Chapter 19
SoftPLC-Based Control:
A Comparison between Commercial and
Open-Source EtherCAT Technologies

Stefano Scanzio
Istituto di Elettronica e di Ingegneria dell’Informazione e delle Telecomunicazioni, Italy

ABSTRACT

Nowadays, most control systems are based on programmable logic controllers (PLCs) and on commercial programs, but the use of conventional PCs is becoming an increasing by popularity and competitive alternative. After an introduction that underlines the positive and negative aspects of both solutions, EtherCAT, a hard real-time industrial Ethernet protocol that allows the use of both conventional PCs and open-source software, is analyzed. Two different EtherCAT master implementations are compared, which are based on a commercial and on an open-source software, respectively. In order to highlight differences and provide a short reference, a simple example was implemented in the two environments.

INTRODUCTION

In the past decade, the use of conventional PCs in the place of special purpose hardware is becoming more and more attractive in factory automation environments. At the same time digital communication networks, based on the standard Ethernet technology, are becoming progressively dominant in the market, replacing legacy fieldbuses based on proprietary hardware. Factory automation, usually less responsive to innovation than the information and communication technology (ICT) world, achieved from this synergy clear benefits in terms of cost, portability and performance. Some Ethernet-based protocols, known as real-time Ethernet (RTE), have been developed in order to exploit completely this technology. Very important
SoftPLC-Based Control

examples are EtherNet/IP, Ethernet Powerlink and EtherCAT.

Among them, a relatively new and promising protocol is EtherCAT. EtherCAT is a real-time network protocol based on Ethernet, with an high communication efficiency (higher than 90%). It relies on a master/slave approach, where the master cyclically send commands to slaves. While slaves are built using custom hardware components, the master is implemented in software and can be run directly on conventional PCs, with negligible degradations in performance. Nowadays, both commercial and open-sources solutions are available. The most popular commercial implementation of the EtherCAT master (EM) is TwinCAT. On the other side, the most important open-source EM is the EtherLab EM, which is based on Linux.

After an explanation of the EtherCAT protocol, this chapter will focus on the differences between commercial and open-source implementations of EMs. Some aspects of the two solutions are evaluated, taking a special care on their advantages and drawbacks. Then, a simple application is implemented using both TwinCAT (representative of a commercial software) and EtherLab EM (representative of an open-source software). The differences between the two approaches will be highlighted, as well as the specific aspects of the software implementation.

For a better understanding of the problems and of the code, the example will be focused on a simple application that use few slaves and, basically, performs timestamps on digital inputs or carries out actuations on digital outputs.

BACKGROUND

Before dealing with the EtherCAT protocol, a rapid overview of the more popular RTEs (Cena, 2011) will be provided. Basically, today, there is a wide range of RTEs, that differs in many aspects, e.g., the performance, the network topology, the use of special designed hardware, the synchronisation protocol, and so on.

EtherNet/IP and PROFINET IO are two examples of RTEs that rely on a star topology based on switches. The first relies on normal switches and uses the Common Industrial Protocol (CIP) to manage communications. Real-time data are exchanged through UDP and watchdogs are used to guarantee real-time requirements. Real-time messages are usually sent according to a producer/consumer paradigm and in multicast. The use of conventional switches degrades the determinism of this protocol due to the queuing of packets in the output ports of switches. The CIP Sync protocol, based on the IEEE 1588 protocol, is used for the synchronisation of the network nodes. On the other side, PROFINET IO follows a very different approach, where modified switches are used to obtain determinism. PROFINET IO divides the network bandwidth into four phases depending on traffic priorities, namely RED, ORANGE, GREEN and YELLOW. The RED traffic is the one with highest priority. This phase is characterised by a reserved bandwidth in the domain of the time, enforced by means of the modified switches. In this phase only RED traffic can be sent. The size and number of RED packets is defined and scheduled “a priori” during network configuration, so this phase is characterized by high determinism. In the ORANGE phase the infrastructure is reserved for ORANGE traffic which is routed using the conventional Ethernet MAC protocol, while the GREEN phase is used for best effort transmissions. The YELLOW phase allows the sending of packets with the same priority as those sent in the GREEN phase. Those packets must be completely transferred within the end of the YELLOW phase. The PTCP protocol, that is a modification of the last version of IEEE 1588 protocol, is used to synchronise switches.

Ethernet Powerlink uses hubs instead of switches. It is a master/slave protocol where all the communications are started by the master node, the managing node (MN). Each slave,
Related Content

Equipment Replacement Decisions Models with the Context of Flexible Manufacturing Cells
[www.igi-global.com/chapter/equipment-replacement-decisions-models-context/69297?camid=4v1a](www.igi-global.com/chapter/equipment-replacement-decisions-models-context/69297?camid=4v1a)

Missing Value Imputation Using ANN Optimized by Genetic Algorithm
[www.igi-global.com/article/missing-value-imputation-using-ann-optimized-by-genetic-algorithm/209380?camid=4v1a](www.igi-global.com/article/missing-value-imputation-using-ann-optimized-by-genetic-algorithm/209380?camid=4v1a)

Train Timetable Construction
Masoud Barah (2013). *Graph Theory for Operations Research and Management: Applications in Industrial Engineering* (pp. 262-270).
[www.igi-global.com/chapter/train-timetable-construction/73166?camid=4v1a](www.igi-global.com/chapter/train-timetable-construction/73166?camid=4v1a)

A Study of Product Development Engineering and Design Reliability Concerns
[www.igi-global.com/article/a-study-of-product-development-engineering-and-design-reliability-concerns/202422?camid=4v1a](www.igi-global.com/article/a-study-of-product-development-engineering-and-design-reliability-concerns/202422?camid=4v1a)