

THE SYNCHRONIZATION OF PROCESSES IN REGISTRATION TRACK OF DATA FROM ELECTRICAL POWER SYSTEM

*Romuald Masnicki*¹, *Janusz Mindykowski*²

¹ Faculty of Electrical Engineering, Gdynia Maritime University, Gdynia, Poland,
romas@am.gdynia.pl

² Faculty of Electrical Engineering, Gdynia Maritime University, Gdynia, Poland,
janmind@am.gdynia.pl

Abstract – In paper, the synchronization of operations performed in registration track of the parameters from electrical power system is discussed. The synchronization operations are carried out in the track during communication between functional blocks, on several levels of these operations (bit, frame and information synchronization). The successive stages of data registration track in instrument are shown. The different ways of frame synchronization in transmission between selected blocks and some obtained effects are presented.

Keywords: data registration, interface, synchronization.

1. INTRODUCTION

Ensuring the proper transmission of data (transmitted through signals) between devices or their functional blocks requires to synchronize the receiver of data to the transmitter. Synchronization is time coordination of at least two processes, namely the pursuit them of parallel, independent of their course, coordinated in time.

Most naturally occurring physical quantities (e.g. pressure, temperature, change the angle, lighting) are continuous, i.e. analog. There are two different ways to transfer of analog information through communication interfaces: analog transmission (the information is stored directly in the parameters of the signal) and digital transmission (the information is converted into digital form first, followed by the assignment certain signals to digital data).

The synchronization in analog interface (e.g. current loop 4..20 mA) is based on assumption of the simultaneity of signals on the side of the transmitter and receiver.

For typical applications of such interfaces, this assumption is acceptable because of the low dynamics of the relevant processes, and thus, slow changes (large time constants) of the interface signals, carrying information about these processes.

Electrical signals are propagated in copper cable at a speed of the order of $2 \cdot 10^8$ m/s, which means that the signal sent from the transmitter to the receiver 100 m away, appears in receiver after a time of approx. 0.5 μ s. This delay is not important in many industrial processes, such as temperature control. However, analog interfaces also have other limitations, like the lack of "memory" of information. Information prepared to send and transmitted from the

transmitter, if does not reach the receiver (e.g. in case of a disruption in the interface), can not be recovered and re-sent.

Digital interfaces send binary data that can be easily stored in the source device, and thus can be restored in case of loss during transmission to the receiver. The synchronization process for the transmission of digital data usually takes place in several stages:

- receiver clock synchronization with the received signal transmitting digital data stream - bits (bit synchronization - implemented at the physical layer of interface),
- frame synchronization (data link layer),
- synchronization of information (application layer).

The bit synchronization enables accurate detection of consecutive bits of the frame thanks to their identification in signal at the right point in time.

In the asynchronous interfaces, synchronization of the receiver clock is obtained based on the slope of the received signal carrying a start bit. The receiver clock is set to the same frequency as the transmitter clock. In the receiver, when it detects a signal edge, the phase of the receiver clock is tunable, respectively, and then the receiver clock become the reference to determine the moments for decoding the bits from carrying signal, in relatively short frames transmitted via an asynchronous interface. The limited length of a frame results from the limited stability of the receiver clock frequency with respect to the frequency of the clock, in fact the received bits were transmitted.

The receiver clock in synchronous interface is synchronized directly with the clock of transmitter. In the interfaces for short-range (when the delay of signals in interface lines are negligible in relation to the period of the clock signal), the clock synchronization can be carried out by the appropriate device (e.g. in SPI (Serial Peripheral Interface)). In the interfaces of the long-range, the synchronizing signal source is usually a transmitter data clock (because of the delay in lines, almost identical to the data and the clock signal). A clock signal can be transmitted via the line independent of a data line or via a common line, together with the data, using selected the so-called self-synchronizing code.

Frame synchronization is necessary for the proper recognition of the start, end and the other parts of the continuous sequence of bits that make up the frame. It takes place after the bit synchronization (bit synchronization error, e.g. loss of bit of data, is causing the synchronization error

of the frame). Frame synchronization is usually achieved by identifying a preamble contained in the introductory part (characteristic string that, with high probability, will not appear in the rest of the frame).

Another problem is the initiation of frame transmission: it can be started by software or hardware. Later in the paper, the results of efficiency of data transmission between two blocks of instrument, using frame synchronization by software as well as by hardware, are shown.

The information synchronization is based on the purposeful and proper use of the data transmitted in frames, corresponding to the functions performed in the system.

The paper is organized as follows. The general assumptions concerning the synchronization in digital measurement track are formulated. Next, the configuration of estimator/analyzer instrument and some aspects of synchronization of operations performed in the track are shortly described. In the following step the results of implementation of hardware in contrast to software synchronization in data exchange between two blocks are discussed. Finally, the conclusions are drawn and commented on.

2. THE OPERATIONS PERFORMED IN TRACK OF DIGITAL DATA REGISTRATION

In digital systems, implementing the analog quantity measurements and registration of the results, the typical functional blocks can be specified. The configuration of the system mainly depends on the complexity of fulfilled functions and the amount of processed data, but also on the communication capabilities of essential circuits applied in measurement system.

Figure 1 shows a typical configuration of measuring tracks of the analog quantities. Analog information collected from the input circuits (e.g. voltage dividers, transformers, transducers) is fed to the input of the ADC (Analog-to-Digital Converter). The configuration in Figure 1.a is used in the measurement tracks, in which the processor GPP (General Purpose Processor) may be coupled directly to the ADC interface ports using the same standards, available in both systems.

Processor GPP keeps up with the processing of data from the ADC, and also supports integrated user interface UI (user interface), both in terms of visualization of the results and their registration. GPP also supports other tasks, e.g. commands issued by the user.

Synchronization of operations in the measurement track involves the provision of fluent data transmission between the ADC and the GPP and its management in the UI system. The sequentiality of operations in different blocks of the measurement track is associated with the fact that in order to maintain the fluency, the total execution time (t_{GPP} and t_{GPPout} in GPP and t_{UI} in UI) can not exceed the time to prepare a new set of data t_{ADC} at the output of the ADC (i.e. the processing time of the A/D in ADC) and the time t_{ADCout} needed for them to be sent to the external system.

$$t_{ADC} + t_{ADCout} \geq t_{GPP} + t_{GPPout} + t_{UI} \quad (1)$$

The times of individual operations rely on the complexity of a given operation in the individual blocks and

the characteristics of the communication interfaces connecting the functional blocks.

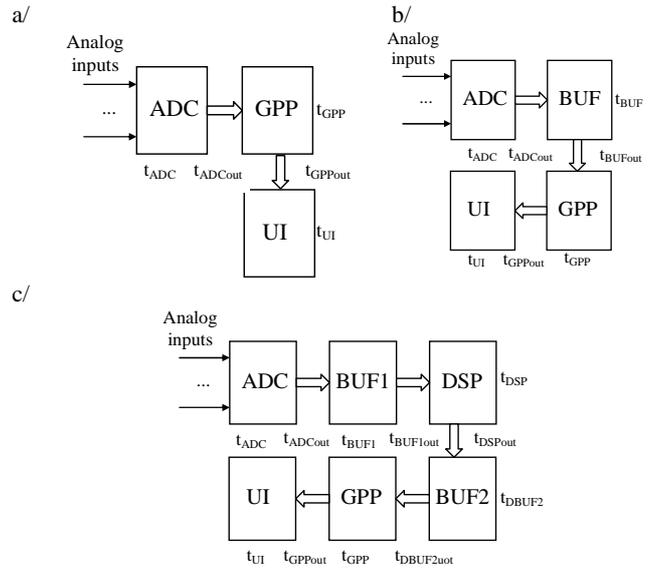


Fig.1. The configurations of the tracks for registration of analog data.

A similar range of complexity of the operations in measurement track can execute a system whose configuration is shown in Figure 1.b. Because the ADC and GPP devices are equipped with different communication interfaces, the additional circuit BUF is therefore required, to perform the conversion between standard interfaces (in the physical layer and in the data link layer), and for buffering and processing of measurement data. In order to maintain the fluency condition of data flow in measurement track the condition must be met:

$$t_{ADC} + t_{ADCout} \geq t_{BUF} + t_{BUFout} + t_{GPP} + t_{GPPout} + t_{UI} \quad (2)$$

The configuration of the measurement track, shown in Figure 1.c, is usually used when the complexity and number of operations related to the processing of data exceeds the capacity of a typical GPP processor and requires the use of a DSP (Digital Signal Processor). Typical DSP is the processor, whose architecture is oriented towards rate of operation and the amount of data processing, with a relatively poor set of peripheral devices. GPP, in contrast to the DSP, typically has a rich set of communication ports in different standards. Just as BUF in the system shown in Figure 1.b, BUF1 and BUF2 blocks in Figure 1.c implement standards conversion between interfaces as well as buffering and processing of data. The fluency condition of data flow includes the times of additional operations:

$$t_{ADC} + t_{ADCout} \geq t_{BUF1} + t_{BUF1out} + t_{DSP} + t_{BUF2} + t_{BUF2out} + t_{GPP} + t_{GPPout} + t_{UI} \quad (3)$$

3. THE SYNCHRONIZATION OPERATIONS IN THE ESTIMATOR/ANALYZER INSTRUMENT

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 determining the parameters of electrical power quality [1]. The values of these parameters are made available to the user on the display device, as well as stored in the portable device (SD (Secure Digital) card or USB Flash Drive).

Figure 2 shows the structure of the connections between main functional blocks of the instrument. The figure shows only the voltage tracks, for simplicity the currents measuring lines were omitted. Additionally, the procedures of voltage data processing in instrument are much more complicated. The complexity of the instrument configuration corresponds to the configuration of the track shown in Figure 1.c.

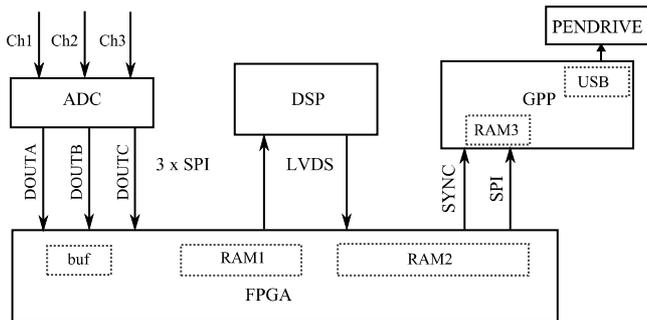


Fig. 2. The functional block diagram of the instrument [1].

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 AD7656 converters (Analog Devices) [2]. The ADC can output data through a three-line-interface SPI. A single clock from the FPGA (Field Programmable Gate Array) synchronizes the data readout on 3-line interface (DOUTA, DOUTB, DOUTC). The FPGA acts as the intermediate device between the ports of interfaces in different standards: the ADC converters and processors DSP (Digital Signal Processing) and GPP (General Purpose Processor) [3]. The instrument uses the FPGA type Spartan-3 XC3S1000 (Xilinx) [4]. The functions of DSP are performed by the TS201 TigerSHARC (Analog Devices) [5]. It does not have connectivity with external devices using the SPI interface, but it can be connected using the LVDS (Low Voltage Differential Signalling) interface [5]. The processor LPC3250 (ARM9 family) on the base board phyCORE-LPC3250 (NXP) [3] was used as the GPP. This processor has a standard access to peripherals, such as SD card reader, 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). GPP can work under control of the embedded Linux or Windows CE 6.0 operating system. Both these systems are not real-time systems, which is associated with limited possibilities of on-line control over the periphery.

The synchronization of interface between ADC and FPGA buffer is controlled by the 17 MHz clock signal from FPGA. The triple synchronous SPI interface transmits the data in tact of clock pulses. Their generating starts when the A/D conversion of samples set is completed and is signaled to FPGA.

After collection of 3-words, 16-bit each, of data in FPGA buffer, they are converted to the form possible to be transmitted to DSP and placed in the RAM consisting of 8 sets of memory cells.

The transmission from FPGA to DSP is controlled by the 100 MHz clock generated in FPGA. The synchronous 4-line LVDS interface starts operation after the data in FPGA RAM are completed and DSP interface port is ready to receive the data.

The measurement algorithm, implemented into DSP program, is taking in real-time received digital samples of voltage and calculating online respective parameters, for the samples collected for a given number of periods of signals from electrical power system.

The transmission from DSP to FPGA is controlled by the 125 MHz clock generated in DSP. The synchronous one-line LVDS interface starts operation after data in DSP are ready to send.

After collection of set of data frames in the FPGA 2-side access memory buffer, the data are ready for transmission to GPP. The transmission from FPGA to GPP is controlled by the clock generated in GPP. The synchronous two-line SPI interface transmission begins when the signal clock from GPP appears.

The basic operations principle of the instrument is that the samples from ADC are collected for ten cycles (for 50 Hz network; 12 cycles in case of 60 Hz system) 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 main option of instrument: the registration of 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.

In [1], the time dependencies between operations carried out during the registration of data from the electrical power system, performed in instrument under consideration, were discussed. The time relations of processes, ranging from the analog signal conversion in the ADC, through writing data to the DSP memory, up to sending them to GPP and finally to USB Flash Drive were there presented.

Table 1 contains the summary list of major time components occurring in registration track of instrument.

According to instrument operations, the requirement (3) has to be divided into two requirements: first, concerning timing on the way from ADC to DSP (part A of Table 1), and second, referring to the timing on the way from DSP to GPP (and USB Flash Drive) (part B of Table 1).

First requirement concerns the rate of samples conversion in ADC and data output to DSP. It is fulfilled. For on-line data processing operations in DSP, before next set of data from ADC is received, it remains for DSP operations about 3,3 μ s.

Table 1. Summary list of main operations timing in instrument registration track.

A	ADC conversion and output time	3,95 μ s
	FPGA processing and sending to DSP time	0,66 μ s
	Time remaining for DSP operations	3,29 μ s
B	Buffering time of 4 kB set of samples in DSP	2,43 ms
	Transfer time of data from DSP to GPP	2,18 ms
	Time remaining for data write to USB Flash Drive	0,25 ms

Second requirement is connected with buffering of 4 kB block of data in DSP. During this time, the previous set of data has to be output from DSP to FPGA, and than to GPP and finally registered. As it is shown in Table 1, for write to portable memory of 4 kB data block remains about 0,25 ms.

While the configuration of data registration channel, starting from the instrument inputs and ADC, through the FPGA link with the DSP as well as the DSP configuration, is reliable, 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 approaches to the problem of frame of synchronization were considered and verified: software and hardware synchronization of data communication.

The first concept of the 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. After such period of time 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.

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

The example of the measured time irregularity of data read in GPP is presented in Figure 3. 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. As a 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.

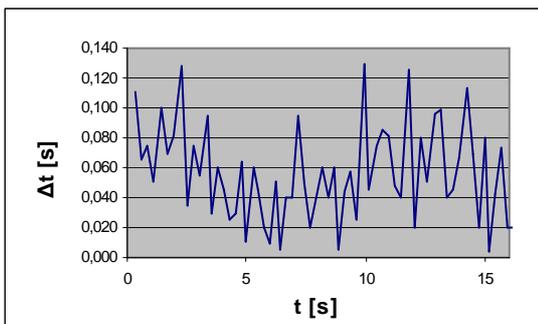


Fig. 3. The dispersion of time of data transmission from FPGA to GPP [1].

4. CONCLUSIONS

The paper presents the problem of synchronization of operations performed in data registration track of estimator/analyzer instrument.

The basic condition in any operations connected with data transmission is reliable bit synchronization. This requirement is fulfilled implicitly in every operation in registration track.

The frame synchronization is implemented in the registration track mostly in hardware, by means of selected control lines. This solution proved to be reliable. The only communication between FPGA and GPP was tested for both types of frame synchronization: by software firstly and then by hardware synchronization.

The synchronization by software 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. This way the frames were often overwritten or lost because of real lack of control of timing of peripheral processes in GPP.

Through the synchronization by hardware (using the SYNC line - in Fig. 2) 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 rate from DSP to GPP. This synchronization method has proved to be a key for whole instrument functionality.

REFERENCES

- [1] J. Mindykowski, R. Mańnicki, D. Hallmann, "The strategies of data recording in power quality parameters measuring instrument", *20th IMEKO TC4 International Symposium and 18th International Workshop on ADC Modelling and Testing*, pp. 635-640, Benevento, Italy, 2014.
- [2] AD7656/AD7657/AD7658, Data sheet, Rev. D, Analog Devices, 2010.
- [3] PHYTEC, System on Module and Carrier Board Hardware Manual, January 27, 2009.
- [4] Xilinx. Spartan-3 FPGA Family Data Sheet, Product Specification, Application Note DS099; 2005.
- [5] Xilinx. Analog Devices TigerSHARC Link Port, Application Note: Spartan-II and Spartan-3 Families, Virtex and Virtex-II Series, XAPP634 (v1.2), 2004.
- [6] National Semiconductor, LVDS Owner's Manual, 4th Edition, 2008.