Functional Interface for Universal Access to Ethernet-based Field Bus Protocols

Armin LECHLER
Institute for Control Engineering of Machine Tools and Manufacturing Units (ISW), University of Stuttgart, Seidenstr. 36, 70174 Stuttgart, Germany

and

Alexander VERL
Institute for Control Engineering of Machine Tools and Manufacturing Units (ISW), University of Stuttgart, Seidenstr. 36, 70174 Stuttgart, Germany

ABSTRACT
Today’s field bus systems for connecting automation components in and between machines and units have a very high performance in regard to their transmission rate and real-time capability. Furthermore, they comprehend a multitude of functionalities like cross communication, safety protocols and automated address assignment. In addition, they define special logical views on the individual types of machines in terms of machine profiles. But these extensive functionalities make the configuration and programming of applications communicating via these field bus systems extremely complex. On top of the complexity in the communication of a single system there is a multitude of systems that are totally incompatible. Nevertheless, they hardly differ in most applications in regard to their range of functions and performance.

In this paper, a method is presented with which these systems can be standardized so that the application communicating over such field bus systems will be independent of the system applied. This method is generally described and exemplified by means of a standardized state machine of two systems.

Keywords:
Functional Interface, Field Bus, Ethernet, Universal Application, Communication

1 INTRODUCTION
In the middle of the eighties field bus systems were developed in order to reduce the extent of wiring within automation systems. Various branches had different requirements on the systems in regard to real-time capability and bandwidth, and consequently adequate field bus systems for the respective demands of each sector were developed. But these systems are incompatible not only in terms of their transmission, the bus access and the protocol structure, but also in regard to the device profiles. The description of the logical view on the individual device types as, for example, drives, I/Os or pneumatic components is differing in the presentation.

The trend in automation engineering towards decentralization and intelligent mechatronic systems leads to an increased amount of data in and among such systems. In order to meet these requirements, since the year 2000, the various field bus systems on the market had been ported to the Ethernet technology (Figure 1) which is widespread in offices. The use of available Ethernet components should result in additional cost reduction and reduced implementation effort. Along with the usage of the new technology for field bus systems new systems with new protocols were generated.

Figure 1 Development of Ethernet-based Field Buses

Even though all field busses were now based on Ethernet, this did not lead to a standardization of the systems. The higher bandwidth by using the Ethernet technology only contributed to the fact, that there is almost no more unique selling point for individual systems. Systems that, for example, so far mostly have been applied for the integration of drives and are characterized by a short cycle time, have been expanded by mechanisms, in order to logically integrate I/O devices in the network. Systems that were more so designed to interlink I/Os were extended by the ability to communicate with drives.

The requirements on field bus systems of most automation systems can basically be met by each of the existing Ethernet-based systems. There is a partially standardized transmission on the layers 1 – 2 of the ISO/OSI Layer model, but there is absolutely no standardization to be found in the higher layers, especially not in the system and application layer. Thus, the application layer includes bus system-specific parts of the used communication system, since there is no clear abstraction layer leading to the communication systems. This applies to the programming of the application as well as to the configuration and the operation. In all these disciplines the expertise and knowhow of the used communication system is necessary.
These field bus systems are extremely complex, not at least due to the increased range of functionalities. The implementation effort for a software stack, which processes the communication, is enormous. Producers of components thus have to opt for one or just a few systems for support, on top of the corporate policies. In addition, it is very important that users who develop automation systems, know about the incompatibility and the different user interfaces of field bus systems. Due to these differences between the field bus systems the application program of the automation system has to contain field bus-specific instructions for its correct implementation effort for a software stack, which processes the application programming and process control, but not in communication technology. On the one hand, a field bus-specific application program cannot run on an identical automation system with only one other field bus system. On the other hand, a user needs to have expert knowledge in regard to the respective field bus system for the initial operation or the maintenance.

There must not be any limitations of the functionalities that are provided by the field bus systems. If certain functionalities cannot be standardized, the unified interface will be extended by a generic part that allows the access to non-generalizable functionalities.

2 STATE OF THE ART

Previous work dealt with a standardized hardware interface and general exchange mechanisms [1],[2],[3] as Dual Ported Ram, mailbox procedures and interrupt handling, but not with the actual functionalities. Therefore, there still remains bus-specific knowhow on both sides of the interfaces. They only specify mechanisms to access the bus system physically not functional.

Also producers of chips, who offer products for the support of several field bus systems, need field bus-specific knowhow for operating the chip interfaces [4]. These chips contain all physical elements for the access to various bus systems, but they do not encapsulate and standardize the functionalities and parameters. Those have to be implemented in a separate driver for each system. The chips only allow for a unified hardware design.

The standardization of device profiles for electrical drives IEC 61800 [5] only describes different drive profiles like CANopen, SERCOS and Prodrive, but no unification of these profiles.

Present comparisons of field bus systems were always drawn in regard to the real time capabilities and not the functionalities and their implementation. These comparisons are not apt as a basis universal functional access to such systems. Further work attended to the modeling of field bus systems, but not with their standardization [6],[7],[8].

3 FUNCTIONAL INTERFACE

For the definition of a standardized interface representative field bus systems specified by user organisations are examined by means of an object-oriented analysis. With the here displayed method for the object-oriented analysis, an abstraction of the bus systems is done in the beginning by means of the specifications of the system functionalities. These are described with features so that in a following step the detailed implementations of the system functionalities can be analyzed. The description of the functionalities allows a classification and a comparison of the implementation.
Based on this classification, a structure and a standardized functional interface are defined. This interface allows a unified access to the underlying field bus system. Thus the system and the application layer are independent of the field bus. The developed method is applied exemplary in the bus configuration of the two Ethernet-based field bus systems SERCOS III and EtherCat.

**Abstraction**

During the object-oriented analysis first an abstraction of the bus system functionalities is carried out by using the present specifications. The current systems are analyzed to find out which functionalities they provide (Figure 5).

Based on these functionalities a classification can later be done. At first, a rough definition of these functionalities is established. Based on this, the functionalities that characterize field bus systems are collected. The descriptions of the individual field bus systems are varying to a high degree, because a clear separation between the layers of the ISO/OSI model does not exist. Therefore, first the functionalities are structured that generally characterize field bus systems. Each of these functionalities is assigned various features that relate to the configuration, the data, the algorithms and the call. Based on these features, the functionalities can be modeled exactly.

**Figure 5 Abstraction of field bus systems**

In the following step these functionalities are defined more precisely by means of the particular specifications of the analyzed field bus systems (Figure 6). It is examined how each of the regarded systems implements this functionality. This means, which data is used in this functionality, which information is required for the initialization, what the calls for this functionality look like and which algorithms are used. This results for each of the field bus systems in a concrete model of the functionalities and a description based on the features.

**Figure 6 Functional analyses**

**Classification**

The next step is a classification of the examined systems to find overlapping functionalities and features (Figure 7). Based on the detailed specifications of the functionalities and their features a comparison is made. Here, the individual functionalities are compared and relations between them detected and described. From this classes and their relations between them are derived. Sometimes iterations with one of the previous steps are necessary and, where required, also new definitions of functionalities and features. From this comparison results a classification that describes the similarities and differences of the specific systems.

**Figure 7 Classification**

**Interface definition**

From this a class structure is defined which allows a unified access on all available functionalities as a user interface (Figure 8) and encapsulates the field bus-specific attributes. In which layer of the ISO/OSI layer model this interface can be found differs according to the specific functionalities. For every functionality a standardized interface is defined for its encapsulation. Thus the complexity and the bus-specific knowhow are introduced in the object. This results in a
simplification of the interface and independence of the bus system. Each functionality is described as a class, and it may consist of several subclasses. Functionalities that cannot be generalized because they a bus system-specific are depicted with a generic interface section which allows access to the bus-specific functionalities.

Figure 8 Interface definition

4 RESULTS

At first, this method was described using as an example a class that models the configurations of the field bus systems EtherCat and SERCOS III universal (Figure 9). This class contains the respective universal status machine and the necessary interfaces as methods for the configuration of these field bus systems. The specific bus features as objects all relate to an interface identical to the application layer. The definition of this interface now allows a field bus-independent functional access to these systems and consequently the generation of a universal standardized application program. For the Profinet state machine for boot up the same functional interface could be used.

Figure 9 boot up of SERCOS and EtherCat

5 SUMMERY

In this paper a method for standardizing Ethernet-based field bus systems was presented and demonstrated by an example. It was proven that it is possible to communicate with field bus systems through a standardized interface and thus implement a field bus-independent system and application layer.

6 REFERENCES

[7] Li, Yue; Bewertung der Echtzeitfähigkeit von Feldbussystemen; Fortschr.-Ber. VDI Reihe 10 Nr. 235 Düsseldorf VDI-Verlag 1993