ETHERNET POWERLINK SIMULATOR

Václav Kaczmarczyk

Doctoral Degree Programme (3), FEEC BUT

E-mail: kaczmarczyk@phd.feec.vutbr.cz

Supervised by: František Zezulka

E-mail: zezulka@feec.vutbr.cz

Abstract: This paper is considered to be an introduction to the Ethernet Powerlink communication protocol performance evaluation. After the slight introduction to Industrial Ethernet and Ethernet Powerlink specifically, the idea of the network simulator is described. If such a simulation is made during network design, it can reveal incorrect behavior of the network in early project phase and thus save time and human resources. Economical aspects of such solution are obvious.

Keywords: Ethernet Powerlink, Simulation, Data Link Layer

1. INTRODUCTION

Nowadays Ethernet is the dominant networking solution for the home and office environment. It is fast, easy to install and most equipment now come with a built-in Ethernet interface. The cost of the network infrastructure is decreasing. Ethernet also allows connecting almost any type of device. Moreover, with the aid of the Internet technology, connected devices can be anywhere. With all these pros, Ethernet had a large potential to become an ideal solution also for automation technology. However, it had been known that Ethernet is not suitable for industrial networking because the medium access control method of Ethernet, CSMA/CD, exhibits unstable performance under heavy traffic and unbounded delay distribution [1]. The switched Ethernet shows a very promising prospect for real-time industrial networking, because the full-duplex access, which can eliminate frames collision, is used. The Ethernet without collisions is no longer unstable, under heavy traffic and its delays can be reduced. Therefore the switched Ethernet seems to be inherently suitable for industrial demands.

<u>Industrial Ethernet – necessary conditions, pros, cons, examples</u>

The adoption of Ethernet technology for industrial communication seems to be the forward step, but cannot be acceptable if any of important field area's features should be lost. These features are namely:

- time-deterministic communication;
- time-synchronized actions between field devices like drives;
- efficient and frequent exchange of very small data records.

An essential requirement is that the office Ethernet communication capability is fully retained so that the entire communication software involved remains usable. This results in the following requirements:

- support for migration of the office Ethernet to Real-Time Ethernet;
- using standard components: bridges, Ethernet controllers, and protocol stacks as far as possible [2].

Using Real-Time Ethernet for industrial communication brings some advantages. Well-established popular solution that uses unified physical layer leads to lower prices and vast choice of network components [3]. Link layer, compatible with the TCP/IP standard, allows addressing packets by common layer-two switches (Remark that industrial network components, which implement additional services, e.g. QoS, should be used for industrial applications). Moreover, both real-time communication and non-RT data traffic can be transferred simultaneously, although every industrial standard solves this problem in its own way. The solution of Ethernet Powerlink, designed by B&R, is described in chapter 2.

2. ETHERNET POWERLINK

Ethernet POWERLINK (EPL) was originally developed by B&R GmbH and it is currently managed by the Ethernet POWERLINK Standardization Group (EPSG) [4].

The EPL network is based on the definition of a Data Link Layer (EPL DLL) protocol, placed on top of the IEEE 802.3 Medium Access Control (MAC) natively used by Ethernet. The EPL protocol stack is shown in Figure 1. As the picture shows, the protocol defines the application layer as well. It relies on the well known CANopen profiles. These profiles basically state that process data are to be transferred as communication objects over communication relationship.

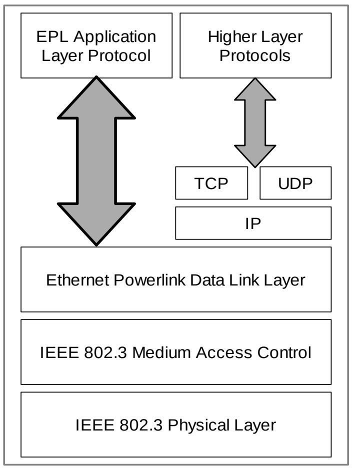

Figure 1: Ethernet Powerlink stack [5]

2.1. DATA LINK LAYER

The EPL data link layer protocol is based on a Time Division Multiple Access (TDMA) technique which grants each station with the exclusive access to the network, avoiding collisions. EPL defines two types of stations, namely Managing Node (MN) and Controlled Nodes (CNs). Each network contains exactly one MN, which represents the controller, and several (up to 240) CNs. The communication is based on the principle of using a master-slave scheduling system on a shared Ethernet segment. The interconnection between network devices may be realized either via hubs or switches. Note that hubs are actually recommended for connection, though differences between latencies and jitters on hubs and switches are negligible nowadays. Note that due to the evolution of the Ethernet technology, hubs are rapidly disappearing from the market and, consequently, their costs are increasing. The announced version of EPL, running at 1 Gbps, will make use of switches [6]

The EPL DLL protocol is based on a master-slave relationship realized by means of a continuously repeated sequence of operations as can be seen in Figure 2. The MN sends a multicast start-of-cycle (SoC) frame to signal the beginning of a cycle. Then, the isochronous period is entered. During this period, the MN sends the poll request frame PReq to each CN which, consequently responds with the poll response frame PRes. PReq frames are issued to carry output data, while PRes to carry input data. MN finishes the isochronous period by sending the PRes broadcast frame to all CNs.

Consequently, the start-of-asynchronous (SoA) frame is broadcasted by MN to notify the beginning of the asynchronous period. The SoA frame contains the specification of a CN that is granted to send one acyclic message. CNs ask for this permission during the isochronous period. When CN finishes sending the asynchronous frame, an idle phase starts.

During the idle phase there is no communication on the network. This phase is intended to equate durations of adjacent cycles, i.e. to ensure broadcasting SoC frame in periodical time instants.

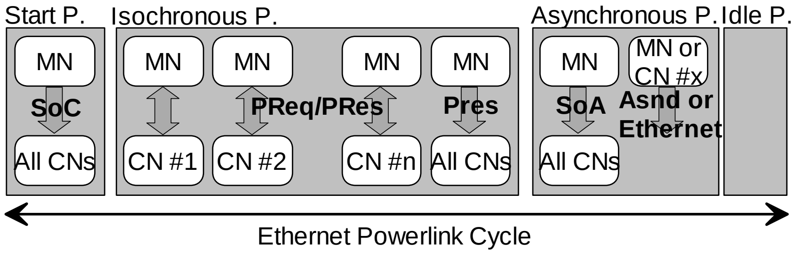

Figure 2: Ethernet Powerlink Cycle

Higher EPL layers (especially the Application layer) are not in the scope of this research and thus are not described in this paper.

3. OMNET++ NETWORK SIMULATION FRAMEWORK

OMNeT++ is a discrete event simulation environment. Its primary application area is the simulation of communication networks. OMNeT++ provides component architecture for models. Components (modules) are programmed in C++, and then assembled into larger components and models using a high-level language (NED). OMNET++ consists of

- simulation kernel library;
- compiler for the NED topology description language;
- OMNeT++ IDE based on Eclipse platform;
- GUI or command line environment for simulation execution;
- utilities, documentation, sample simulations etc.

4. SIMULATION SCOPE DEFINITION

As the Ethernet Powerlink communication protocol is robust and well-defined, it is also very complex. For the simulation purpose it would be disserviceable to understand and express its complete functionality as it is described in specification [4]. At the very beginning stage the decision of the simulation scope should be done. Following bullets specify the scope step-by-step.

- The application layer, although contains huge amount of objects, object dictionaries, flags, control mechanisms, safety parts, is fully implemented (programmed) as a task of the Real Time OS. As a consequence of this we can say, that in case of correct implementation, the functionality of the Application layer is correct and trouble-free. The only problem is to provide powerful enough hardware; hence the application layer is not in the scope of this simulation.
- The data link layer is much more interesting. It defines the EPL Media Access Control mechanism, which is responsible for a collision-free communication between all network nodes. Figure 3 shows the time diagram of the typical EPL network consisting of one MN and several CNs.
- Figure 3 shows typical progress of communication when network is being established. Before the normal operation mode (OP), the MN (Managing Node) walks internally through several states, establishes connection with all connected CNs and configures them. The duration of each and every configuration state (named NA, PO1, PO2 and RO) does not have to fulfill any criteria, i.e. MN can remain in every state for (almost) arbitrary-length period. The initialization process is thus not in our research scope.

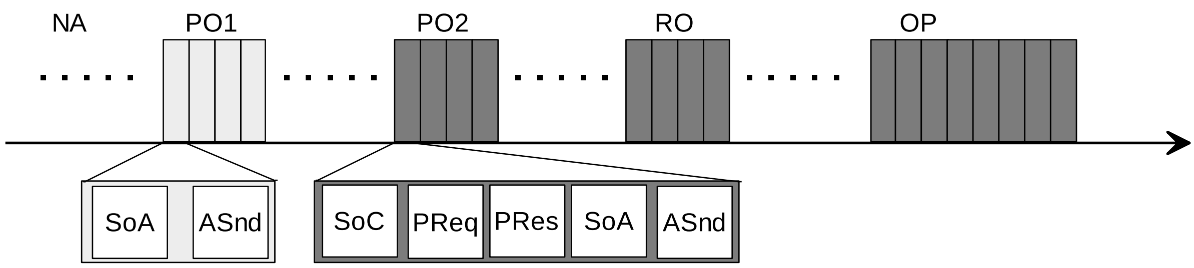

Figure 3: Ethernet Powerlink time diagram

5. MODEL

The research is focused on standard EPL communication cycle when the initialization process is fully finished. Both MN and all CNs are in their "operational" states. Firstly, both MN and CN operational states has to be analyzed. According to the specification [4], behavior has been described and transformed to the OMNeT++ programming language. Figure 4 shows illustration example of an EPL network modeled by OMNeT++ framework.

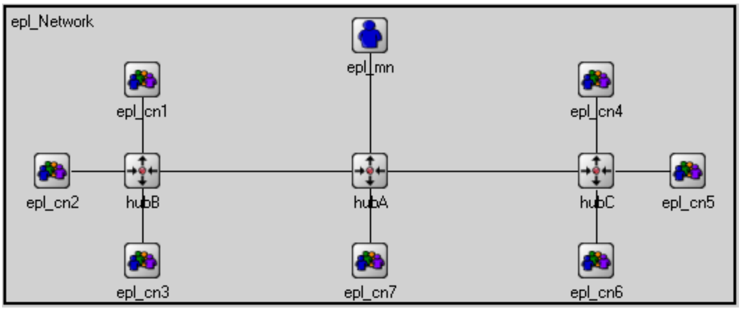

Figure 4: Example of an EPL network

As we can see, there is one instance of the EPL_MN device and several (7) instances of the EPL_CN device. Apart from these devices, three Physical-Layer switches (hubs) are used to connect all components.

- The main functionality of the MN is to keep the network communication. The MN starts the communication cycle in predefined intervals, calls particular CNs, starts the asynchronous phase, decides which CN is allowed to send its message during asynchronous phase. Moreover, MN is responsible for response processing, measuring time spans, guarding all boundaries (e.g. cycle time exceeding) etc.
- All CNs are created as instances of one template (with different parameters). CNs are passively listening the network. In case the MN asks particular CN, this CN should process the MN's message and send the answer to the network. According to specific parameters, CNs can generate asynchronous events. When such an event is generated, CN tries to deliver it to the MN. Mean event delivery time is one of the most relevant network loading indicators.
- Hubs are simple switches that reproduces incoming packet to all physical ports (except of the port the packet has came) with some propagation delay.
- Wires are metallic connections, each of them 100 meters length. Signal propagation speed is set to be similar to real metallic wires.

6. SIMULATION RUN AND EVALUATION

When simulation is run, the first network cycle starts immediately. All statistical data objects are emptied. During simulation, events are generated, transported to MN, and all important time spans are stored. When simulation is finished (according to defined end condition that can be e.g. 1 million cycles elapsed), all stored data are saved to data files and can be shown. Later data files can be processed by OMNeT++ utilities to get understandable results.

Table 1 shows an example of part of the output data file.

```
epl_Network.epl_cn1      "Isochronous messages received"        25474
epl_Network.epl_cn1      "Asynchronous messages received"       25474
epl_Network.epl_cn1      "Asynchronous messages transmitted"    327
epl_Network.epl_cn1      "Asynchronous events generated"        327
epl_Network.epl_cn1      "Asynchronous events still waiting"    0
epl_Network.epl_cn2      "Isochronous messages received"        25474
...
```

Table 1: A part of output data file

7. CONCLUSION

The simulation shown mutual timing relationship between MN and connected CNs. Obtained results validated the model of the real network. After such validation, modeled network can be connected and configured according to model parameters.

Nowadays, EPL device models are still qualitative models. We proposed quantitative parameters on the basis on real network measurement and information from EPL standard [4]. However, our measurement is not precise and has been done only to get slight knowledge of EPL behavior and with insufficient tools. The further research opportunity is thus to make more detailed measurement of quantitative parameters. Moreover, CNs asynchronous behavior should be developed with respect to real demands (e.g. implementation of asynchronous packet flooder).

ACKNOWLEDGEMENT

This work was partially supported in by grant „Modern Methods and Approaches in Automation“ from the Internal Grant Agency of Brno University of Technology (grant No. FEKT-S-10-12), and Grant Agency of the Czech Republic (102/09/H081 SYNERGY – Mobile Sensoric Systems and Network).

REFERENCES

[1] Lee06 Lee, K. Ch. et al.: Worst Case Communication Delay of Real-Time Industrial Switched Ethernet With Multiple Levels. IEEE Transactions on Industrial Electronics, 2006, vol. 53, p. 1669-1676

[2] Felser, M.: Real-Time Ethernet - Industry Prospective. Proceedings of the IEEE, 2005, vol. 93, p. 1118-1129

[3] Prytz, G.: A performance analysis of EtherCAT and PROFINET IRT. IEEE International Conference on Emerging Technologies and Factory Automation, 2008. ETFA 2008. p. 408-415.

[4] Ethernet Powerlink Standardization Group: Ethernet Powerlink, Communication Profile Specification version 1.1.0, available on: http://www.ethernet-powerlink.org/

[5] Seno, L., Vitturi, S.: A simulation study of Ethernet Powerlink networks. IEEE Conference on Emerging Technologies and Factory Automation, 2007. ETFA. p. 740-743.

[6] Cena, G. et al.: Performance analysis of Ethernet Powerlink networks for distributed control and automation systems. Comput. Stand. Interfaces. 2009. vol. 31. p. 566-572.