Contribution to EFOC/LAN-88 in Amsterdam, June 29 - July 1, 1988

Architecture and Performance Evaluation of a MAP–Gateway

Martin Bosch, Ottmar Gihr
University of Stuttgart
Institute of Communications Switching and Data Technics
Seidenstraße 36, D–7000 Stuttgart 1
Federal Republic of Germany

The interconnection of the Manufacturing Automation Protocol (MAP) architecture and a vendor specific protocol architecture for factory automation is presented. After classifying interconnection possibilities, the architecture of a MAP–Gateway is described and modelled for performance evaluation. First simulation results conclude the paper.

1. Interconnection of Networks

1.1 Motivation

Today, many different vendor specific protocol architectures are existing. At the same time many standardization bodies, especially the International Organization for Standardization (ISO), try to achieve stable standards for all layers of the Basic Reference Model for Open Systems Interconnection (OSI). In order to satisfy different requirements of communication, several alternatives of protocols can be selected in these standards.

Special coupling devices are necessary to interconnect networks with different protocol architectures. Such coupling devices are also needed to achieve larger network dimensions and to improve performance, security, availability or fault isolation by subdividing large networks into smaller ones with a great amount of internal traffic.

1.2 Coupling Architectures

Several possibilities for the architecture of a coupling device are existing. In the following, it is assumed, that the protocol profiles of the networks to be interconnected use identical protocols at and above a specific layer N.

The first possibility is depicted in Figure 1. Here the networks are interconnected at the last different layer N–1 by protocol conversion. Protocol conversion is relatively difficult to implement and a loss of functionality is often unavoidable. On the other hand there are advantages like an end to end protocol at layer N and the minimum amount of layers in the coupling device, which results in a minimum transfer time through this device.

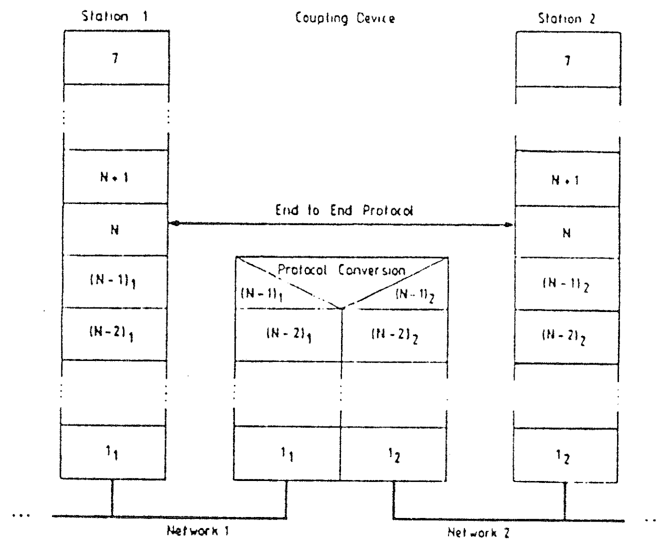


Figure 1 : Network interconnection by protocol conversion

Another possibility to interconnect networks at the last different layer N–1 influences all devices at the interconnected networks and should therefore only be used in new installations. Here the protocols of layer N–1 are enhanced to a global sublayer of layer N–1. This is depicted in Figure 2. Again an end to end protocol exists at layer N.

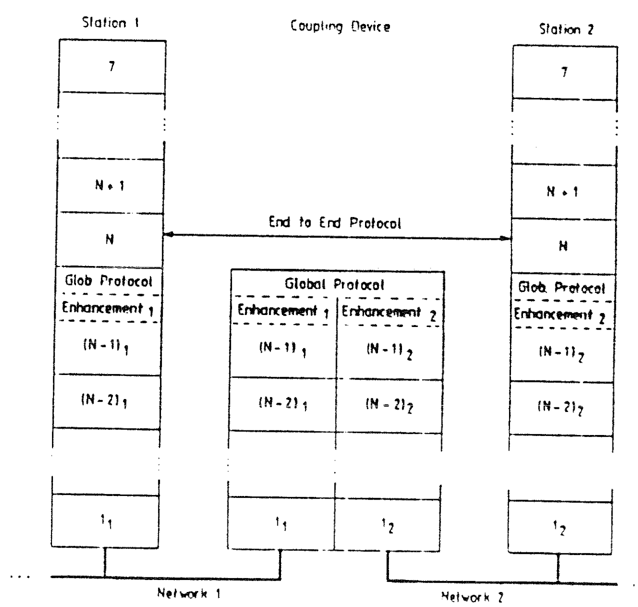


Figure 2 : Network interconnection by a global protocol

It is also possible to interconnect networks at layer N, which is the first layer with identical protocols, see Figure 3. Here the *Protocol Data Units* (PDUs) reach the *Service Access Point* (SAP) of layer N and are then reflected to the other network. This means a relatively simple implementation with the following drawbacks: loss of the end to end protocol at layer N and relatively high transfer times because of the additional layer N in the coupling device.

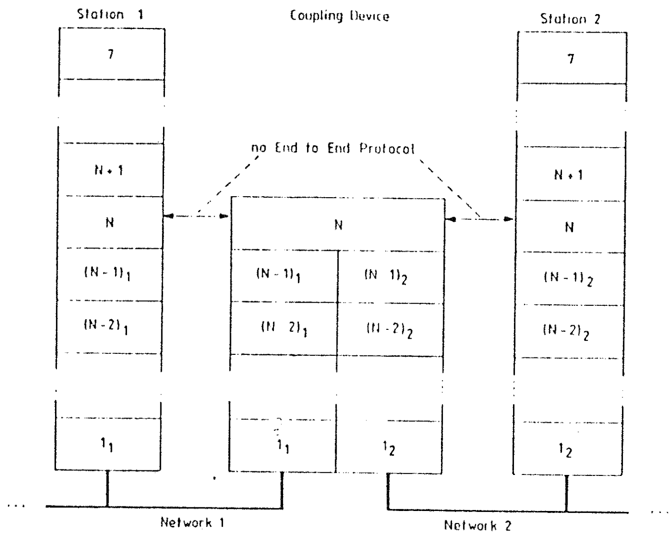


Figure 3 : Network interconnection at the first identical layer

In the special case of interconnecting two identical networks via a *Wide Area Network* (WAN), the WAN can be used as a transit system. In Figure 4, PDUs from network 1 are taken as pure information packets, which are supplied with an additional control information (header) for the WAN. At network 2 this control information is removed to obtain the original PDU. Note, that in this case it is impossible to communicate with devices at the WAN itself.

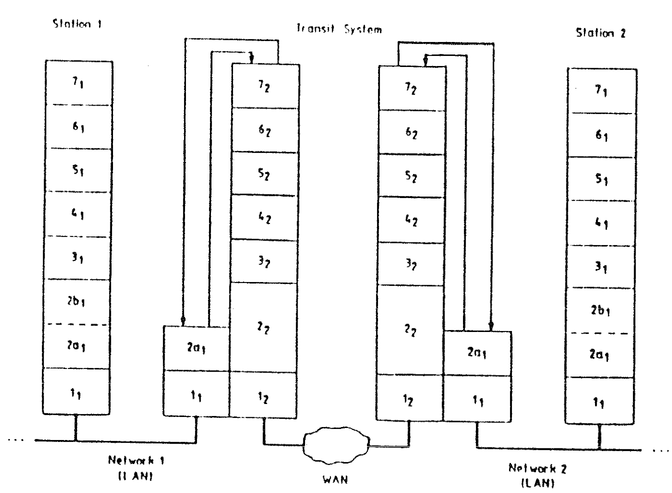


Figure 4 : Network interconnection via a transit system

1.3 Classification of Coupling Devices

If the networks to be interconnected are coupled at the physical layer, the coupling device is called a *repeater.* A repeater is necessary to extend the dimension of a network by interconnecting network segments which are limited in size because of physical constraints. The interconnected segments may use different physical media. For further spatial extension, it is also possible to split a repeater into two halves, which are interconnected point to point for example by fibre optic. This is called a *remote repeater.*

At the data link layer, networks can be interconnected by *bridges.* It is possible to interconnect networks with different Media Access Control (MAC) sublayers. If the Logical Link Control (LLC) sublayer in each interconnected network is identical, the protocols can be converted at the MAC sublayer. In this case the bridge is called a *MAC-layer-bridge.* There are two proposals for standard algorithms in MAC-layer-bridges: the spanning tree algorithm [3] and the source routing algorithm [4]. Bridges are usually used to extend the dimension of networks. In comparison to repeaters they have the advantage, that they can perform a filtering function which increases the performance of the network, and that they have separate media access controls. If two networks are interconnected via a backbone network with a bridge at each interface, the term *remote bridge* is used.

Interconnection of networks at the network layer is done by *routers.* Routers often use a common internet protocol and usually interconnect more than two networks with each other. Routers can be employed to interconnect networks with different protocols at the lower three layers, especially connectionless and connection oriented networks, or to interconnect identical networks to form one huge network. Bridges and repeaters cannot be used in huge networks, because there is no efficient routing function in their coupling layer.

If the protocols at the transport layer or above are different, the networks must be interconnected by *gateways.* In a gateway the interconnection takes place at the transport or at the application layer. Consequently the end to end protocol at the transport layer is lost, and the second network, together with the gateway, appears to the first network as a distributed end system instead of a subnetwork.

2. Protocol Architectures

2.1 Protocol Architectures for Factory Automation

Due to an increasing number of intelligent controllers for machines like Numerical Controllers (NC), Programmable Logic Controllers (PLC) or Robot Controllers (RC), communication requirements between these devices and the hierarchically higher devices like Cell Controllers (CC) are increasing rapidly.

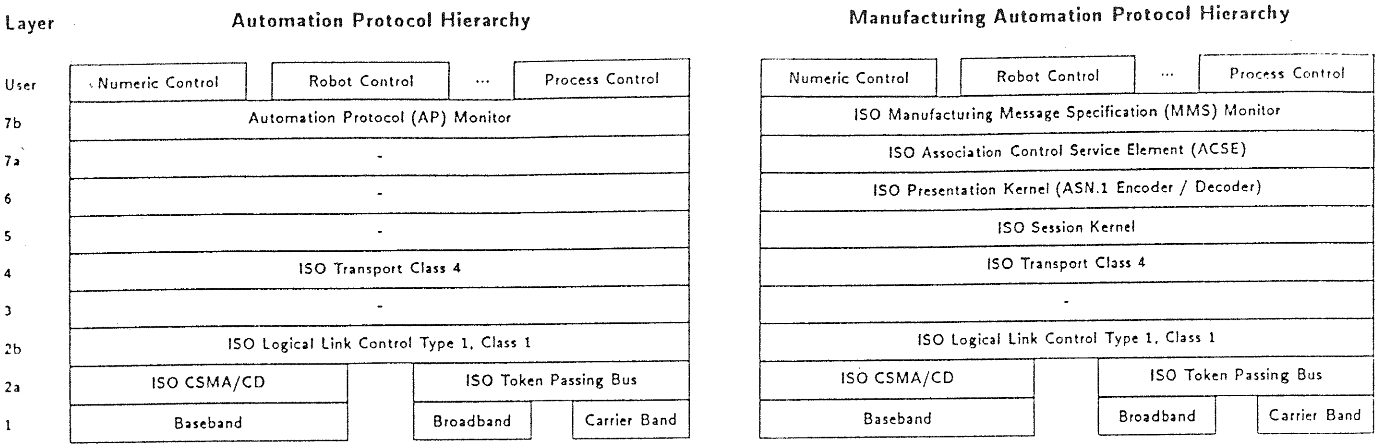


Figure 5 : Protocol architectures for AP and MAP

In many cases computer systems of different vendors can only communicate with each other, if there is a coupling device between the different systems. In order to solve these communication problems, a multi vendor project has been initiated by General Motors in 1980, to develop a standardized *Manufacturing Automation Protocol* (MAP) [15]. This project is accompanied by many other groups or multi vendor projects like the European Map User Group (EMUG) since 1985 or the ESPRIT (European Strategic Program for Research and Development in Information Technology) project Communications Network for Manufacturing Applications (CNMA) since 1986.

MAP is based on the ISO Reference Model for Open Systems Interconnection (OSI) [5]. Suitable options have been chosen from existing standardized protocols whenever possible. For the special application in factory automation a new protocol at the application layer has been specified and proposed for standardization.

Until MAP reaches the status of a stable standard, communicating devices have to use vendor specific protocol architectures like the *Automation Protocol* (AP) architecture [16, 17], which is depicted in Figure 5 together with the MAP architecture.

At layer 1 (physical layer) and 2a (MAC) the *CSMA/CD Access Method* (Carrier Sense Multiple Access with Collision Detection) [11] is used. In the MAP specification [15] the use of the *Token Passing Bus Access Method* [11] is recommended. The LLC sublayer 2b [10] is using the connectionless, unconfirmed datagram service. Up to now, the implementation of layer 3 (network layer) is not necessary, because the communication is limited to one *Local Area Network* (LAN). The connection oriented transport protocol class 4 [6] is used at layer 4 (transport layer).

At the moment, layers 5 (session layer) [7] and 6 (presentation layer) [12] of the MAP architecture contain only kernel functions. Especially the encoder and decoder of the used syntax at layer 7 are located at layer 6. This syntax is described in *Abstract Syntax Notation One* (ASN.1) [13]. The *Association Control Service Element* (ACSE) [9] is the common part of many standardized *Application Specific Elements* (ASEs) at layer 7 (application layer) as for example *File Transfer, Access and Management* (FTAM) [8]. Additionally, the ASE *Manufacturing Message Specification* (MMS) [14] has been prepared for factory automation by the Electronic Industries Association (EIA) and has now reached the status of a ISO Draft International Standard (DIS) 9506.

In the AP architecture the AP Monitor [16, 17] realizes all necessary functions of layers 5, 6 and especially 7.

2.2 Protocol Architecture of a MAP–Gateway

The coexistance of the MAP architecture and vendor specific architectures like AP, will result in *MAP–Gateways* to interconnect the different protocol profiles and thus to enable the development towards *Computer Integrated Manufacturing* (CIM). A MAP–Gateway to AP, which is depicted in Figure 6, is the subject of the following chapters. The implementation of this gateway will be transparent for its users as far as possible.

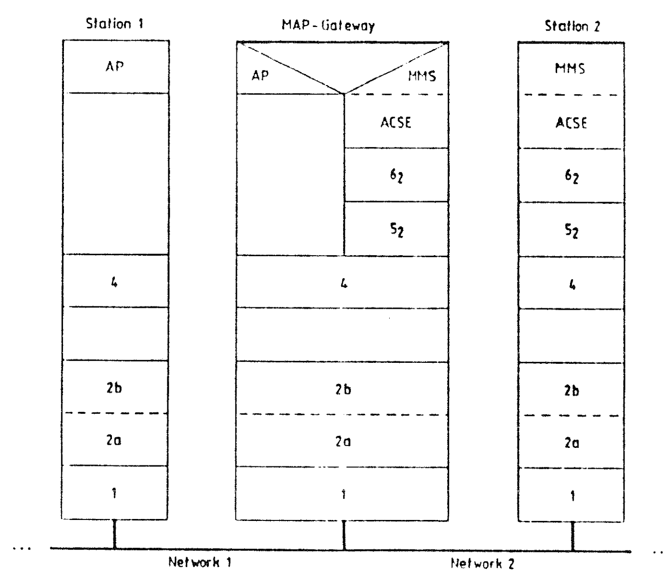


Figure 6 : Interconnection of networks via a MAP–Gateway

3. Application Layer Protocols and their Conversion

Due to different protocols at the application layer, the MAP–Gateway has to do protocol conversion at the application layer between the MMS protocol and the AP protocol. This kind of interconnection leads to unavoidable high communication costs, reduced performance and reduced functionality of the interconnected networks. Therefore these two protocols have to be considered in more detail.

3.1 The Vendor Specific Application Layer Protocol AP

AP is an application layer protocol, which also includes tasks of the presentation layer and the session layer as far as necessary. AP has been defined because of the lack of stable standardized protocols for the higher layers. It will be substituted by standardized protocols as soon as stable versions are available, so that only little effort has been made to achieve a layered architecture at the higher layers. In the vendor specific environment, the communicating partners and especially their external view of data structures are well–known. Therefore, users create PDUs in a form, which can be transmitted directly without the need of a presentation layer.

AP provides services to control and supervise devices, to transfer programs and data, and services for data administration. No standardized transactions or well–defined sequences are existing, but the user has the possibility to define transactions by his own. AP services may either be unconfirmed or confirmed. There is also the possibility to have flow controlled segmented services for large messages and to have a reaction to a service, which again may be segmented or not. Specific services exist for the local organization. There is an administration time for each service, which is the time, the client (active user) is willing to wait for an acknowledgement. If this administration time despires, AP creates a negative acknowledgement for the client. Within the administration time there is the possibility for confirmed services to be repeated each time a response timer expires. Each service is stored in an administration list at the application layer until an acknowledgement is received. The service is then returned to the user together with the acknowledgement, so that the assignment is guaranteed.

Many application channels may be multiplexed onto one transport connection, if they belong to the same server (passive) station. There is a directory service to keep and administrate the network project data. The topology, that means application channels and transport connections, are static during normal operation. Therefore the connectivity has to be projected first, to establish each transport connection and each multiplex channel during the startup phase.

At present, there are no standardized state machines for file transfer or for program control. This leads to severe difficulties for protocol conversion.

3.2 The Standardized Application Layer Protocol MMS

MMS has been developed for the MAP architecture as a specific application layer protocol for manufacturing automation. MMS uses the standardized basic application layer protocol ACSE to establish associations, corresponding to contexts in MMS. This implies, that in MMS contexts can be initiated and concluded dynamically during normal operation. It is also possible to establish some basic contexts during startup phase. The presentation layer contains an encoder/decoder for PDUs described in ASN.1, so that MMS has to handle PDUs described in ASN.1.

Our work is based on an implementation of the CNMA project [2], which uses a well selected subset of MMS at the application layer. In this implementation there is a one to one mapping between context, association and transport connection.

In MMS many scenarios are defined in order to solve different problems. Within these scenarios confirmed or unconfirmed services are possible. In each context only one scenario is allowed at a time. The context initiating station is the calling station and the other station is the called station. There are services, which are only used by a calling station. Therefore, it might be necessary to initiate a second context between the same entities in the opposite direction in order to satisfy specific scenarios. During each context initiation a set of vertical and horizontal Conformance Building Block (CBB) classes is negotiated to define which services and data structures will be allowed on this context during its lifetime.

The essential part of the MMS philosophy is the *Virtual Manufacturing Device* (VMD), the server of a communication relationship. The station using a VMD is the client. A VMD always contains a domain. A domain can contain data, a program or part of a program and refers to a corresponding state machine. Optionally it can contain further domains, program invocations, a virtual filestore, variables, semaphores and event conditions. Many MMS services are available for reading the contents or attributes of the listed objects. There are also MMS services to create, delete and manipulate these objects or to ask for the state of a corresponding state machine.

3.3 Protocol Conversion at the Application Layer

PDUs of one network must be converted into PDUs or sequences of PDUs of the other network. Various possibilities for the parameters of these PDUs can occur: parameters used in both networks, parameters that must be converted, parameters only used in the first network, which are at most useful to the MAP–Gateway and which can be ignored in the second network, and parameters only used in the second network, which must be created by the MAP–Gateway using tables or directory services.

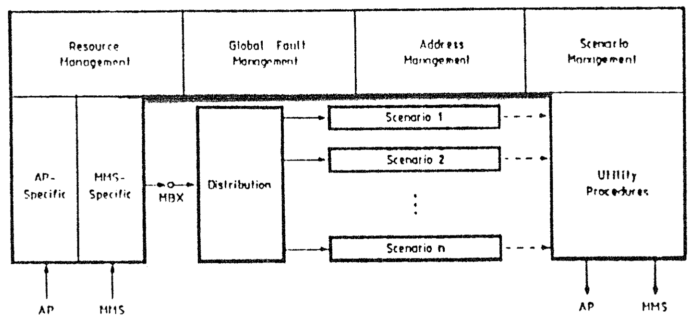
No additional flow controls are necessary, because existing application layer flow controls are used such, that a confirmation to the original sender is returned only, if the gateway has received an acknowledgement from the receiver in the case of confirmed services. In the case of unconfirmed services no connection of flow controls is intended.

Addressing is another aspect in interconnecting different networks at the application layer. Because of the corresponding peer entity in the gateway, the gateway itself instead of the server has to be addressed from the client. The gateway has to look in a directory for the connection, which refers to the arriving connection and it has to address the server. If there is no corresponding connection, a new connection has to be established and the directory must be updated.

4. Implementation Aspects

We are implementing a prototype of the described MAP–Gateway. At the institute, we have installed a LAN with the CSMA/CD media access method. This network (SINEC H1, Siemens Network for Automation and Engineering High Performance 1) uses standardized and identical protocols for both protocol architectures at the lower four layers. The stations in our network (including the MAP–Gateway itself) are six Personal Computers (Siemens PCs 16–20) with memory extensions and intelligent LAN controller boards, which handle the lower four layers (layers 1 and 2a in hardware and layer 4 in software). The higher layers have to be handled by the PC processor. Due to identical protocols at the lower four layers, it is possible to interconnect both networks physically on the same medium, although there are logically two networks and communication between stations with different architectures is only possible via the MAP–Gateway. Therefore it is possible to have only one LAN controller board in the MAP–Gateway and to separate the protocol architectures above the transport system. This configuration is depicted in Figure 6. If two physically separated networks would have to be interconnected, two LAN controller boards would be necessary in the MAP–Gateway. In practice it is unlikely to have two identical media instead of one installed in parallel and therefore, we are implementing the version with one LAN controller board. The PC processor has to handle both protocol stacks above the transport layer as well as the protocol conversion software, simultaneously. If there is also user software running on the station (for example, if the MAP–Gateway is a cell controller and able to use both protocol stacks), this software must be handled additionally by the same PC processor. However, this would result in a slow and unreliable software and should therefore be avoided.

There are two steps during the development of the MAP–Gateway. In the first step, two independent tasks can be done in parallel. The first task is to adapt the existing protocol stacks and drivers to the used operating system as well as to the hardware configuration and to implement a station which is able to use the two protocol stacks simultaneously (that is the MAP–Gateway of Figure 6 without the protocol conversion software). The MAP–Gateway is constructed by using available components. The second task is the development and implementation of the protocol conversion software itself, which is completely done at the institute. The protocol conversion software can be divided into a gateway management system being responsible for all common tasks, and a library for the individual scenarios to be handled by the MAP–Gateway, as depicted in Figure 7.



AP ... Automation Protocol
MBX ... Mailbox
MMS ... Manufacturing Message Specification

Figure 7 : The protocol conversion software

The second step is the integration of the protocol conversion software into the station which is able to use the two protocol stacks in order to form a prototype of the MAP–Gateway.

5. Performance Evaluation

We have implemented a basic version of a simulation program to evaluate the performance of our MAP–Gateway. The modelling is done essentially by composing two existing models [1], one of the AP architecture and one of the MAP architecture. We have used an event by event simulation technique. First simulation results with realistic assumptions for the parameters [1] are depicted in Figure 8 for the case of a one to one mapping of confirmed services in the MAP–Gateway and for the direction from MAP to AP. The transfer time is defined as the time from the generation of a service to the arrival at the server application. The buffer occupation time is defined as the time from the generation of a service to the arrival of the acknowledgement at the client application.

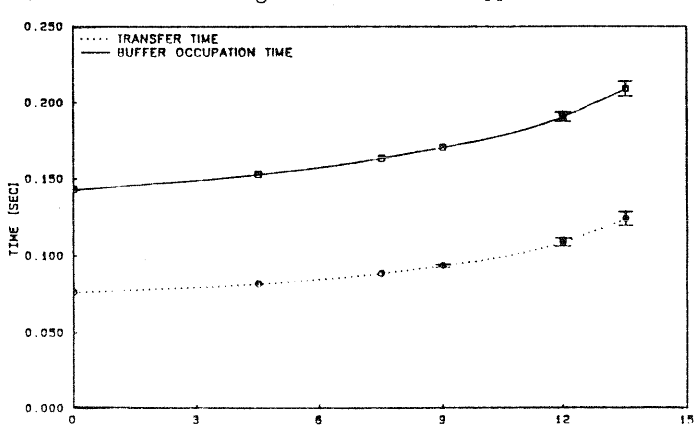


Figure 8 : Transfer time and buffer occupation time

6. Conclusion

In this paper several possibilities for the interconnection of networks with different protocol architectures have been presented. The architecture and implementation of a MAP-Gateway has been described. The implemented prototype will be continuously evolved and improved and it will be presented in the framework of a CNMA Pilot Installation in Stuttgart. Furthermore, we have implemented the basic version of a simulation program for this MAP-Gateway, which will be extended to allow the simulation of various specific scenarios.

Acknowledgement

The authors would like to thank J. Nonnast for the investigation of the application layer protocols and the possibilities of a protocol conversion between MMS and AP, and G. Mauch for the implementation of the simulation program.

References

[1] M. Bosch, O. Gihr; "Modelling and Performance Comparison of Two Application Layer Protocols for Manufacturing Automation", International Teletraffic Congress (ITC) 1988, Torino, June 1 – 8, 1988

[2] CNMA (ESPRIT Project 955); "MMS Interface Specification", Version 5.0, Siemens AG, March 18, 1988

[3] IEEE 802.1; "MAC Bridges", Working Draft, Part D, August 1987

[4] IEEE 802.5; "Enhancement for Multi-Ring Networks", Draft Addendum, September 15, 1987

[5] ISO 7498; "Information Processing Systems – Open Systems Interconnection – Basic Reference Model", November 1983

[6] ISO 8072; "Transport Service Definition", May 1984
ISO 8073; "Transport Protocol Specification", May 1984

[7] ISO 8326; "Basic Connection Oriented Session Service Definition", September 1984
ISO 8327; "Basic Connection Oriented Session Protocol Specification", September 1984

[8] ISO DP 8571; "File Transfer, Access and Management", April 1985

[9] ISO DIS 8649; "Service Definition for Common Application Service Elements – Part 2 : Association Control", April 1986
ISO DIS 8650; "Protocol Specification for Common Application Service Elements – Part 2 : Association Control", April 1986

[10] ISO DIS 8802/2; "Local Area Networks – Logical Link Control", 1985

[11] ISO DIS 8802/3; "Local Area Networks – CSMA/CD Access Method", 1985
ISO DIS 8802/4; "Local Area Networks – Token Passing Bus Access Method", 1985

[12] ISO DP 8822; "Connection Oriented Presentation Service Definition", May 1986
ISO DP 8823; "Connection Oriented Presentation Protocol Specification", May 1986

[13] ISO DIS 8824; "Specification of Abstract Syntax Notation One (ASN.1)", 2nd DIS, September 1986
ISO DIS 8825; "Specification of Basic Encoding Rules for Abstract Syntax Notation One (ASN.1)", 2nd DIS, September 1986

[14] ISO DIS 9506; "Manufacturing Message Specification, Part 1: Service Definition", December 1987
ISO DIS 9506; "Manufacturing Message Specification, Part 2: Protocol Specification", December 1987

[15] MAP; "Manufacturing Automation Protocol", Version 3.0, General Motors, Warren/Michigan, April 7, 1987

[16] SINEC; "Automation Protocol (AP 1.0), Teil 1: Protokollspezifikation", Siemens AG, 30. Juni 1987
SINEC; "Automation Protocol (AP 1.0), Teil 2: Technologische Funktionen", Siemens AG, Dezember 1987
SINEC; "Automation Protocol (AP 1.0), Teil 3: AP-Protokollabwickler", Siemens AG, Januar 1988

[17] SINEC; "SINEC Communication Processor (SCP) – Software", Siemens AG, 19. Dezember 1986