

Maciej Rosół*, Dawid Knapik*, Dariusz Marchewka*,
Krzysztof Kołek*, Andrzej Turnau*

Analysis of Industrial PROFINET in the Task of Controlling a Dynamic System**

1. Introduction

The paper analyses the behaviour of industrial PROFINET for time determinism during the transfer of controls and measurements in real time.

The acquisition and analysis of network packets are performed using the Wireshark. The subject of interest is the performance of the network control system. An access to the packets is achieved by the port mirroring which is a function of the managed switch. Test results demonstrate the usefulness of communication by the PROFINET industrial-standard that successfully implements control tasks and maintains time determinism of these tasks.

2. PROFINET as the standard

PROFINET is the communication standard dedicated to automation. It was developed in 2003 as a successor to the PROFIBUS protocol (1989) [1]. There are two PROFINET variations: PROFINET IO and PROFINET CBA. Both are based on the Ethernet protocol. PROFINET CBA allowing for the definition of logical assignments of equipment and control-measuring equipment and the adoption of the resulting groups as technological modules. The resulting technology modules in the network based on the PROFINET CBA are seen as network components, and the protocol itself enables the exchange of information between them. The principle of operation is based on the DCOM mechanism which originates from OLE. The data transfer time is on the order of hundreds of microseconds.

* AGH University of Science and Technology, Krakow, Poland

** This work was partially supported by AGH University of Science and Technology Research Grant 11.11.120.768

PROFINET IO describes the communication as well as a layer between the hardware control system (e.g. PLC master) and peripheral devices (e.g. slave sensors and actuators). A core of the standard is to maintain the real-time exchange of information between devices. Depending on the desired properties of the PROFINET IO is divided into three classes (conformance classes). CC-A class provides the basic functionality and the C-C is the most advanced class (Fig. 1).

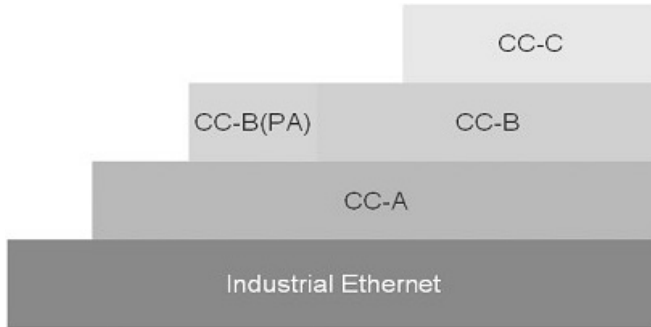


Fig. 1. The breakdown into PROFINET protocol classes.
Retrieved from PROFINET System Description

Within the Class CC-A the Real-Time protocol is implemented (also called Soft RT). It allows exchanging data in a cyclic manner with the lengths of the cycles of microseconds or more. According to information published on the foundation supporting PROFINET protocol [2], the cyclic data exchange could be realized by using classical network infrastructure such as switches. It is recommended to use switches with the support of VLAN Tag prioritization described in IEEE802.1Q. PROFINET RT frames have a high priority. PROFINET RT runs directly on the other layer ISO/OSI model of the protocol identifier (called EtherType) 0x8892. An important feature of the protocol is the lack of the use of protocols/data TCP (UDP)/IP. The CC-B class extends the CC-A class capabilities adding diagnostic tools and mechanisms for the topology discovery (SNMP, MIB2, LLDP-EXT MIBPDEV). Additionally, within the class, the CC-B(PA) extension is designed for redundant systems.

The CC-C class, also known as Isochronous Real Time (IRT) is designed for applications with the highest requirements for network determinism. Communication in this class allows for jitter of less than 1 microsecond and the exact synchronization devices. The minimum time for data exchange is 31.25 ms. The use of the CC-C class requires the application of special devices with dedicated Ethernet controllers (e.g. ERTEC-Enhanced Real-Time Ethernet Controller manufactured by SIMENS). The dedicated controllers reserve a part of the network bandwidth for the real-time communication. The remaining bandwidth allows for the use of classical protocols PROFINET making the CC-C class compatible with the standard Ethernet.

A times less than 1 ms refresh rate, a low jitter and a few microseconds synchronization allows to apply PROFINET RT in demanding applications such as a high-performance drive control.

Figure 2 shows a typical PROFINET frame construction. The frame fields relevant to the analysis of the network control system are as follows: “Frame ID” indicating the real-time PROFINET class, “Cycle Counter” containing the internal clock cycle a number in the numerator of cycles (a single period is 31.25 ms), “Data Status” and “Transfer Status” which confirm the transmission correctness.

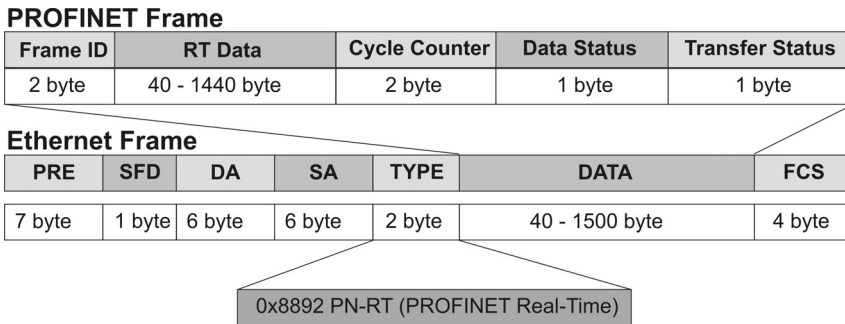


Fig. 2. The PROFINET frame

3. Acquisition and analysis of the Wireshark network packets

The acquisition and analysis of the network packets is performed using the Wireshark program. It is a comprehensive program to analyze network traffic and is able to capture and identify the packets sent by hundreds of communication protocols (including used PROFINET IO). Built-in application filters and statistics allow for the analyses of selected frames and a communication link. Figure 3 shows the main window of the Wireshark program. It contains a description of the package data for the PROFINET IO protocol.

Due to the monitoring function in the managed switch (known as “port mirroring”) a copy of network packets from an arbitrary port of the switch to another port is sent without disturbing the original motion. It should be noted that the main emphasis of research put on the analysis of the control system network efficiency, and not on the parameters of the controlled object response.

A deviation – called jittering – between the measured sample period $T_{0,n}$ and the set (desired) T_0 is evaluated. Moreover a parameter $\Delta\tau_n$ calculated as a difference between the actual and assumed time to send the package that contains the control value. The method of calculation of these parameters is shown graphically in Figure 4. It is clear that the time moments $t_1, t_2, \dots, t_n, \dots$ are read in Wireshark (column „Time”) for S7-1200 transmission → SINAMICS S110 which contains the control value (frame identifier ID = 0x8020).

Fig. 3. The main window of the Wireshark program analyzing the PROFINET frame

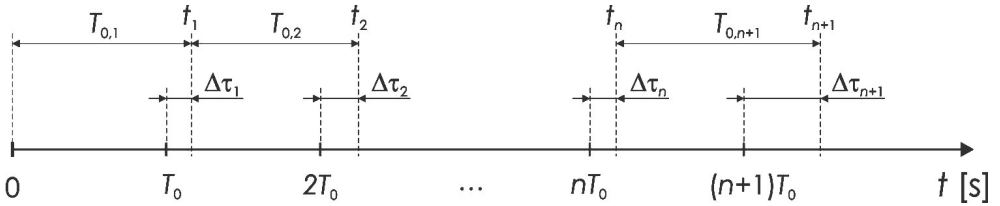


Fig. 4. The method of calculating the transmission parameters

The value of the parameter $\Delta\tau_n$ can be calculated as (1).

$$\Delta\tau_n = nT_0 - t_n \quad (1)$$

where:

nT_0 – the assumed time to send the n^{th} packet,

t_n – the measured time to send the n^{th} packet,

$\Delta\tau_n$ – time deviation between the measured and assumed times to send the n^{th} packet.

Parameter values $\delta T_{0,n}$ are obtained from equations (2) and (3).

$$T_{0,n} = t_{n+1} - t_n \quad (2)$$

$$\delta T_{0,n} = T_0 - T_{0,n} \quad (3)$$

where:

- t_{n+1} – the measured time to send the $(n+1)^{\text{th}}$ packet,
- t_n – the measured time to send the n^{th} packet,
- $T_{0,n}$ – the calculated sampling period,
- $\delta T_{0,n}$ – the calculated deviation of the sampling period $T_{0,n}$.

4. Experiments

Experiments were carried out on a two-node network control system, in which the first node provides measurements and performs control algorithm for the cart and pendulum system, and the second node works as a remote control servo. The first node is a PLC S7-1200 SIEMENS and the second is the servo drive SINAMICS driver. The transmission uses standard Ethernet infrastructure and protocol PROFINET IO CC-B (see Chapter 2). To monitor network traffic the managed switch is applied (which is D-Link DES-3010F) and PC computer working under Ubuntu 11.04 with the installed Wireshark program is used. To study and to analyze the network control system behaviour the pendulum on a cart laboratory model is selected, see Figure 5.

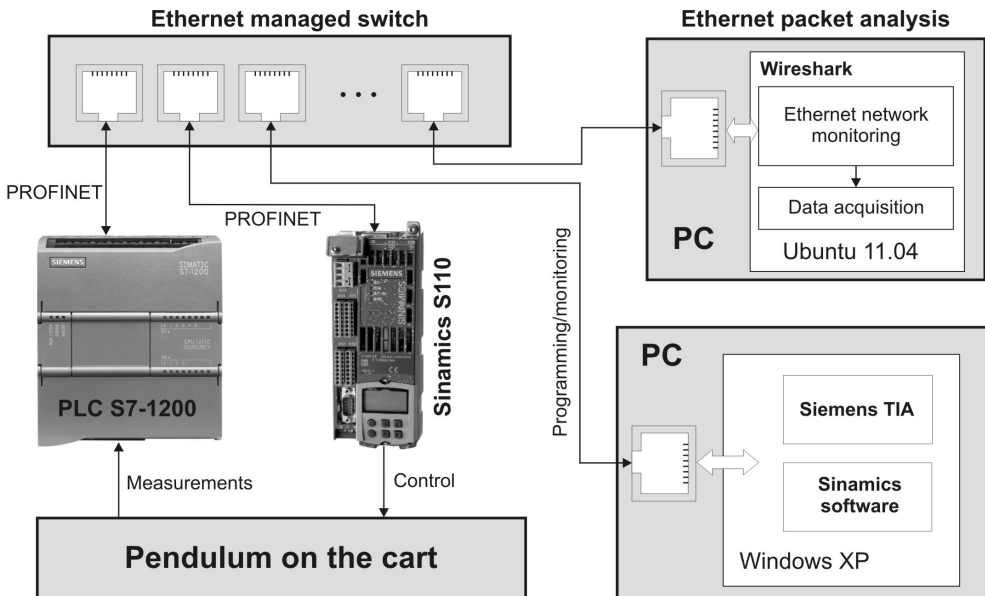


Fig. 5. Schematic of laboratory set

The test procedure consists of the following steps:

1. Determining the period T_0 to send the PROFINET frames containing the control value to the SINAMICS driver.

2. Specifying the additional network traffic, which increases the load on the Ethernet link. The load is increased by sending a certain number of data packets at fixed moments of time (the ping command).
3. Running control tasks.
4. Collecting in the Wireshark program all of the Ethernet packets.

In Figures 6 and 7 the results of the surveys in the form of the time deviation $\Delta\tau_n$ and histogram of the deviation from the sampling period δT_0 obtained for $T_0 = 2$ ms are shown. In the moments of time 10 s, 40 s 70 s the network traffic was increased by sending an additional 10 packets containing 1024 bytes each.

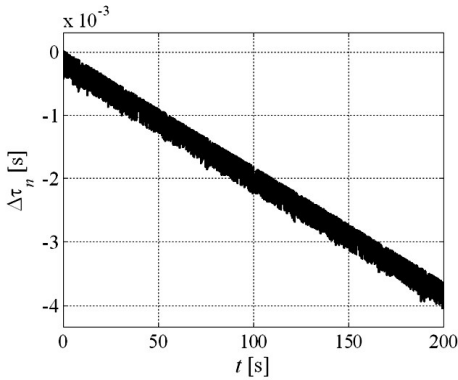


Fig. 6. $\Delta\tau_n$ vs. time for $T_0 = 2$ ms

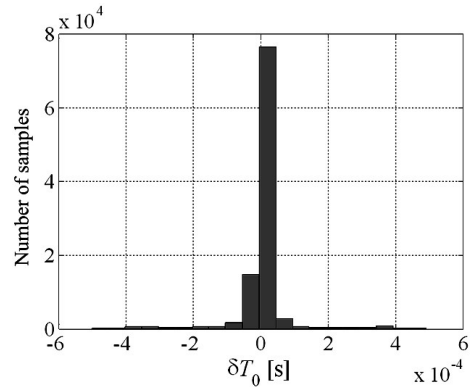


Fig. 7. Histogram of ΔT_0 for $T_0 = 2$ ms

The average value of the calculated sampling period is 2.0000365 ms. The maximum deviation from the average value is 501.9635 μ s, which gives the maximum value of δT_0 equal to 25%. The network traffic generated as an extra burden does not affect the value of the obtained parameters (Fig. 8). As can be seen in Figure 6 the parameter value $\Delta\tau_n$ decreases over time while maintaining a linear trend. This means that the time difference between the measured and the assumed time of sending the package is increased by the value of -18.23 μ s/s.

In Figures 10 and 11 the waveforms $\Delta\tau_n$ and histogram δT_0 obtained for $T_0 = 8$ ms are shown. Other conditions of the experiment are consistent with those established for $T_0 = 2$ ms.

The average value of the sampling period is equal to 8.0001489 ms. The maximum deviation from the average value is equal to 406.8511 μ s, which gives the maximum value of δT_0 equal to 5.09%. Similarly to the previous case the network traffic generated as an extra burden does not affect the value of the obtained parameters (see Fig. 9).

The parameter value $\Delta\tau_n$ also decreases over time while maintaining a linear trend. It is increased by the value of ~ 18.64 μ s/s (see Fig. 10).

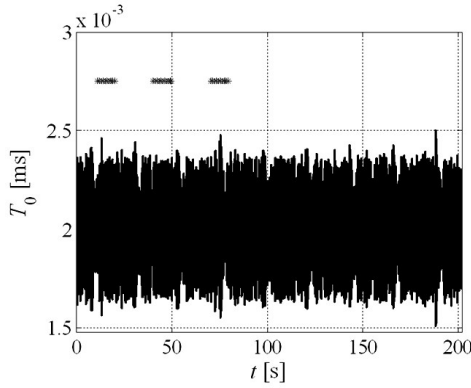


Fig. 8. $\Delta\tau_n$ vs. time for $T_0 = 2$ ms

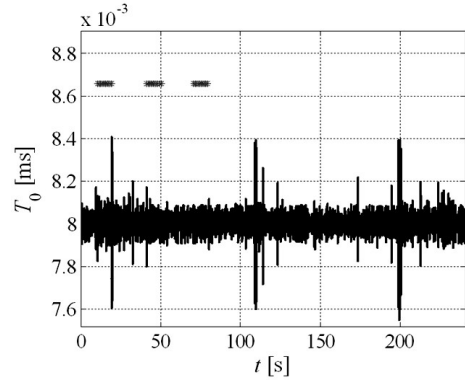


Fig. 9. $\Delta\tau_n$ vs. time for $T_0 = 8$ ms

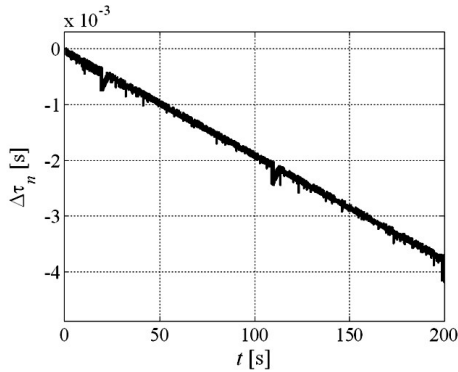


Fig. 10. $\Delta\tau_n$ vs. time for $T_0 = 8$ ms

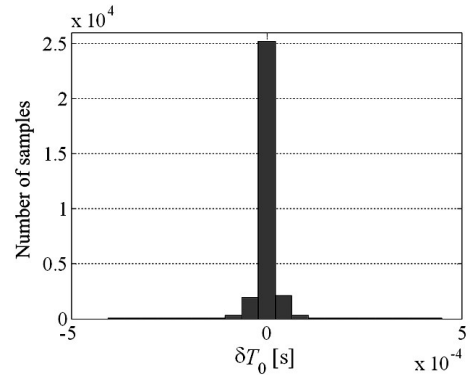


Fig. 11. Histogram of δT_0 ($T_0 = 8$ ms)

5. Conclusion

The linear trend $\Delta\tau_n$ observed in Figures 6 and 10 with a value of about $-20 \mu\text{s/s}$ is present due to the quartz oscillators errors. These oscillators operate as clock signals for PLC and computers that perform measurements while using Wireshark. Typical stability of quartz resonators is tens of ppm which is consistent with the observed value.

The jitters observed for both test sample periods are equal to $500 \mu\text{s}$. They locate the test configuration in the central area of the real-time control system [3]. On the one hand, real-time operating systems provide a jitter of a few-dozen microseconds, on the other hand, control systems built on the basis of general-purpose operating systems (MS-Windows or Unix) generate a jitter value of several tens of milliseconds.

References

- [1] PROFIBUS Nutzerorganisation, PROFINET System Description – Technology and Application, 2011.
- [2] <http://www.profibus.com> (PROFIBUS & PROFINET International webpage).
- [3] Kołek K., Turnau A., *FPGA as a part of MS Windows control environment*. Computer Science: year book of the AGH University of Science and Technology in Krakow, 2007 vol. 8, special edition, pp. 61–68.