

RTAI based measuring and control over EtherCAT

Fodrek Peter

Kamensky Jaroslav

Department of Information and Communication systems,
Institute of Automatic Control and Industrial Informatics,
Faculty of Electrical Engineering and Information Technology
Slovak University of Technology

Ilkovicova 3, 812 19 Bratislava, Slovak republic
peter.fodrek@stuba.sk

Abstract

This paper describes educational stand for teaching distributed control systems. It uses industrial real-time Ethernet framework called EtherCAT. For I/O we used modules from German company Beckhoff. EtherCAT master kernel module is used for Etherlab in Linux with RTAI. We mention Hardware configuration, system software and application architecture as well as some troubles in the solution. This paper is based on master thesis of Jaroslav Kamensky from June 2009 supervised by Peter Fodrek.

1 Introduction

Paper is about distributed control over EtherCAT. This technology is replacement of the fieldbus based on Ethernet. It solves main problem of the fieldbuses that is its cost because of low count of sold components compared to the standard hardware as Ethernet. Problem of Ethernet is that it was not designed for low latency and hard realtime but for soft realtime of a large amount of data. EtherCAT or Ethernet POWERLINK solves this problem. Then there is section about Ethercat I/O modules from German company and EtherCAT inventor Beckhoff. We used module from it as they were taken to us as a gift. Software contains three modules: 1st for communication over EtherCAT, 2nd for control and 3rd a only user space module for GUI in JAVA. We uses analog Computer to simulate analog process in task of prove functionality. We used discrete PID to control non-linear system but there is possibility to use any other control law.

2 Ethernet

As we mentioned using Ethernet as the control loop fieldbus replacement we saw three main disadvantages:

- I. Too much ISO/OSI levels and therefor too much metadata for device communication with communication with low amount of data.
- II. Too much wiring and therefor too addition cost for low number of devices and CPU performance.
- III. Not enough realtime capabilities because of too much CPU utilization.
- IV. Star topology is not suitable for industrial control in many cases.

Because of this there was developed addition and changes to Ethernet to use it as fieldbus in distributed control. Most common are Ethernet PowerLink and EtherCAT. In the paper there we will be

mention only EtherCAT.

3 EtherCAT

Best addition is to add linear structure of the cables as show on the Figure 1

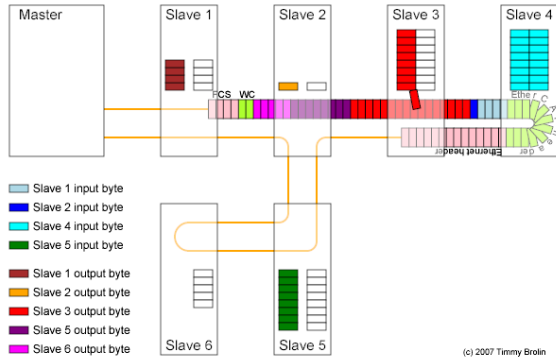


FIGURE 1: *Ethercat Principle*[2][3]

Frame of the Ethercat based on Ethernet is shown in the Figure 2

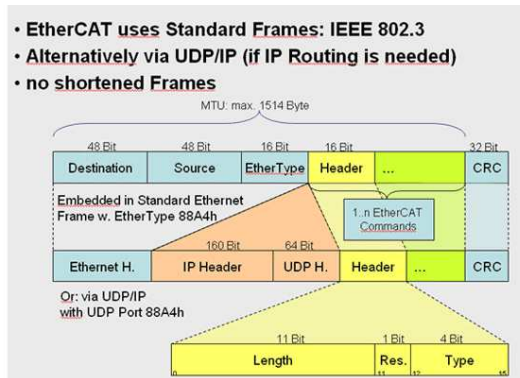


FIGURE 2: *Ethercat Frame*[2][3]

EtherCAT used on the fly read/write to the Ethernet Frame which is produced by EtherCAT master. EtherCAT master is software and therefore a standard Ethernet adapter can be used. Only devices must be EtherCAT devices. This approach made the author of EtherCAT German company Beckhoff famous. Each device can have several bytes on several places in the EtherCAT frame as shown in Figure 3.

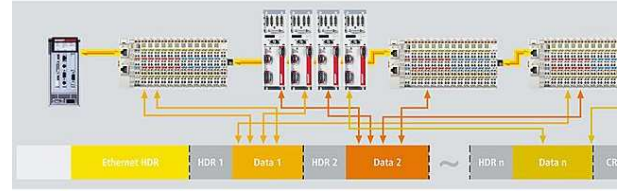


FIGURE 3: *Ethercat Device data in the EtherCAT Frame*[2][3]

Another problem is how to deal after all modifications will return to the master for processing in the PC and GUI. Each system has a virtual ringbus architecture with the possibility to use hardware ringbus when the device can be sent in which of go-return part of the frame processing will process which bit of the frame as shown in Figure 4.

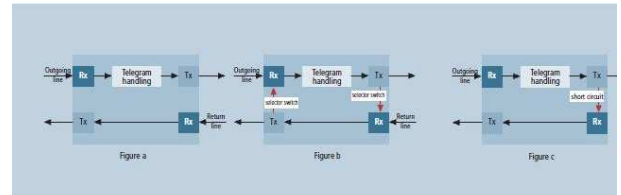


FIGURE 4: *EtherCAT ringbus*[2][3]

These changes are enough to achieve hard real-time capability with not non-realtime devices on the net. There is a possibility to use non-realtime hardware, but then there is a need for a special EtherCAT master to avoid frame collision with not real-time communication. Hard real-time hardware module is, for example, a configurable module from Beckhoff as seen

on the figure 6

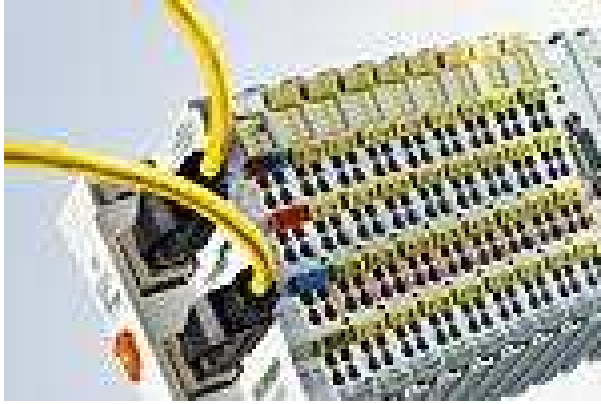


FIGURE 5: *Beckhoff device we used*

For our application only three modules are used communication module on the Figure 6, Analog input module for E-bus on the Figure 7 and Analog output module on the Figure 8. all these modules are from Beckhoff. We were connected to the communication module with much more modules on the e-bus but we use these modules

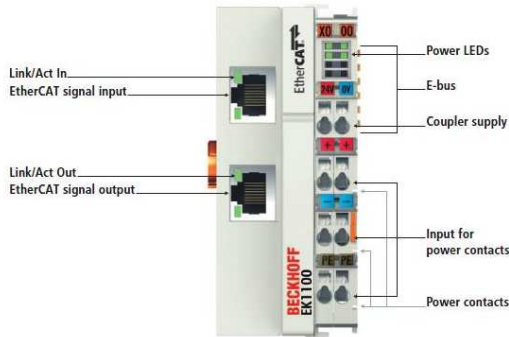


FIGURE 6: *Ethercat block*

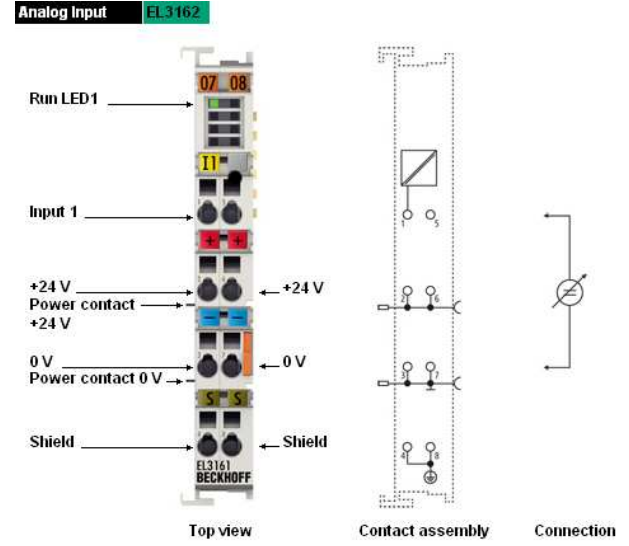


FIGURE 7: *Ethercat block*

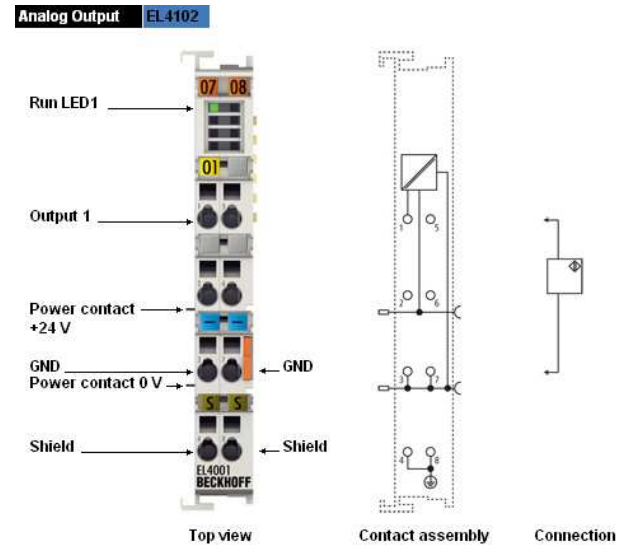


FIGURE 8: *Ethercat block*

4 Analog Computer

We have connected analog output and input to the analog computer MEDA-50 (MEDA=Maly Elektroniczny Diferencjalny Analizator or Small Electronic Differential Analyzer) as shown on the Figure



FIGURE 9: *Analog Computer Meda-50*

As we shown TV is not requirer but can be used in some cases to monitor proces but we do not used it.

5 Applictaion architecture

We used RTAI as Realtime nanokernel, EtherCAT master program form project Etherlab with Realtek Ethernet adapter. Therefor only several versions of the Ettherlab will work on specific kernel version with it. We use one mole to communicate with two kernel modules of the Etherlab master. One of it if for Ethernet adapter and one for API generalization. Our module set Etherlab master make communication domains and channels for specific and very well documented hardware form Beckhoff. Next module is used to count controller output or control law. it communicated with user space JAVA GUI application via Realtime FIFO pipes. GUI is able to change type of discrete PID with windup effect of controller output limiter and with anti-windup correction. It also use two types of discrete PID for both cases. Each of this four controllers can have own parameters of the gains and period of the timer of the control loop or discrete system period.

6 Application

We produced application in the Slovak language to manage control loop and for displaying values as of Figure 9. Values are like in control theory: w - for wish value, e - for deviation of the wish value and process output y for process output and u - for controller output. There are two controller outputs u_{poc} for algorithm output and u_{real} for real input of the system with mentioning bounds. Really only u_{real} is send to the Output modules because it does not accept larger value then maximum output. As seen on the Figure 10. there are 4 controllers and parameters are changes only when Zapis=write for controller is pressed Choice of the active controller is applied automatically. You can choose which of the values to display in the graph. As well as the wish value. It is applied when nastav=set button is pressed. Controller can be enabled(zapnuty) or disabled(vypnuty) and as well as integral part to be reset to the 0.. Graph can be static (staticky) or dynamic (dynamicky). static graph means that data are refreshed after data for new full screen graph arrived. Dynamic plots each time value arrive to the GUI application. For controller you can set gain for (P)roportional, (I)ntegral and (D)ifferential part a well as time period T .

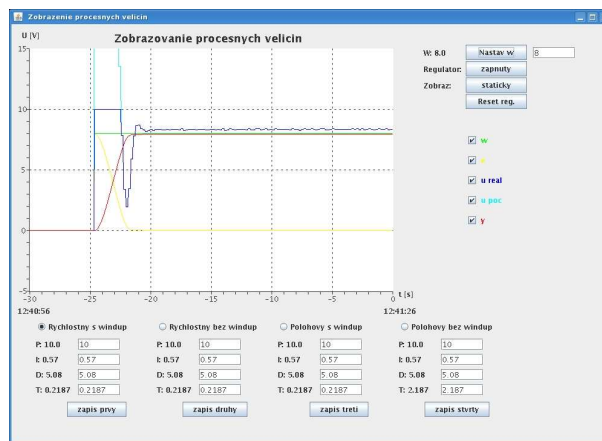


FIGURE 10: *Application*

7 Conclusions

We described our application used to demonstrate possibility to control in the industry via RTI and EtherCAT. It is not perfect and needs future improvements but student was learned ho to write kernel module RTAI application and JAVA GUI. As well as Java Native interface to use RTAI FIFO pipes for part of application to communicate.

References

- [2] [*http://en.wikipedia.org/wiki/Ethercat*](http://en.wikipedia.org/wiki/Ethercat)
- [3] [*http://www.ethercat.org/en/technology.html*](http://www.ethercat.org/en/technology.html)
- [1] [*http://www.etherlab.org/en/ethercat/index.php*](http://www.etherlab.org/en/ethercat/index.php)
- [4] [*http://www.rtai.org*](http://www.rtai.org)

This paper is supported by grants from APVV and VEGA which solution finished in 2008 or first half of 2009.