

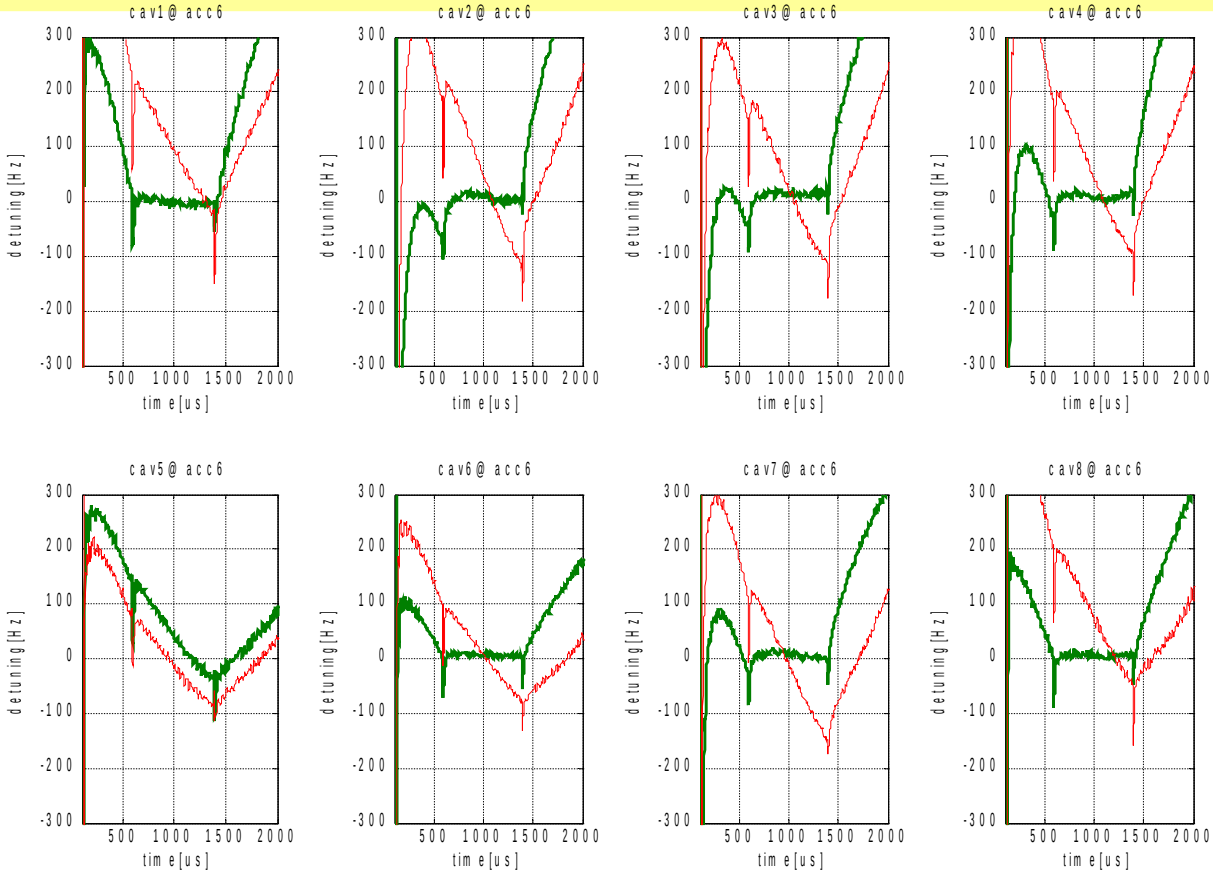
2.4.4 Design and laboratory tests of piezo drivers prototypes





2.4.4 Design and laboratory tests of piezo drivers prototypes





2.4.5 Design and development of 8-channel piezo driver for permenent installation in FLASH accelerator

1) Short description of deliverables:

The 3 boxes of piezo drivers were designed and tested. The signle piezo driver unit is capable of driving the 8 piezos. Moreover the overvoltage and overcurrent protection circuits were added to avoid the piezo failure. For diagnostics the temperature monitoring circuits were added.

2) Current subtask status:

3 boxes of 8-channel piezo driver units (80%) – stand alone box
Programmable function generator (80%) – VHDL code, Matlab code, C/C++ code

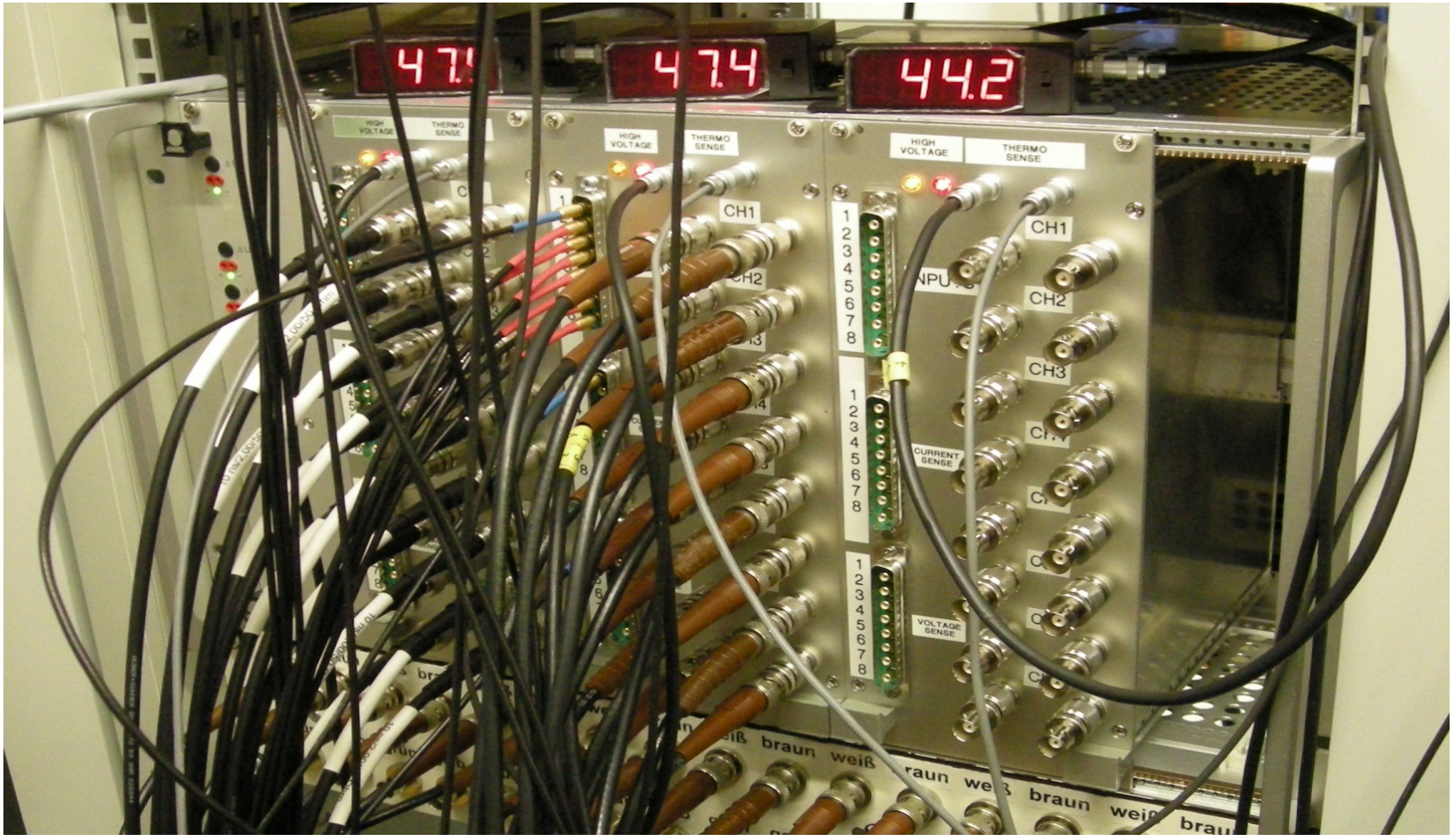
3) Important achievements:

Laboratory and FLASH tests

4) Delivery date, reviewer name, the date when you plan to contact the reviewer:

30.11.2008, M. Grecki, not specified yet

2.4.5 Design and development of 8-channel piezo driver for permenent installation in FLASH accelerator



2.4.6 Design and development of 8-channel piezo driver

1) Short description of deliverables:

The 3 boxes of 8-channels piezo drivers will be designed and tested. The piezo driver units will have integrate protection circuits as well as back connections to control system (driving signals and power monitoring signals).

2) Current subtask status:

ATCA architecture studies

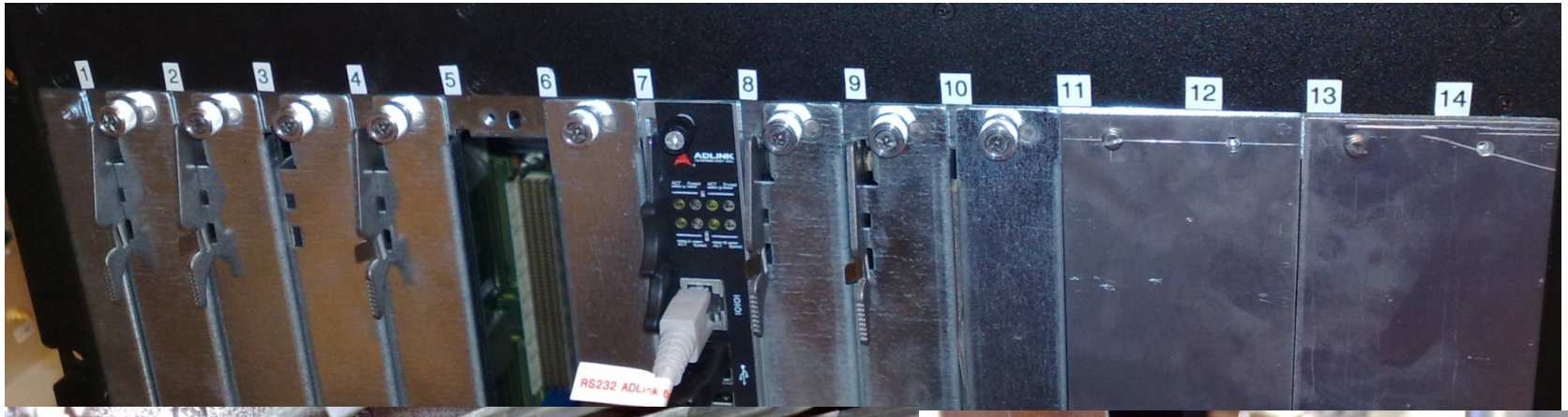
3) Important achievements:

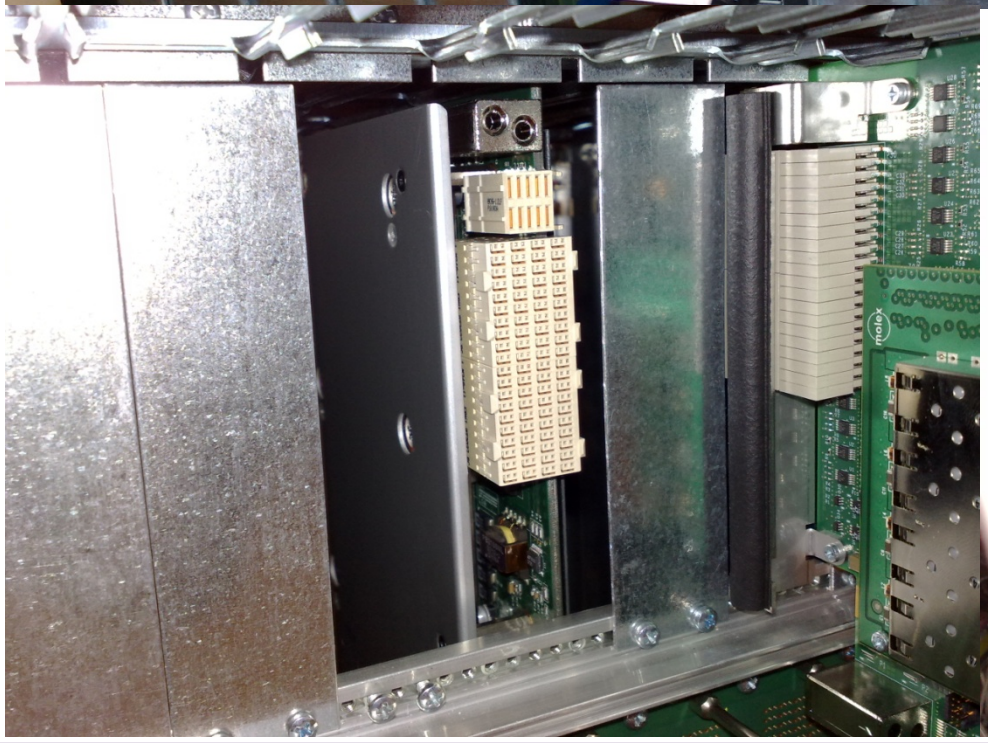
Some new ideas

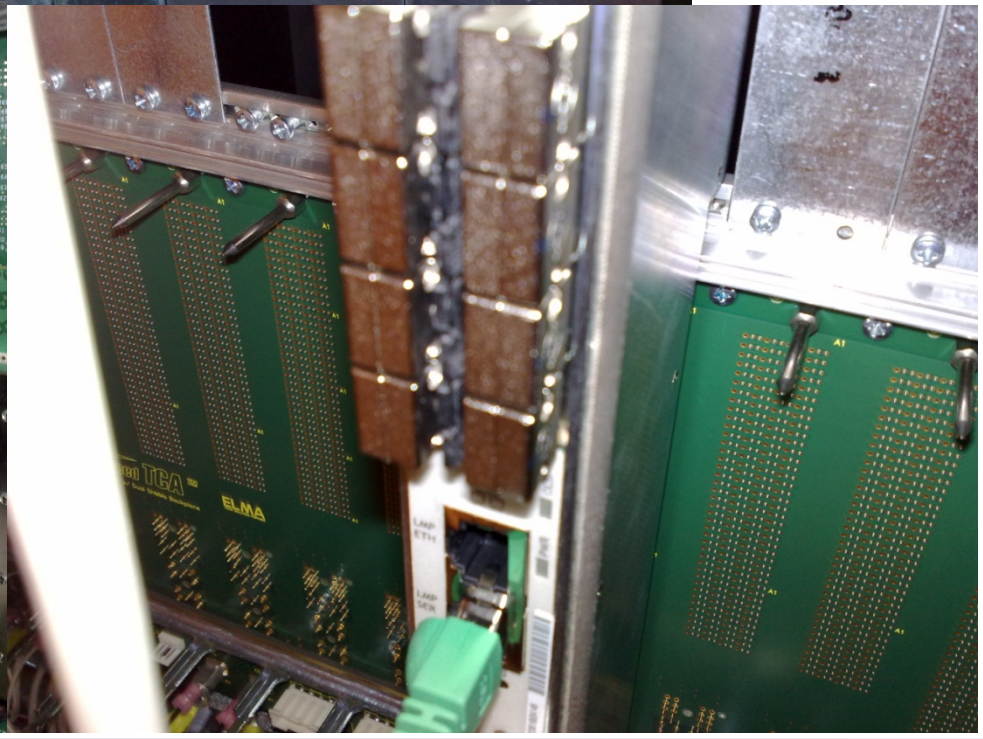
4) Delivery date, reviewer name, the date when you plan to contact the reviewer:

30.11.2008, M. Grecki, not specified yet

2.4.6 Design and development of 8-channel piezo driver







2.4.6 Design and development of 8-channel piezo driver

4.1.2 Supported voltage levels

4.1.2.1 Voltage level background information (informative)

Table 4-1 ETSI EN 300 132-2 static voltage levels

	–48 VDC systems	–60 VDC systems
Nominal operating voltage	–48 VDC	–60 VDC
Maximum operating voltage	–57 VDC	–72 VDC
Minimum operating voltage	–40.5 VDC	–50 VDC
Degraded operating voltage	–40.5 VDC to –44 VDC	Not specified
Non-operating voltages with no equipment damage	0 VDC to –40.5 VDC, –57 VDC to –60 VDC	0 VDC to –50 VDC, –72 VDC to –75 VDC

2.4.7 Design and development of 32-channel control system with gigalink

1) Short description of deliverables:

The dedicated pcb boad was designed and fabricated. It is capable of driving 32 piezos. Moreover it can be used for 32 piezos redout. The board can be controlled by other systems using gigalink channel.

2) Current subtask status:

The control system is under development (60%)

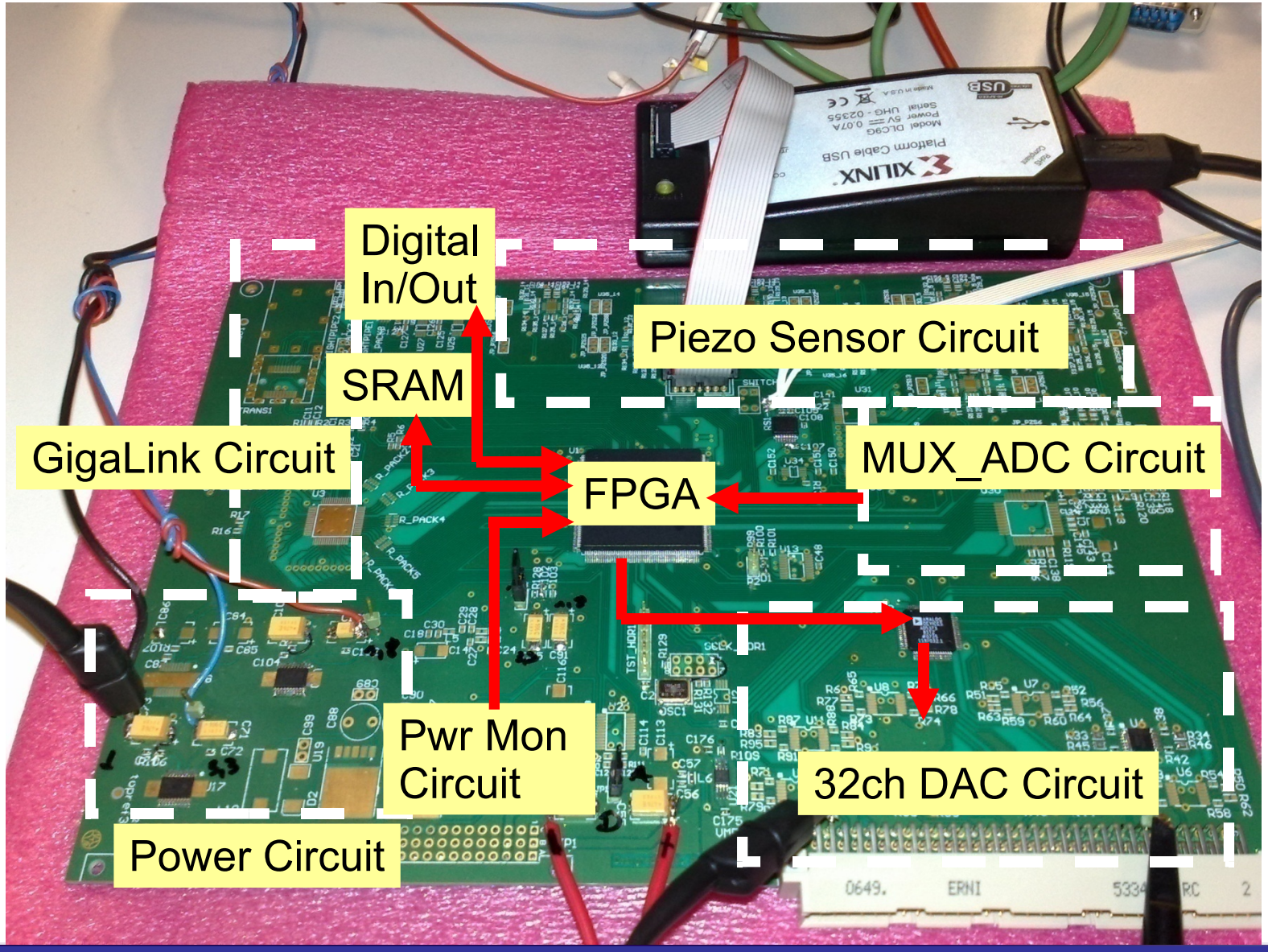
3) Important achievements:

Tests of power supply, FPGA, diagnostic and driving circuits

4) Delivery date, reviewer name, the date when you plan to contact the reviewer:

30.10.2008, M. Grecki, not specified yet

2.4.7 Design and development of 32-channel control system with gigalink



2.4.8 The priliminary tests of whole system before FLASH studies

1) Short description of deliverables:

The dedicated 32-channel control system should be tested with piezo drivers and control sysem i. e. SimconDSP. The communication interface between boards should be developed (connection between rocketio and gigalink serdes TLK2501)

2) Current subtask status:

The rocketio core simulations, the 32-channel control core with TLK2501 interface simulations (40%), hardware tests delayed by subtask 2.4.7

3) Important achievements:

-

4) Delivery date, reviewer name, the date when you plan to contact the reviewer:

30.10.2008, M. Grecki, not specified yet

2.4.9 Tests of piezo drivers and control system in FLASH

1) Short description of deliverables:

The prototype piezo control system will be installed in FLASH. It will be capable of driving as well as sensing 32 piezos.

2) Current subtask status:

Delayed by subtask 2.4.7

3) Important achievements:

-

4) Delivery date, reviewer name, the date when you plan to contact the reviewer:

15.12.2008, M. Grecki, not specified yet

2.4.10 Detuning computation algorithm implemented in PowerQUICC III processor

1) Short description of deliverables:

The detuning algorithm was implemented in PowerQUICC III processor. The computation library was implemented inside linux device driver. The laboratory tests of computation latency without DMA were carried out. The IQ components detectors were implemented inside Virtex 5 FPGA. After RF pulse the interrupt is requested by PCIe device and the computations are performed. The computation results can be seen using external DAC device.

2) Current subtask status:

Detunign library for PowerQUICC (80%) – C/C++ code, Matlab code
IQ componentes detectors (80%) – VHDL code

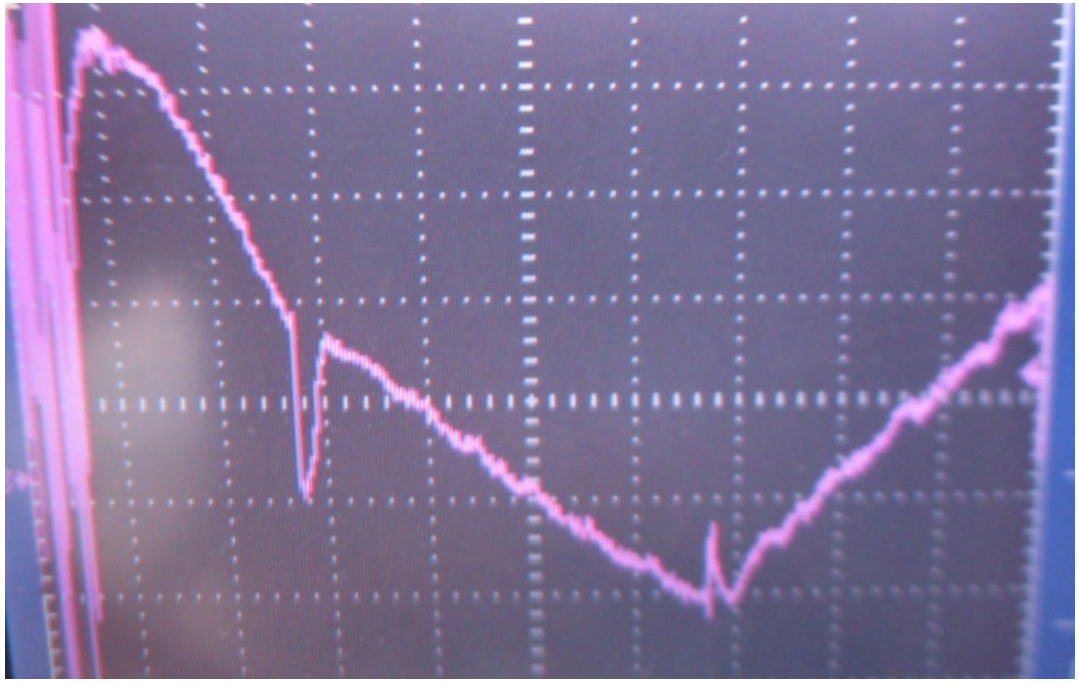
3) Important achievements:

Laboratory tests, latency measurements of PCIe without DMA

4) Delivery date, reviewer name, the date when you plan to contact the reviewer:

30.09.2008, M. Grecki, not specified yet

2.4.10 Detuning computation algorithm implemented in PowerQUICC III processor



operation	latency
write transaction	0.2 μs
read transaction	2 μs
detuning computation (32 cavities)	0.524 sec

```
//detn algorytm start
        detuning(2048, i_p, q_p, i_f, q_f, detn);
// write result to DAC
        addr=0x22000;
        write_pcie_int(addr, 2048, detn);
//interrupt stop
        data = 0;
        addr=0x40004;
        write_pcie_int(addr, 1, &data);
```

Thank You
For Your Attention