HUNGARIAN JOURNAL
OF INDUSTRIAL CHEMISTRY
VESZPRÉM
Vol. 38(2). pp. 95-98 (2010)

RELATIONSHIP BETWEEN INPUT CHANNEL EXCITATION TIME AND PROFINET IO REFRESH TIME

I. FERENCZI

Department of Transportation Science and Infotechnology, University College of Nyíregyháza
Nyíregyháza, Sóstói út 31/b, HUNGARY
E-mail: ferenczi@inform.hu

Profinet IO is the communication concept for implementing modular, distributed applications such as Profibus DP, based on the Industrial Ethernet. Distributed IO and field devices are integrated into the Ethernet communication by means of Profinet IO. The user data from the field device are transmitted cyclically in a real-time channel to the process image of an automation system. In this paper, I will present what relationship exists between bus refresh time and input channel excitation time in several specific situations.

Keywords: Profinet IO, cycle time, real-time, refresh time, PLC, IO device, IO channel, delay time, communication

Introduction

Profinet IO permits direct interfacing of distributed field devices on the Ethernet. All devices used are connected in a uniform network structure, and therefore offer uniform communication throughout the entire production plant. The system consists of three main elements:

1. *IO-Controller;* it has control over a distributed process of one or more field devices. It receives process data and alarms and processes them in a user program. In automation installations, an IO-Controller is normally a programmable logic controller (PLC).
2. *IO-Device;* A PROFINET IO-Device (max. 128 device) is a field device connected in a decentralized way belonging to the process. It periodically transmits process data to the IO-Controller, and the IO-Controller transmits control data to the filed devices (*Fig. 1*).
3. *IO-Supervisor;* might be an engineering station (PC or Laptop) in an installation, which has temporary access to the field devices or controller during the commissioning process.

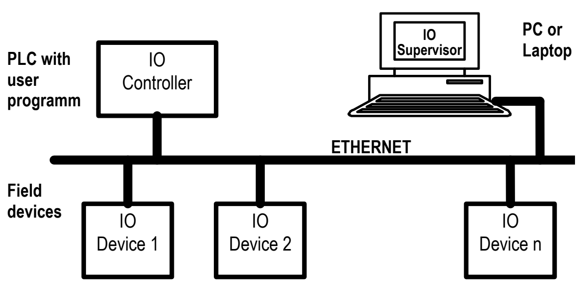

Figure 1: Elements of Profinet IO

With Profinet IO, the master-slave principle known from Profibus DP has been converted into a provider-consumer modell. So far communication is concerned, all Profinet devices have equal privileges on the Ethernet. However, a type of privilige is assigned to each device during configuration, and this defines the type and manner of communication according to the provider-consumer modell [3].

Profinet communication

In a real-time communication based on TCP/IP protocol, real-time (RT Class1,2) packets and Isochronous real-time (IRT or RT Class3) packets of controls requiring synchronization, which follow strictly real-time procedures, are also present beside non-real time (NRT) packets. [1]. The RT packets are supplied with priority to prevent a collision, and the IRT switch always provides an open line to the IRT packets [2].

Context management (CM)

An IO-Device delivers input data from the automated process to the IO-Controller, and receives output data from controller in order to control the process (provider-consumer). An IO-Supervisor can also communicate simmultaneously with an IO-Device. In order to permit data exchange to occur with all devices, an application and communication relation is necessary to exist. The task of context management is to manage the application and communication relations.

To establish a communication channel between provider and consumer is necessary to create a virtual logical channel for each channel of devices. This is the Application Relation (AR). The IO-Controller establishes an AR with each IO-Device. Establishment is carried out instantly during system startup. Several communication relations (CR) can be established within an AR *(Fig. 2)*. The following types of CR exist:

- Record Data CR for acyclic transmission of records, e.g. for configuration, parameterization, etc,
- IO Data CR for cyclic transmission of I/O data,
- Alarm CR, for acyclic transmission of events.

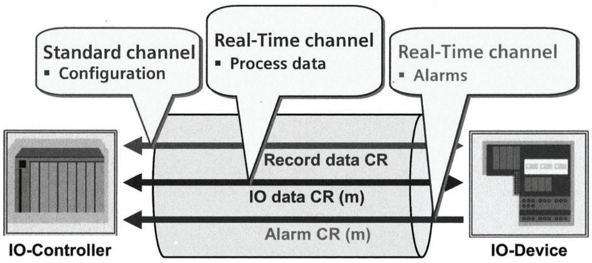

Figure 2: Communication relations

The last two communicate through a real-time, and the first through a traditional Ethernet channel. In this paper I deal only with the I/O Data CR.

The IO Data CR operation principle

The task of the IO Data CR is to transmit I/O data between IO-Controller and IO-Devices, according to the provider-consumer concept. The following parameters are transmitted during the establishment:

- A list of I/O data objects to be transmitted as well as their structure,
- The parameters of the send interval (send clock time, scaling, phase etc),
- The transfer frequency.

The number of IO Data CRs to be established is defined in the device configuration, depending of the number of devices installed. Two opposite IO Data CRs are always established, thus permitting bidirectional data exchange between IO-Controller and IO-Device. The data are sent cyclically from the provider to the consumer according to configured transfer frequency. In the relation, explicit acknowledgements of transmitted data frames do not take place but the consumer generate an error when the data frame listed does not arrive during a three I/O bus cycle. The data frame contains a cycle counter element, which is incremented, when the data does not arrive during a bus cycle time [3].

The IO Data CR features

1. The *send clock time* is the interval at which cyclic data are sent within an IO Data CR. It is defined device-specifically as an integral multiple of the basic time unit of 31.25 μs. This time is usually defined during configuration by the user.

   Send clock time = Send clock factor · 31,25 μs

   The *send clock factor* is between 1 and 128. A value of 32 corresponds to a send clock time of 1 ms that for a basic rate can be considered in a not synchronized RT communication *(Fig. 3)*.

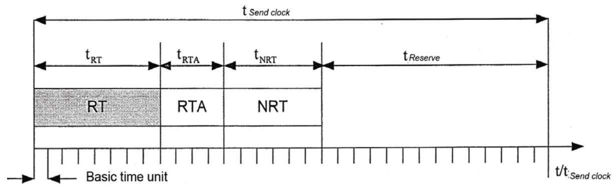

Figure 3: Send clock time for RT data

In Fig. 3 RT is a cyclical, RTA is an acyclical real-time, and NRT is a non-real-time data.

2. *Send interval (refresh time)*; since high-performance transmission of all data is usually not required, the communication transmission frequency may differ from the IO device. However, the slowest station must not determine the complete data throughput. For this session, low performance data are transmitted with scaling based on the send clock time:

   Send interval = Send clock time · 2^n

   where: 2^n – scaling; n – scaling ratio.

So, the refresh time is transmitted periodically that defines the send cycle. In this interval, the IO-Devices receive data from the controller, and transmit simultaneously the process data to the controller. This duplex communication is exemplified in *Fig. 4*.

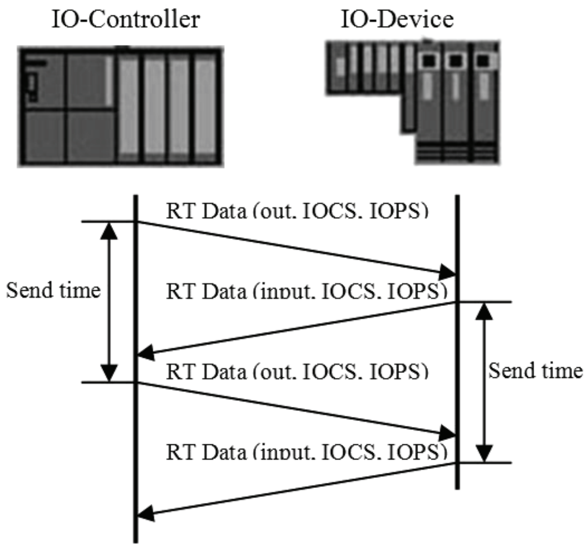

Figure 4: Real-time cyclical data transmission

Each item of I/O data contain two attributes, the IOPS (IO Provider Status) and IOCS (IO Consumer Status), which permit the IO-Controller and IO-Device to evaluate the quality of the transmitted value.

Determining some critical time values

In this paper I analyzed some critical situation when the input signal excites the input channel of IO-Device. The length of the time sequence between the starting moment of the channel excitation and the moment at which the reaction in the addressed output channel occurs can be very often decisive. Furthermore, it can also be of interest how frequently the incoming signals take place, by which obvious reactions are still observable. What is the minimal excitation time period that can still be identified by the input?

In order to answer these questions, the factors determining reaction time in the Profinet system should be identified:

- User program cycle time (C_T)
- Input channel delay time (I_D)
- Profinet IO refresh time set (U_T)

It is clearly visible that out of the three factors only the refresh time can be modified under the criteria of real-time systems. The other two determinants are not possible to be decreased considerably. It is also equally relevant when the input excitation occurs relative to the cycle time and the refresh time. In effect, the reaction time has a lower and upper boundary value.

The lowest boundary value (S_{RT}) denotes the, theoretically speaking, best possible situation where the response occurs within one cycle.

$$S_{RT} = C_T + I_D \quad (1)$$

In other words, the input excitation is initiated just in time to reach the end of the refresh interval and the update of the process image input (PII). Similarly, the response reaches the end of the refresh interval during the update of the process image output (PIO).

The worst possible is the situation (L_{RT}) in which the signal coming from the input of the IO device just misses the current refresh time and also the moment at which the process image input of the cycle gets updated after the next refresh time. Thus it will only be loaded and executed during the next cycle. The response then just misses the refresh interval during the update of the process image output of the next cycle; as a result, it will be only forwarded once the next cycle starts [5].

$$L_{RT} = 2 \cdot C_T + I_D + 4 \cdot U_T \quad (2)$$

Note: The situation described above can occur if the length of the refresh time sequence is nearly identical to the cycle time. If the refresh time is much longer than the cycle time, the response comes during the two refresh intervals even in the worst possible case.

Profinet IO test system

The PROFINET IO system takes the form of a star topology: a 100 Mb full duplex Ethernet network with a SCALANCE X005 switch. A S7-300 CPU315F-PN PLC with a 16-16 digital I/O module constitutes the controller. The IO device is an ET-200S with an IM151-3 PN IO device controller. It has 4 modules for the input and output respectively, with 2 digital channels each. A notebook has been used as an IO-Supervisor. Beside that, an S7-300 CPU314-2DP PLC is also part of the network, which connects to the network via a CP343-1 Lean interface. The latter does not take part in the measurement; it simply assumes a separate application task *(Fig. 5)*.

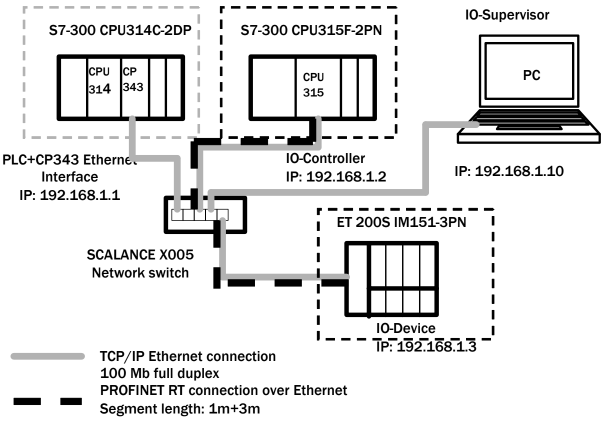

Figure 5: Structure of test system

Measuring with an universal counter

For the purpose of input excitation an impulse generator was used, which provided for 0.5 –1 kHz 24 V impulses with 1–50% duty cycles. The edge coming to the input and appearing on the output will be joined together after the differentiation. The input and output is visualized with a dual channel oscilloscope, and the resulted signal was applied to the input of the universal counter *(Fig. 6)*.

The advantage of this method is that a measurement of even 10 μs accuracy can be conducted. As a disadvantage, a supplementary circuit and a counter device are necessary.

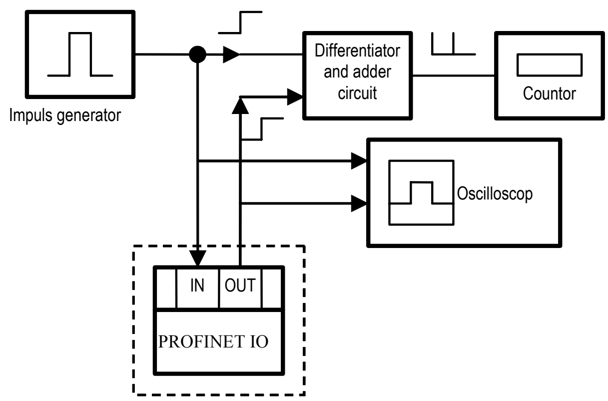

Figure 6: Measuring circuits

Minimal excitation time

At real-time communication system beside the definition of the reaction time we have to know the frequency of input signal applicable, respectively the signal span, mainly a minimal excitation time of input channel. This chapter is concerned with the finding the minimal excitation time that is still able to cause the input reaction.

This is largely determined by two factors: the input channel delay time and the refresh time. Let us denote the sum of these as t_{in}. We can state that:

$$t_{in} > U_T + I_D \quad (3)$$

In other words, the channel excitation time should be longer than the sum of the refresh time and the channel delay time. That is, in case of a 16 ms refresh time and a 3 ms delay time, this value has to be higher than 19 ms. The impulses with 0.5 Hz/1% duty cycle used during the test had a time duration of 20 ms [4]. Thus U_T up to a value of 16 ms was to function with safety. However, according to (3) at a 36 ms refresh time, the duty cycle has to be increased to 2%.

The following table shows the maximum frequency of the square wave signal in the given refresh time if the channel delay time is 3 ms or 0.5 ms.

Table 1: Results of the measurements

U_T	t_{in} [ms]		f_{in} [ms]	
[ms]	I_d = 3 ms	I_d = 0.5 ms	I_d = 3 ms	I_d = 0.5 ms
1	4	1.5	125	330
2	5	2.5	100	200
4	7	4.5	71	110
8	11	8.5	45	58
16	19	16.5	26	30
32	35	32.5	14	15
64	67	64.5	7	7

Also, it follows from the table what refresh time and channel delay time is needed to be set in order for the system to function safely. For instance, if a 50 Hz square wave signal has to be used on the input, then a refresh time of at least 8 ms with 0.5 ms delay time will be needed. However, if a channel delay time of only 3 ms can be used, it follows that the refresh time has to be set to 4 ms.

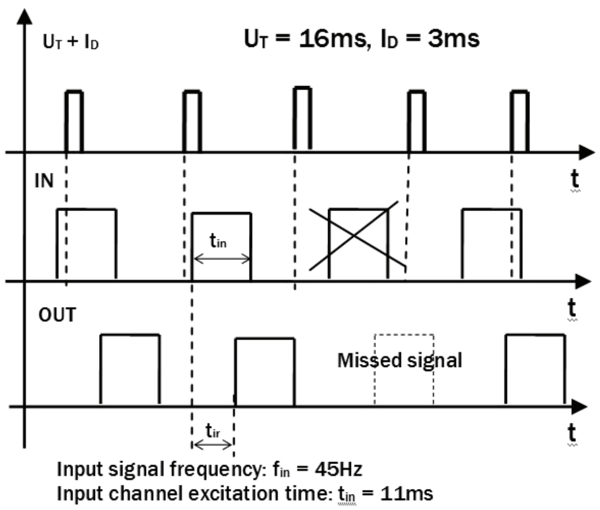

Figure 7: Bad input channel excitation

In the previous figure a case is shown, in which the excitation time takes a smaller value than that in (3). This can yet lead to an error because it may occur that the input signal can not cause a reaction *(Fig. 7)*.

In the case shown, the output channel reaction time (t_r) has no significance. Besides it always take a value between S_{RT} and L_{RT} but it does not depend on the input excitation time [4].

In the most critical case observed, in which the refresh time was 1 ms and the delay time was 0.5 ms (t_{in} = 1.5 ms), it followed that the input channels still react to a 330 Hz square wave signal. The measurements verified this assumption. Errors were not encountered even after multiple tests of longer durations. Further, the reaction times did not change considerably. It is not to say that the input won't react above 330 Hz but that cases may very well occur where input impulses remain without responses when an input excitation takes place in between two refresh times.

Conclusion

From the results of the measurements described, practical conclusions can be drawn with respect to how the Profinet IO system should be configured according to critical time limits of real-time processes. In the case of non-synchronized processes, it is imperative to know what differences occur between given limits, with other words, what are the value extremes between which the criteria of a real-time system are met entirely.

During the measurements that Ethernet communication took place without disturbance. The log file of the IO controller did not register a single collision or bad frame, albeit only a minimal Profinet IO configuration (controller, IO device and IO supervisor) were part of the network.

REFERENCES

1. R. PIGAN, M. METTER: Automating with Profinet (2006)
2. Profinet Real-time communication by Siemens AG (2007)
3. I. FERENCZI: Profinet Real-time kommunikációs stratégiák (Gép, 2009. dec.)
4. I. FERENCZI: Válaszidők vizsgálata egy Profinet IO rendszernél (Factory Automation 2010, Proceedings, 105–111)
5. Siemens Simatic CPU 31x: Specification Manual, 06/2008, A5E00105475-08