

The strategies of data recording in power quality parameters measuring instrument

Romuald Masnicki¹, Janusz Mindykowski², Damian Hallmann³

¹ Gdynia Maritime University, Morska 81-87, 81-225 Gdynia, Poland, romas@am.gdynia.pl

² Gdynia Maritime University, Morska 81-87, 81-225 Gdynia, Poland, janmind@am.gdynia.pl

³ Gdynia Maritime University, Morska 81-87, 81-225 Gdynia, Poland,
d.hallmann@we.am.gdynia.pl

Abstract –In paper the analysis of the two strategies of recording the parameters from electrical power system is presented. The strategies differ in the ways of synchronization between the functional blocks of measuring system. The paper presents the time dependencies between operations carried out during the registration of data from the electrical power system, stored on USB flash drive. The successive stages of data registration are shown, starting from power system signals conversion into digital form, by data gathering and processing in the DSP and then sending it to the FPGA, transferring to memory of the GPP, and finally, storing on USB flash drive. The way of synchronization of data readout from the FPGA to the GPP is crucial for the instrument operation, for the continuity of visualization and recording of data from power electrical system.

I. INTRODUCTION

The Department of Marine Electrical Power Engineering of Gdynia Maritime University designed and made the estimator/analyzer instrument that implements a set of measurement functions which allow to determine the parameters of electrical power quality [1], among others: RMS values of voltage and current, frequency, voltage unbalance factors, interferences, such as harmonic, subharmonics and interharmonics disorders voltage in the range from 50th harmonic to 9 kHz, transient and notching, as well as powers, coefficients of power distribution between the units of parallel operating generators and allocation factors for the evaluation of current overloads the selected generators.

The values of these parameters are made available to the user on the display device, as well as stored in the portable device (SD card or USB stick). The functional properties of the device, developed on the basis of the standard [2], include the registration and storage of data, implemented in certain regimes of time. This allows providing the additional off-line assessment of the results, as well as the verification of algorithms for determining the values of the individual parameters.

Figure 1 shows the structure of the connections between main functional blocks of the instrument.

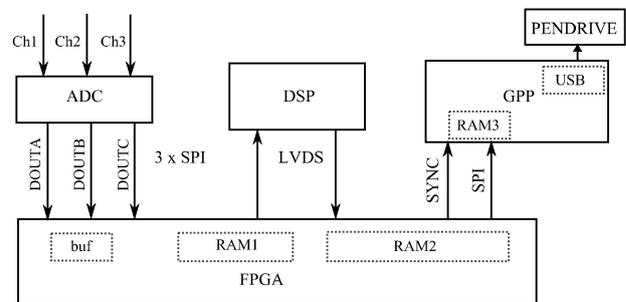


Fig. 1. The functional block diagram of the instrument.

The input circuits of voltage signals from electrical power system supply the input lines Ch1, Ch2 and Ch3 of the ADC (Analog-to-Digital Converter). The ADC unit has been carried out using transmitters AD7656 from Analog Devices [3]. Its function is to convert the instantaneous values of signals from the power system to a digital form. The ADC communicates with other devices via a triple interface SPI (Serial Peripheral Interface) [4]. A single clock from the FPGA (Field Programmable Gate Array) synchronizes the data readout on 3 lines interface (DOUTA, DOUTB, DOUTC). The FPGA acts as the intermediate device between the ports of interfaces with different standards: the ADC converters and processors DSP (Digital Signal Processing) and GPP (General Purpose Processor) [5]. The device uses the FPGA type Xilinx Spartan-3 XC3S1000 [6]. There is possible to configure its I/O lines as symmetrical or unbalanced lines. The functions of DSP are performed by the TS201 TigerSHARC from Analog Devices [6]. It does not have connectivity with external devices using the SPI interface, but it has the ability to connect with the protocol LVDS (Low Voltage Differential Signaling) [7, 8]. The processor LPC3250 (ARM9 family) on the base board phyCORE-LPC3250 from NXP [9] was used as the GPP. This processor has a standard access to peripherals, such as SD card reader (Secure Digital), USB

(Universal Serial Bus), Ethernet controller, LCD display (Liquid-Crystal Display) as well as the interfaces: UART (Universal Asynchronous Receiver and Transmitter), SPI and I²C (Inter-Integrated Circuit). There is possible to install the embedded Linux or Windows CE 6.0 operating system. The Linux is currently used in most multimedia devices, such as mobile phones or tablets. Both these systems are not real-time systems, which is associated with limited possibilities of on-line control over the periphery.

The basic operating principle of the instrument is that the samples from ADC are collected for ten cycles of the power network waveforms in DSP memory, where the data are on-line processed and appropriate parameters are calculated, depending on selected measurement function. Every about 200 ms they are sent inside the frames from DSP via FPGA to GPP, where respective data were displayed or registered. This is the option of registration of frames containing the power quality parameters.

The auxiliary option of instrument functioning is the registration of the digital data coming from momentary values of signals from electrical power system. In this option, the power quality parameters are not calculated, the only rough samples are registered for the waveform visualization and further off-line analyze. This is the samples registration option.

These options differ in capacity of frames sent from DSP to GPP. The size of frame for first option depends on actually performed measurement function (and it is not greater than 4 kB). Each function has its own assigned sort of display, where each screen has its own available set of variables. The volume of frame for second option is constant, equal to 4 kB.

II. DATA TRANSMISSION TIMING

In the chain of operations performed between ADC and GPP (Fig. 1), the operation "read and write data" takes place a few times. These operations result from the need of conversion the format of the data, associated with the different properties of interfaces of individual blocks of instrument.

Below, the processing times in individual blocks of instrument are analyzed (for 4 kB data frames sent from the DSP to the GPP).

The time relations of processes, ranging from signal conversion from analog to digital form in the ADC converter (conversion rate of 210 kS/s) to write data inside 4 kB frames to the DSP memory, are shown in Figure 2. The conversion time of the analog signal samples to digital form in the ADC is constant, equal to $t_{ADC} = 3 \mu\text{s}$. Time t_{tr} is the time for collection of data in the FPGA buffer via SPI from three ADC channels: DOUTA, DOUTB, DOUTC. This time is determined by the frequency of clock, synchronizing the data readout. Under device conditions, the t_{tr} time is equal to 943 ns. The t_{ACQ} time of data acquisition in FPGA consists of

conversion time of the instantaneous values of the analog signal to digital form and the time of results transfer to the FPGA (1).

$$t_{ACQ} = t_{ADC} + t_{tr} = 3,943 \mu\text{s} \quad (1)$$

During the next cycle of the ADC conversion the data from the buffers are written to another part of internal memory (BlockRAM) in the FPGA.

The data from BlockRAM memory are sent using LVDS protocol to the DSP during the t_{RAM} (2). In this operation, one 128-bit frame is sent by LVDS interface at the rate of 200 Mb/s.

$$t_{RAM} = \frac{l_p}{v_R} = 640 \text{ ns} \quad (2)$$

where:

l_p - the number of bits stored in the DSP (128 b),

v_R - frame transfer rate from the FPGA to the DSP (200 Mb/s).

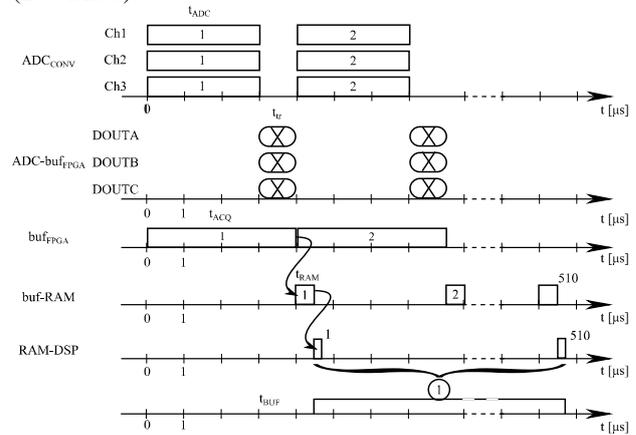


Fig. 2. Timing in data transmission from ADC to DSP.

In memory of the DSP processor, the 4 kB block of data is collected, which consists of 510 frames, each containing 128-bit of data, sent from the FPGA during t_{BUF} (3).

$$t_{BUF} = \frac{n_r}{v_B} = 2,4 \text{ ms} \quad (3)$$

where:

n_r - the number of frames stored in the DSP (510),

v_B - transmission speed of frames from FPGA to DSP, corresponding to the speed of conversion in the ADC (210 kS/s).

The time relations, that occur when transferring data from collection of these data in 4 kB frames in DSP memory until they are written to the memory stick, are shown in Figure 3.

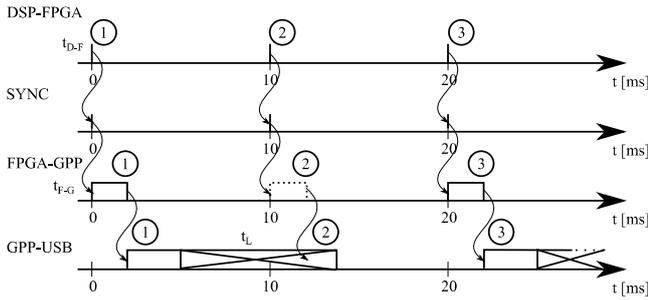


Fig. 3 The data transmission from DSP to USB.

The frame with a volume of 4 kB is sent via LVDS port from the DSP to the FPGA at a rate of 250 Mb/s for t_{D-F} time (4).

$$t_{D-F} = \frac{n_b \cdot l_b}{n} \cdot t_{LVDS} = 131 \mu s \quad (4)$$

where:

n_b - the number of bytes sent in one frame of data (4096 B),

l_b - number of bits per one byte (8 bits),

n - number of bits transmitted in a single clock cycle t_{LVDS} (2),

t_{LVDS} - time to send 2 bits of data (8 ns).

The operation of the data sending from the FPGA to the GPP is carried out under the supervision of the software running in the GPP. The start of transmission can be initiated by the GPP timer or by an interrupt from the SYNC line. The transmission of the 4 kB frame via SPI takes place in the clock cycle at a frequency of about 16 MHz. The frame transmission time, designated as t_{F-G} , is equal to (5):

$$t_{F-G} = n_b \cdot l_b \cdot t_{SPI} = 2,048 \text{ ms} \quad (5)$$

where:

n_b - number of bytes sent in one frame of data (4096 B),

l_b - number of bits in one byte (8 bits),

t_{SPI} - time to send 1 bit via SPI (62,5 ns).

The frame is buffered in the memory of GPP, and then saved to the flash memory connected to the USB port of the GPP, using drivers available in Linux. The envelope mark on the GPP-USB waveform (Fig. 3) means the limits in data writing to flash drive.

While the configuration of data registration channel, starting from the instrument inputs to the FPGA link with the GPP as well as the GPP configuration is no uncertain, the path between the FPGA and the GPP was the subject of additional considerations. These considerations were focused on how to synchronize the data exchange between the FPGA and the GPP. Two strategies of synchronization were considered and verified: software and hardware synchronization of data communication.

III. SOFTWARE SYNCHRONIZATION

The first concept of project assumed the synchronization of data reading from the FPGA to GPP

using the software synchronization of communication process by means of the GPP timer. The timer was set to 200 ms. General operation algorithm is shown in Figure 4. According to this algorithm, at the beginning of device startup the synchronizing thread is started and then the Linux Qt library, dedicated to the operation of software in the GPP, is activated. This thread runs in the background. The GPP software has been divided into several screens, each of them deals with a specific part of the software (according to the actual measurement function), that displays the data on the instrument screen and records it on memory stick.

This loop checks also a state of the keyboard device as well as awaiting the remote command to change the screen (measurement function in DSP) by means of keyboard or remotely by ETH controller.

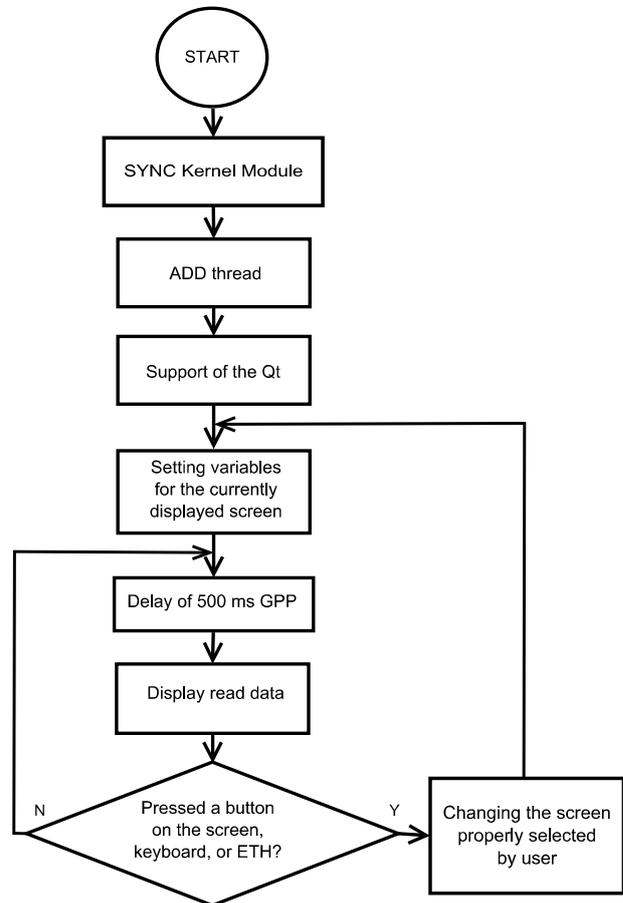


Fig. 4. The algorithm of the main application with the software synchronization.

In Figure 5 the algorithm of the ASYNC thread operations, dealing with reading the data from the SPI, the action of keyboard and a remote control device has been presented. This thread establishes the delay of 200 ms and then the data are read from the FPGA via SPI to GPP. Afterwards, the frame header is searched and data are written into RAM in GPP. In the next step there is

checked, if the keyboard was pressed or any command was read remotely.

Initial concept of acquisition frames assumed that the frame will be collected every ten periods of the sinusoidal waveform with a frequency of 50 Hz. Unfortunately, the grid frequency networks may change slightly, so the data can be overwritten and be incorrect after receiving in GPP in cycles differing from the DSP transmission cycles.

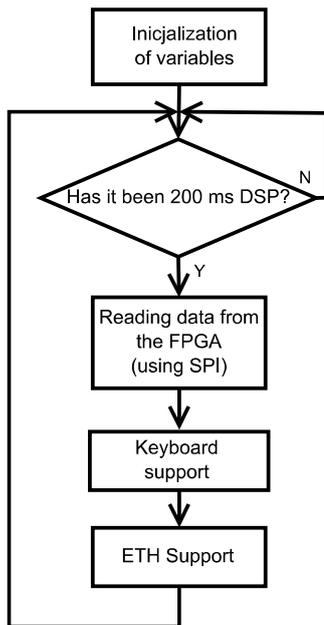


Fig. 5. The algorithm of the software synchronization ASYNC thread, providing the data for the main application shown in Fig. 4.

The example of the measured time irregularity of data read in GPP is presented in Figure 6. The time step between successive frames is changing from the nominal time step, for transmission cycle from DSP equal to 200 ms, by the value Δt in a wide range. It simply results in lost frames.

Table 1 shows the exemplary results of correctness test of data transmission from DSP to GPP with software synchronization. It is worth to notice, that these faults can not be observed on the instrument screen.

Table 1. The exemplary results of software synchronized transmission.

Time T_t [s]	544,559
Present frames number N_r	2097
Nominal frames number N_s	2723
Lost frames number N_l	626
Average read time T_A [s]	0,259

where:

T_t - total time of communication test,
 N_s - nominal number of frames sent,

N_r - received correct frames number,

N_l - lost or faulty frames number,

T_A - average time between frames.

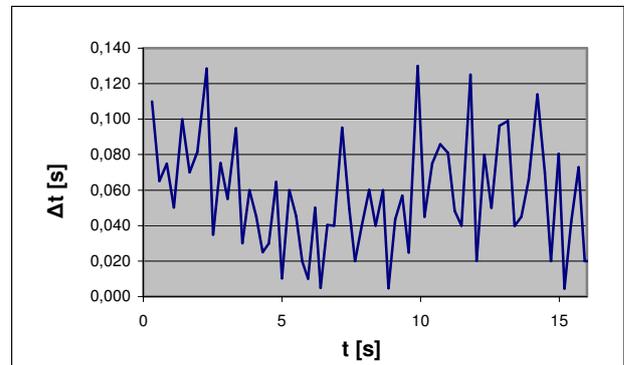


Fig. 6. The dispersion of time of data transmission from FPGA to GPP.

In the result, this concept had to be rejected. The concept did not meet the requirements of both kinds of transmissions: of frames as well as of samples.

IV. SYNC LINE SYNCHRONIZATION

The second concept of recording the data read from DSP (the hardware synchronization) assumes using the SYNC line from FPGA. After the data block from DSP to FPGA is transferred, the SYNC line is set. Therefore, there was developed the additional driver module of Linux kernel for support the SYNC line and the readout of frames from the FPGA via the SPI. This driver supports interrupt triggered when it detects edge on the SYNC line.

The general algorithm of the application is shown in Figure 7. When the instrument is energized and SYNC Kernel Module is loaded into memory, the ADD thread and support the Qt environment are started too. In every iteration, the delay of 500 ms and then the readout of the data frame from the memory kernel module and data visualization are performed. The frames are written to the kernel module memory in tact of the DSP clock (every about 200 ms) and then they are recorded to the memory stick.

After the corresponding data is displayed on the screen, it is checked if there was any request from the keyboard or remote computer (via ETH controller).

Figure 8 presents the algorithms of the Linux SYNC Kernel Module for synchronization of data read and additional ADD thread for the periphery control. The Linux kernel module in the initial stage is designed to initiate all variables and setup of interrupt. The data readout via the SPI is executed only when the corresponding signal on the SYNC line is detected. Then the frame is read via SPI, and if it is set, the data writing

to the flash memory operation is also performed. The main loop of the program works in a similar way as in the first concept of synchronization, with the difference that only the keyboard and commands from remote computer are enabled in main application and performed in the ADD thread, but the SPI as well as the USB interfaces are controlled from the level of system kernel.

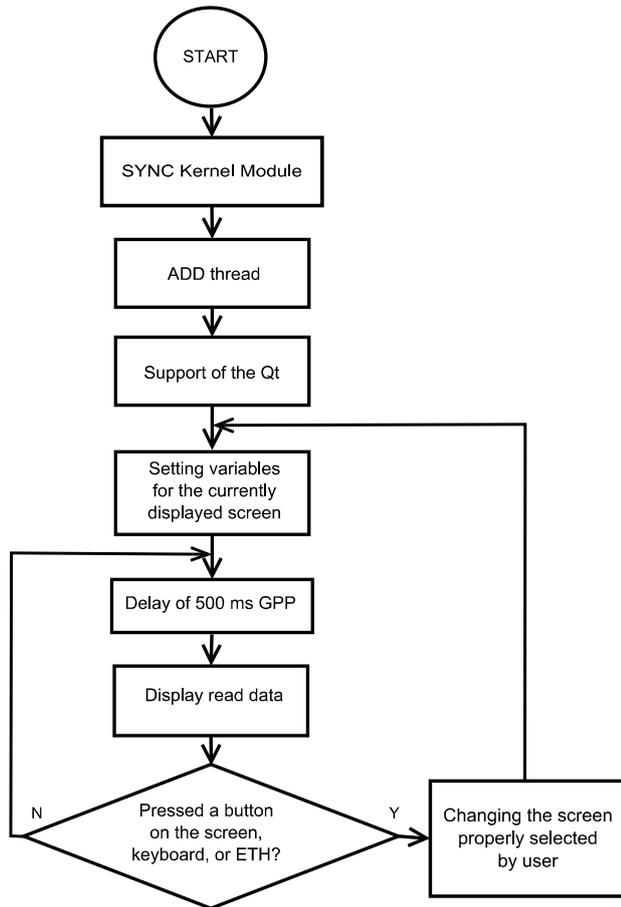


Fig. 7. The algorithm of the main application with the SYNC line synchronization.

This firmware version allows reading the data when they are already stored in the FPGA. This prevents overwriting any part of the data frame. The situations as illustrated in Table 1 can never happen under hardware synchronization.

The hardware synchronization of data receiving in the GPP enables correct registration both frames of data with power quality parameters elaborated in the DSP and raw samples of signals from electric power network. The time limit appears during write operation from the GPP memory to the flash drive via the USB port. The limit concerns only the writing big volume of data connected with samples of three phase voltages with high speed of analog-to-digital conversion at the rate of 210 kS/s. The other measurement task can be performed without this limit.

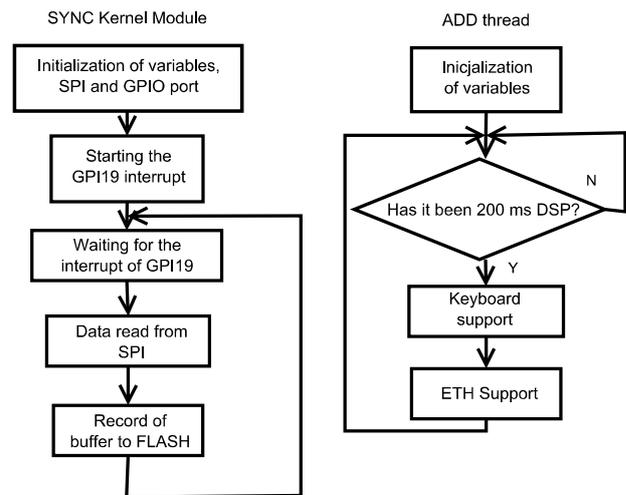


Fig. 8. The algorithms of the SYNC Kernel Module and the ADD thread, co-operating with the main application shown in Fig. 7.

V. CONCLUSIONS

As a rule, the manufactures do not reveal of the internal structure or measurement algorithms of offered instruments for power quality measurement. The analysis of the basic functional properties of similar instruments is available in [10].

The paper presents the two different ways of synchronization of data received from the DSP through the FPGA to GPP. At constant time condition for operations in individual blocks of instrument, the way of data synchronization between FPGA and GPP is essential for instrument functionality.

The software synchronization allows checking the elapsed time since the last transmission. This kind of synchronization did not give the ability to read the data when they actually were buffered in the FPGA and ready to read. The frames read in GPP were often overwritten or lost because of real lack of control of timing of peripheral processes under Linux operating system. As shown in Figure 5, the SPI handling, the keyboard and the Ethernet access operations have been concluded in the common thread. This could be the cause of additional delays to the GPP timer delay (200 ms), when data was written to the GPP.

Through the hardware synchronization, by using the SYNC line, the reading from the FPGA has been made possible when the data were actually written there. This is to avoid overwriting of data as well as effectively improve the data transmission speed to the GPP. This type of synchronization forced the authors to develop the Linux kernel software and write the own driver that uses the interrupt from the SYNC line. By separating the SPI handling and the read and write memory stick operations from the rest of the software available in the thread it has become possible to implement these processes more

frequently than every 200 ms. This fact has been used to develop the additional features available in the instrument.

The paper presents an analysis of time relationships in a recording system of signals from the electrical power system. The data transfer time from the end of conversion of analog signals to digital form until the write the digital data to the memory stick were analyzed. The time of sending a single frame from the ADC to the DSP and further to GPP is relatively small compared to the time to write data in the memory stick.

The time required to write the data to a memory stick under Linux operating system is difficult to clearly identify. This environment is multithreaded, so the processes that run in the background are activated depending on the priority that has been them granted.

The authors carry out further work on the option for shorter acquisition times and write data to external media device. They are conducted towards the elaborating of the USB port driver, which enables the data recording at a shorter time than heretofore, and brings the ability to record more data to be sent from the DSP to the GPP via FPGA.

REFERENCES

- [1] Mindykowski J., Mańnicki R., Hallmann D., The implementation of FPGA for data transmission between ADC and DSP peripherals In the measurement channels for Power quality assessment, ACTA IMEKO, 2013, Vol. 2, No 1 (2013) pp. 49-55.
- [2] International Standard IEC 61000-4-30, Electromagnetic compatibility (EMC) – Part 4-30: Testing and measurement techniques – Power quality measurement methods, 2003.
- [3] AD7656/AD7657/AD7658, Data sheet, Rev. D, Analog Devices, 2010.
- [4] Analog Devices, Interfacing to High Speed ADCs via SPI, Application Note AN-877, Rev. A 12/05.
- [5] Thanee S., Somkuarnpanit S., Saetang K., FPGA-Based Multi Protocol Data Acquisition System with High Speed USB Interface, IMECS 2010, Hong Kong, 2010.
- [6] Xilinx. Spartan-3 FPGA Family Data Sheet, Product Specification, Application Note DS099; 2005.
- [7] Xilinx. Analog Devices TigerSHARC Link Port, Application Note: Spartan-II and Spartan-3 Families, Virtex and Virtex-II Series, XAPP634 (v1.2), 2004.
- [8] National Semiconductor, LVDS Owner's Manual, 4th Edition, 2008.
- [9] PHYTEC, System on Module and Carrier Board Hardware Manual, January 27, 2009.
- [10] Mindykowski J., The universal analyzer-estimator of electrical power quality. Part I. The origins and the concept of a new device, Scientific Journal of Gdynia Maritime University, Nr 66, pp. 7-19, Gdynia, 2010 (in Polish).