Design, Implementation, Modelling and Simulation of a MAP–Gateway for Flexible Manufacturing

Martin F. BOSCH

Institute of Communications Switching and Data Technics
University of Stuttgart
Stuttgart, Federal Republic of Germany

It is necessary for the development of flexible manufacturing to migrate towards a standardized Manufacturing Automation Protocol (MAP) profile for the information flow, which allows communication between devices from various vendors. This migration will only succeed, if the new MAP networks can be embedded into existing proprietary environments using vendor specific protocol profiles for factory automation. As central components in this context MAP–Gateways are necessary to allow communication via the network boundaries and to prevent manufacturing islands. This paper describes design and implementation of a specific MAP–Gateway, as well as its modelling and the simulation of various scenario classes.

1. Introduction

1.1. Development of Factory Automation

In order to achieve a maximum of flexibility and to be able to produce efficiently small series of products in a minimal time, it is necessary to integrate all computers and control systems of a company. Therefore, *Computer Integrated Manufacturing* (CIM) means the development from isolated components towards a distributed system of control systems and computers. *Local Area Networks* (LANs) are necessary to interconnect all components of this distributed system.

Today, various proprietary protocol profiles for factory automation 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) [6], including network management and directory service. In order to satisfy different needs of communication, several alternatives of protocols can be selected in these standards.

1.2. Various Requirements for Networks

There are various requirements that networks have to meet in companies. For the administration or technical office, the *Technical and Office Protocols* (TOP) profile has been specified. *Gateways* allow the transfer of data via *Wide Area Networks* (WANs) to distant locations and *Bridges* [4, 5] or *Routers* [17, 18] to the LANs in the manufacturing environment.

For factory automation, MAP has been defined as a specific standardized protocol profile. This profile is adequate for the upper levels of the functional hierarchy in a company, due to usually missing real time requirements on these levels. The transfer of a large amount of data is necessary there, although not very frequently. There are also proprietary protocol profiles due to a missing complete stability of the MAP standard up to now. These proprietary networks can be connected to MAP networks via *MAP–Gateways*.

On the lower levels of the functional hierarchy, the requirements are quite different to those mentioned above. Small messages comprising only a few bytes have to be transmitted frequently in real time. Due to its complexity full MAP is too slow to fulfil these requirements. Therefore, *field busses* are necessary in restricted areas as manufacturing cells.

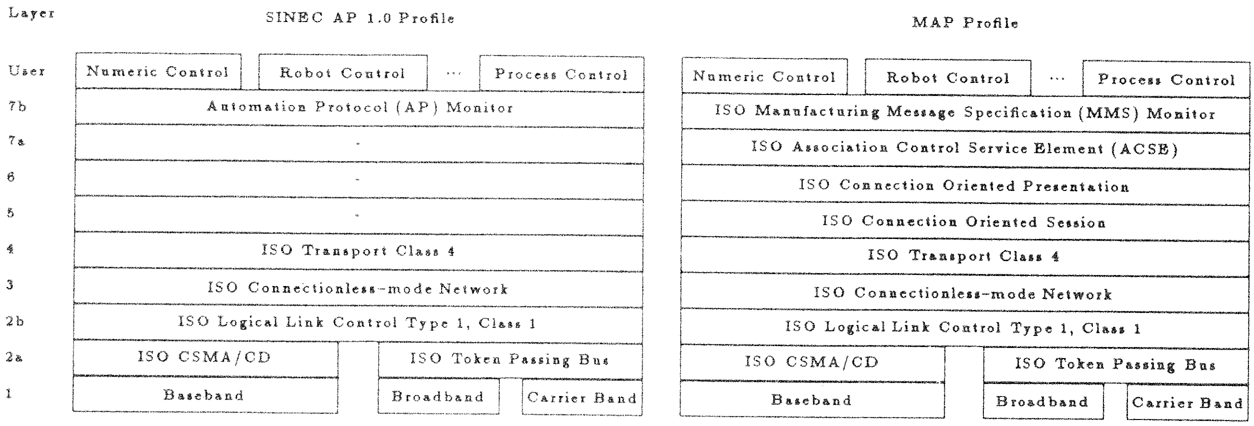
2. Specific Protocol Profiles for Factory Automation

2.1. The Proprietary Protocol Profile SINEC AP 1.0

Until MAP reaches the status of a stable and complete standard for all aspects of communication, including network management and directory service, communicating devices have to use proprietary protocol profiles. One example is the *SINEC AP 1.0* (**SI**emens **NE**twork **A**r**C**hitecture for Automation and Engineering, **A**utomation **P**rotocol **1.0**) profile [20, 21], which is depicted in Figure 1 together with the MAP profile.

At layer 1 (physical layer) and 2a (MAC) the *CSMA/CD Access Method* (Carrier Sense Multiple Access with Collision Detection) [12] has been used. In the MAP specification [19] the use of the *Token Passing Bus Access Method* [12] is recommended. The LLC sublayer 2b [12] uses the connectionless, datagram service. At layer 3 (network layer) the connectionless-mode network protocol [9] has now replaced the previously used inactivity subset. The connection oriented transport protocol class 4 [7] is used at layer 4 (transport layer). The implementation of this transport system has been named SINEC H1 (High Performance 1) by Siemens.

The application system is not layered here. The AP Monitor [20, 21] realizes all necessary functions of layers 5, 6 and especially 7.



Layer	SINEC AP 1.0 Profile			MAP Profile		
User	Numeric Control	Robot Control ...	Process Control	Numeric Control	Robot Control ...	Process Control
7b	Automation Protocol (AP) Monitor			ISO Manufacturing Message Specification (MMS) Monitor		
7a	-			ISO Association Control Service Element (ACSE)		
6	-			ISO Connection Oriented Presentation		
5	-			ISO Connection Oriented Session		
4	ISO Transport Class 4			ISO Transport Class 4		
3	ISO Connectionless-mode Network			ISO Connectionless-mode Network		
2b	ISO Logical Link Control Type 1, Class 1			ISO Logical Link Control Type 1, Class 1		
2a	ISO CSMA/CD	ISO Token Passing Bus		ISO CSMA/CD	ISO Token Passing Bus	
1	Baseband	Broadband	Carrier Band	Baseband	Broadband	Carrier Band

Figure 1 : Protocol Profiles for SINEC AP 1.0 and MAP

2.2. The Standardized Protocol Profile MAP

The specification of MAP has been initiated by General Motors in 1980. This project is being accompanied by many other groups or multi vendor projects like the European Map User Group (EMUG) since 1985 or the ESPRIT (**E**uropean **S**trategic **P**rogram for **R**esearch and Development in **I**nformation **T**echnology) project *Communications Network for Manufacturing Applications* (CNMA) since 1986.

MAP is based on the ISO Basic Reference Model. Suitable options have been chosen from existing standardized protocols whenever possible. For the specific application in factory automation a new protocol at the application layer has been specified and standardized in the meantime.

On the lower four layers of our implementation, the same ISO protocols have been used as in SINEC H1. At the layers 5 (session layer) [8] and 6 (presentation layer) [13] of the MAP profile the connection oriented options have been selected. Especially the encoders and decoders of the used syntax at layer 7 are located at layer 6. This syntax is described in *Abstract Syntax Notation One* (ASN.1) [14]. The *Association Control Service Element* (ACSE) [11] is used for connection control by many standardized *Application Service Elements* (ASEs) at layer 7 (application layer), as *File Transfer, Access and Management* (FTAM) [10], *Common Management Information Service Element* (CMISE) [16] or *Manufacturing Message Specification* (MMS) [15], which will be considered later in this paper.

2.3. Field Busses

Realtime requirements lead to protocol profiles, which are different from those depicted in Figure 1. To avoid unnecessary computing power, only the layers 1, 2 and 7b are used. This simplification is possible due to the locality of communication relationships on the lower levels of a company's functional hierarchy. One such profile is the *Enhanced Performance Architecture* (EPA), specified in [19]. Some others have been developed and proposed for standardization by various field bus promotion committees, see [22]. Due to the time constraints, gateways are not adequate tools for communication on these low functional levels.

3. Design of a MAP-Gateway for the Integration of Networks

3.1. The Protocol Architecture

The coexistence of MAP and proprietary protocol profiles like SINEC AP 1.0 will demand for *MAP-Gateways* to interconnect these different networks for the development towards CIM. With the help of a MAP-Gateway, a proprietary installation in a company, as shown at the left side of Figure 2, can be extended by MAP components. In the long term this will allow a migration to a homogeneous MAP network. The protocol architecture of a specific MAP-Gateway to SINEC AP 1.0 is depicted in Figure 3 together with its environment. It will be considered in more detail in the following sections.

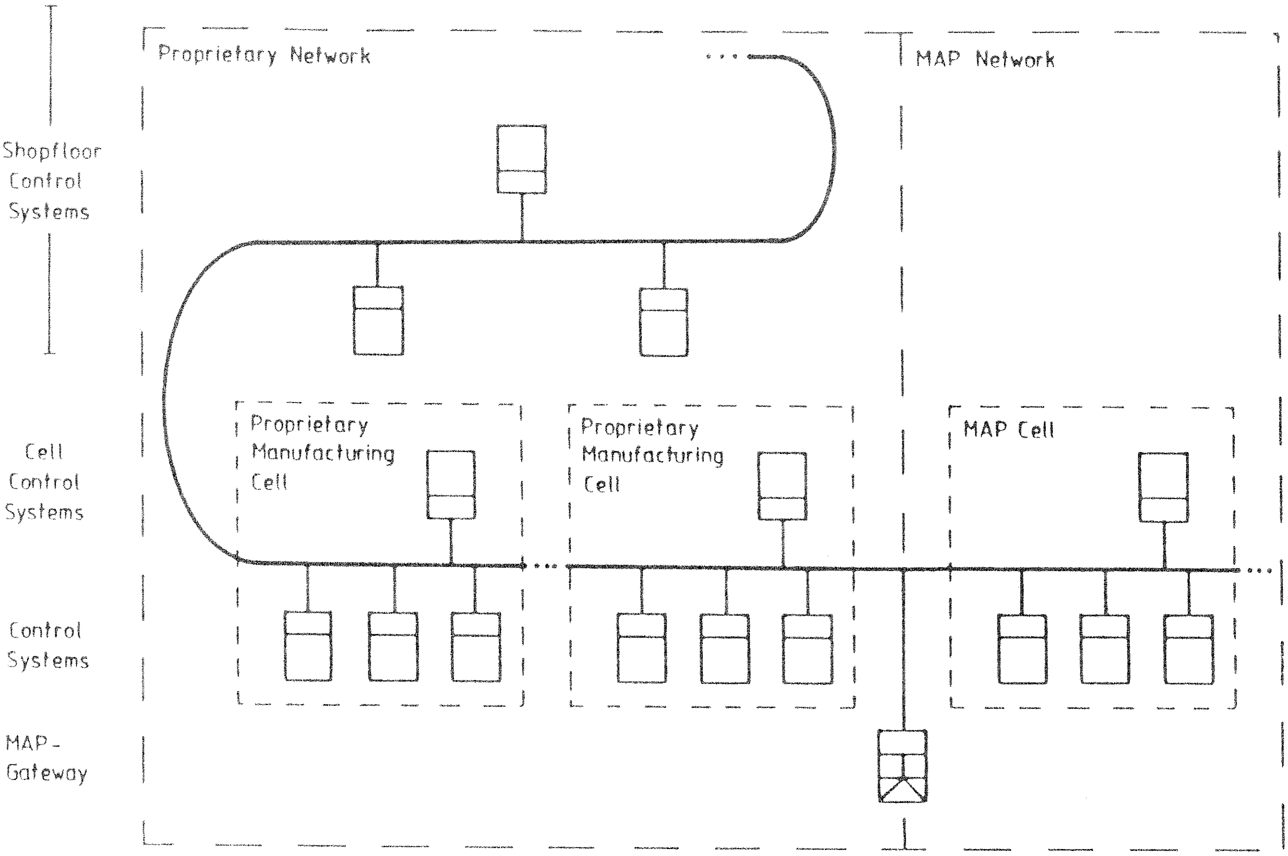

Proprietary Network
MAP Network
Shopfloor Control Systems
Cell Control Systems
Proprietary Manufacturing Cell
Proprietary Manufacturing Cell
MAP Cell
Control Systems
MAP-Gateway

Figure 2 : Integration of Different Networks for Factory Automation

3.2. Hardware Configuration and Operating System

At our institute, we have installed a SINEC H1 LAN with the CSMA/CD media access method. This network uses standardized and identical protocols for both protocol profiles at the lower four layers. The stations in our network (including the MAP-Gateway itself) are multiple Personal Computers (Siemens PC 16–20) and Intel 310 computers with memory extensions and intelligent communication boards.

These boards are handling the lower four layers. Layers 1 and 2a are implemented in hardware (LAN controller 82586) and layers 2b, 3 and 4 in software (Intel processor 80186 with adequate memory). The higher layers have to be handled by the Intel host processor 80286. Due to identical protocols at the lower four layers, it is possible to use physically the same medium for both networks, although there are logically two networks and communication between devices with different protocol profiles is only possible via the MAP-Gateway. It is therefore possible to use only one communication board in the MAP-Gateway and to separate the protocol profiles above the transport system. This configuration can be seen in Figure 3.

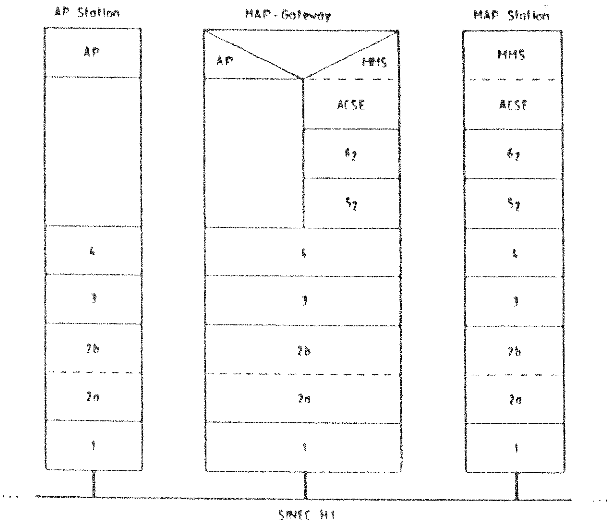


Figure 3 : MAP-Gateway Architecture

If two physically separated networks would have to be interconnected, two communication boards would be necessary in the MAP-Gateway. In practice, it is unlikely to have two identical media installed in parallel and therefore we have implemented the version with one communication board. The host processor has to handle simultaneously both protocol stacks above the transport layer and the protocol conversion software. Parts of version 1.0 of the

MAP–Gateway have been implemented on a Intel 310 computer using the realtime multitasking operating system iRMX 86 from Intel. For version 2.0 with the total MAP–Gateway on one computer a portation to iRMX 286 has been necessary due to memory constraints.

3.3. The SCP Protocol Stack

AP is an application layer protocol, which also includes aspects of the presentation layer and the session layer as far as necessary. One implementation of AP is that for the SINEC Communication Processor (SCP), a communication board for *Numerical Control* (NC) systems [21]. This SCP software has been partly modified and then used in the MAP–Gateway. AP has been defined due to the lack of stable standardized protocols for the higher layers at that time. It will be substituted by standardized protocols as soon as MAP is completely stable. Therefore, only little effort has been made to achieve a layered architecture at the higher layers. In a proprietary environment, the communicating partners and especially their external view of data structures are well–known. Therefore, users create messages in a form, which can be transmitted directly without the need of a presentation layer.

AP in its original version provides services to control and supervise devices, to transfer programs and data, and services for data administration. The services of AP will be more and more adapted in future to the standardized protocol MMS. No standardized transactions or well–defined sequences do exist, but the user himself has the possibility to define them. AP services may either be unconfirmed or confirmed. There is also the possibility to have segmented services with flow control for large messages and a reaction for 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 that the client (active user) will wait for an acknowledgement. If this administration time expires, AP creates a negative acknowledgement for the client. Within the administration time there is the possibility for confirmed services to be repeated when a repeat timer expires. Each service is stored in an administration list at the application layer until the related 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 user). There is a data base for network project data. The memory for the data base is not administrated by the used operating system. The communication paths, that means application channels and transport connections, are static during normal operation. Therefore, the communication relationships have to be projected first, using a special data base editor, to allow connection establishment during the startup phase. With the help of an application relationship table the mapping between application relationships and communication paths is done. A service distribution table allows the assignment of services on the corresponding application modules.

The SCP software has been subdivided into several tasks of the multitasking operating system. The central component is the AP Monitor task, containing the actual protocol. Communication between multiple tasks is done by the exchange of request blocks via mailboxes (MBXs).

3.4. The CNMA Protocol Stack

This work is based on an implementation for the CNMA project [3], which uses a well selected subset of MMS at the application layer. MMS has been developed for the MAP profile as a specific application layer protocol for factory automation. A one to one mapping between context, association and transport connection has been realized in the used software. MMS contexts may be initiated and concluded dynamically during normal operation. It is also possible to establish some basic contexts during startup phase.

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 may only be used by a calling station. Therefore, it can be necessary to initiate a second context between the same entities in the opposite direction in order to fulfil 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 making use of 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 (executable programs), a virtual filestore, variables, event conditions and semaphores to control concurrent accesses to the same resources. 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.

The mapping between application entity titles and real addresses is done by ACSE with a name table, which has been created by a special name table editor.

3.5. The Protocol Conversion Software

Due to different protocols at the application layer, the MAP–Gateway has to realize protocol conversion at the application layer between MMS and AP. This kind of interconnection leads to unavoidable high communication costs, reduced performance and reduced functionality of the interconnected networks.

Service Data Units (SDUs) of one network must be converted into SDUs or sequences of SDUs of the other network. Various possibilities for the parameters of these SDUs 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 may be ignored in the second network, and parameters only used in the second network, which must be provided by the MAP–Gateway.

No additional flow controls are necessary, due to the use of existing application layer flow controls in such a way, that a confirmation to the original sender is returned only, if the MAP–Gateway has received an acknowledgement from the receiver in the case of confirmed services. For unconfirmed services no interconnection of flow controls is intended.

The protocol conversion software has been structured 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 including the *Distribution* function, as depicted in Figure 4.

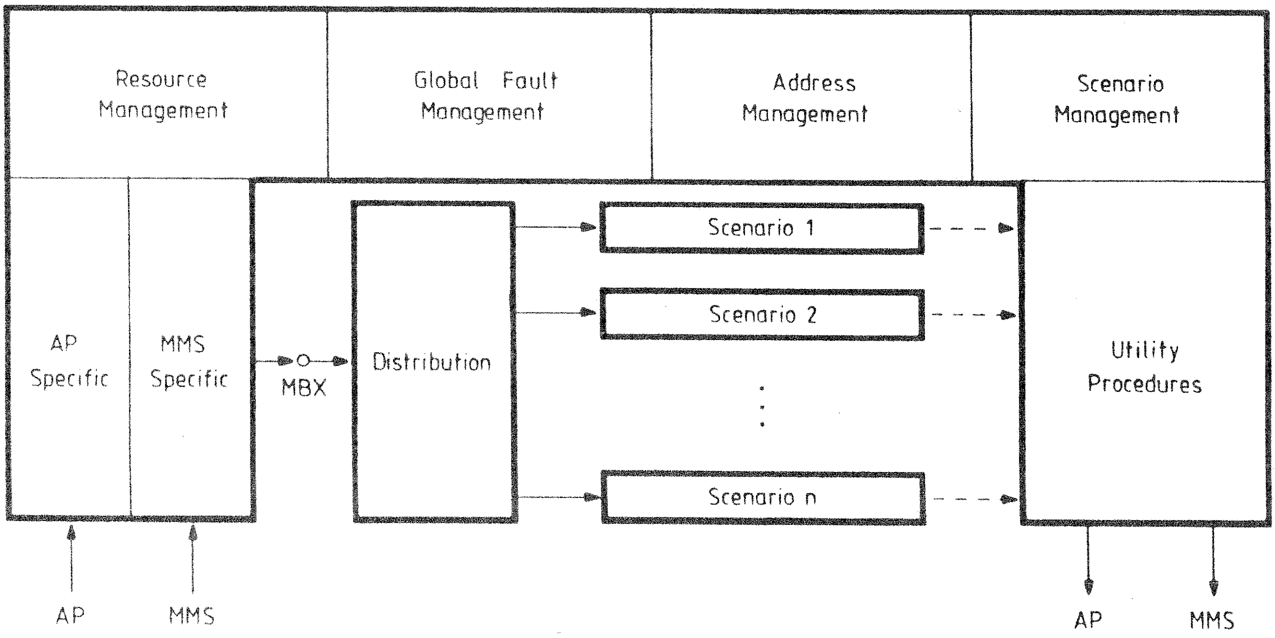


Figure 4 : Functional Areas of the Protocol Conversion Software

The gateway management system has further been subdivided into the following functional areas:

The *AP* and *MMS Specific* parts are responsible for receiving, analyzing and forwarding SDUs from the AP or MMS monitor, respectively.

The *Scenario Management* is responsible for the administration and addressing of active scenarios. Therefore, a dynamic process table is necessary in the protocol conversion software. The service distribution table mentioned in subsection 3.3 has been used here to provide the corresponding scenario class to the AP SDU arrived. An equivalent table has been implemented in the protocol conversion software for arriving MMS SDUs. The scenario addressed by the Scenario Management is then selected by the Distribution function.

The *Resource Management* is responsible for the administration of global resources, especially of memory segments necessary for SDUs to be created or returned. MMS contexts are also administrated as resources with the help of a context list.

SDUs, which are the output of the protocol conversion software, are filled and sent by specific *Utility Procedures*.

Global and scenario independent faults are handled by the *Global Fault Management.* In order to discover duplicates, services not yet completed are entered in tables. Usually negative confirmations, supplemented by a reason code, are returned if a protocol error is recognized.

Addressing is a further interesting aspect in interconnecting different networks at the application layer. As the corresponding peer entity is in the gateway from the point of view of both networks, the gateway instead of the server has to be addressed at the application layer from the client using its tables. In the gateway the addressing is done by the *Address Management.* The existing tables of the SCP and CNMA protocol stacks are used for the mapping between logical names and real addresses, as well as for the addressing of the corresponding scenario class. Additionally a communication table is necessary in the MAP–Gateway for the mapping between AP connections and MMS contexts. This table is filled partly during the initialization phase by reading a projected configuration file, completed with the help of organisatory services and updated at context initiation or conclusion. This communication table also contains context specific informations as CBBs, whether the context is active or passive and whether it is static or dynamic.

4. Implementation Details

4.1. Installation of Two Protocol Stacks on a Single Device

In a first step, the existing protocol stacks had to be adapted to the hardware configuration used. Due to the multitasking operating system it was possible to create a station which is able to use the two protocol stacks simultaneously (that is the MAP–Gateway of Figure 3 without the protocol conversion software), without the need of a further software modification. It proved to be useful to integrate the protocol stacks as first level jobs into the operating system. First level jobs are jobs, which are started in the background automatically after the booting of the pure operating system is finished.

4.2. Implementation of the Protocol Conversion Software

The protocol conversion software was developed in C on a Siemens PC 16–20 under a portation of the operating system iRMX 286 to this PC. It has been structured into three tasks communicating with each other via MBXs, as depicted in Figure 5.

The *Gateway Test Task* has been necessary as a testing environment during development time, substituting future *AP* and *MMS Monitor Tasks.* The *AP* and *MMS Message Handler Tasks* receive and handle AP and MMS SDUs, respectively. The *Gateway Distribution Task* contains the scenario library including the distribution function. Negative confirmations resulting from the Global Fault Management are returned directly by the corresponding Message Handler Task. All other outputs of the protocol conversion software are produced by the Gateway Distribution Task.

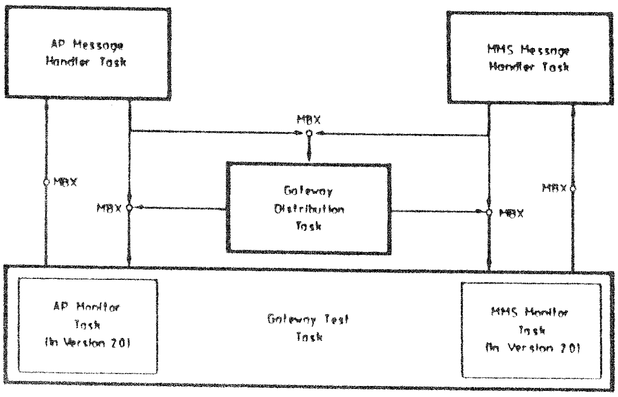

AP Message Handler Task
MMS Message Handler Task
MBX
MBX
MBX
Gateway Distribution Task
MBX
MBX
AP Monitor Task (In Version 2.0)
Gateway Test Task
MMS Monitor Task (In Version 2.0)

Figure 5 : Protocol Conversion Software

4.3. Integration of the MAP–Gateway via a Serial Interface

The integration to version 1.0 of the MAP–Gateway has been done by connecting the two development computers via their serial interfaces. The reason for it was the limited range of addressable memory under iRMX 86, which was not large enough for two protocol stacks plus the protocol conversion software. Therefore, the Gatway Test Task of Figure 5 had to be replaced by an adequate driver, running a protocol with the corresponding driver at the other computer. These drivers include the necessary buffer handling and the creation of the output mask on each screen. An output mask contains information about the conversion direction, the SDUs being handled at both interfaces of the protocol conversion software, the available and used buffer segments, the contents of the communication table and information about the driver (status as well as number and size of buffer segments transferred).

4.4. Integration of the MAP–Gateway on one Device

For version 2.0 the two protocol stacks had to be ported from iRMX 86 to iRMX 286, an operating system with enough addressable memory for the complete MAP–Gateway. Then the MAP–Gateway could be integrated on a single device, as depicted in the configuration of Figure 3. The output on the screen has here slightly been modified.

5. Modelling with Extended Queueing Networks

The modelling has essentially be done by composing two existing extended queueing models, one for of SINEC AP 1.0 and one for MAP, see Figure 6. These models have been presented and explained in [1] and [2]. They contain the usual symbols for queueing networks extended by selfdefined ones where necessary. The symbols for generators, queues, processor phases, processors, decisions, duplications, infinite servers, synchronizations and flow controls are depicted in Figure 7. The horizontal queue of the last symbol contains the credits. If there is at least one credit available, an arriving message in the vertical queue can pass the rhombus without consuming any time, taking with it a credit. If no credit is available, an arriving message has to wait in the vertical queue until a new credit has arrived.

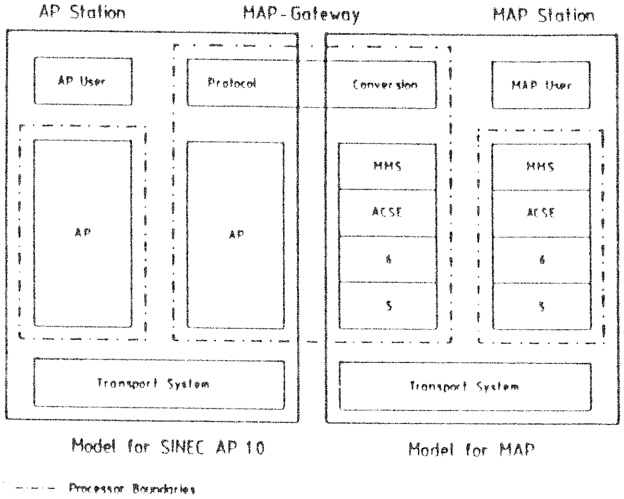


Figure 6 : Modelling of the MAP–Gateway

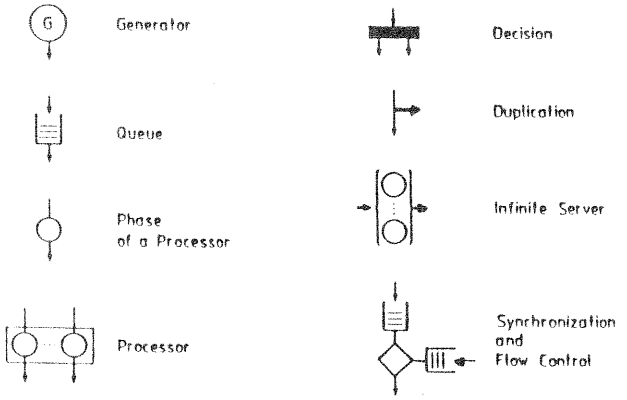


Figure 7 : Symbols for the Modelling

For the simulation of the MAP–Gateway, the two existing simulation programs have been composed to one simulation program. The original users of the stations now forming the MAP–Gateway have been substituted by a new component for protocol conversion. This new component is responsible for the synchronization of file transfer services (modelled equivalent to the flow control mechanism) and it allows the simulation of various scenario classes as 1:1, 1:k or k:1 mapping of SDUs.

The processor phases of the MAP–Gateway above the transport system can be gathered within one processor, reflecting the real implementation of version 2.0, but other possibilities with more than one processor are also possible to investigate the performance improvement. The common resources of the two protocol stacks within the MAP–Gateway are administrated jointly. They have also be modelled with the help of the flow control symbol. The credits represent the resources in this case. The possibilities of having end–to–end (*E-t-E*) or station–to–station (*S-t-S*) acknowledgements have been taken into consideration.

6. Simulation of Various Scenario Classes

6.1. Simulation Technique

The *event by event simulation method* has been used here. The system state is represented by a set of variables and the changes of the system state caused by events are assumed to happen immediately, consuming no simulated time. Therefore, the simulation program processes an event entirely updating all affected system variables and planning possible later ones. After processing an event, the simulation program searches for the next one in the order of time and after updating the simulated time it processes that event. The sequence of events is organized by a *calendar* in which all events which will happen in future are planned.

Simulating the system in this way, measurements for means and coefficients of variation for transfer times, server utilizations or queue lengths can be taken. The simulation is subdivided into 10 independent *part tests* to achieve *confidence intervals* of the measures according to the Student–t–theory.

Due to the complexity of models containing several layers, such models cannot be simulated as a whole. Only one to three layers should be simulated simultaneously. Therefore, the lower layers (transport systems) are aggregated to delay equivalent service centers (infinite servers). The first moments of the distribution function of the infinite servers are obtained by other simulations of the lower layers, by analytical studies or by measurement of real systems. Usually, the first or the first two moments are used for an approximation. So the influence of the lower layers is taken into consideration, but need not be simulated explicitly together with the higher layers.

For this work the usage of general simulation tools with user friendly interfaces would have been very critical, due to the restricted functionality contained in the model components and the relatively long computing time. Therefore, the implementation of the simulation program has been done by using a modular simulation library developed at our institute. The library is a compromise between simulation tools and individually written simulation programs. A pattern of the dynamic data structures and procedures or functions for frequently used tasks in a simulation program as queue handling, server handling, random number generation or statistic evaluation, are included in that library. Carefully implementing the simulation programs and looking for the locality of data guarantees an optimal program for the considered problem with relatively short computing times and acceptable program development times.

Especially for the simulation program presented in this paper, two complete simulation programs for SINEC AP 1.0 and MAP (see [1, 2]) could be used as basis. The resulting program however may only contain one version for the calendar, the input, the statistic evaluation and the output. Each logical name must be selected to be unambiguously. The statistic evaluation and output modules have been extended to evaluate further interesting values. The relatively long transfer times have to be taken into consideration at the dimensioning of timer values or the amounts of credits at the sending station.

To look at the behaviour of the system if the arrival rate is not constant but jumps from a low value to a high and vice versa, the stationary simulation program has further been extended to allow also an instationary simulation. The following main extensions have been done:

- The control of the simulation program has been extended to handle not only part tests, but also measurement intervals and elementary tests.
- The generator has to allow jumps in the arrival rate, which have been realized with the help of additional calendar events.
- Statistic evaluation and output have been extended to obtain multiple points (total curves) instead of one.

6.2. Simulation Results

Stationary simulation results with realistic assumptions for the parameters (see [1, 2]) are depicted in Figures 8 to 15. The generator produces traffic according to a negative exponential distribution of the interarrival time. The curves are depicted in the stable range of the arrival rate caused by the generator. The simulation points are marked and 95% confidence intervals show the accuracy of the simulation. The *total transfer time* is defined as the time from the generation of a message to the arrival at the server application. The *buffer occupation time* is defined as the time from the generation of a message to the arrival of the acknowledgement at the client application. The results have been obtained for one channel and one context at the application layer. In the sending station 20 credits are available for the channel individual flow control, due to long buffer occupation times, so that the credits do not represent a bottleneck of the system. Therefore, the bottleneck can be found at the processor with the largest cycle time, usually in the MAP–Gateway. Due to the scheduling strategy used, handling older messages with a higher priority than younger messages, the bottleneck can be found at the entrance of the MAP–Gateway. Timers in AP are adjusted such, that they do not expire during normal operation.

The abbreviations *US* for unconfirmed service and *CS* for confirmed service will be used in the following. For most results (with the exception of Figure 12) the implementation of version 2.0 with one processor in the MAP–Gateway for the SCP protocol stack, the CNMA protocol stack and the protocol conversion software has been simulated. With the exceptions of Figures 11, 13 and 14 the direction from MAP to AP is depicted. Differences to the opposite direction may be neglected if not stated otherwise. In Figures 8 to 10 different service types are considered for a 1:1 mapping in the MAP–Gateway.

In Figure 8 US has been used. The boarder of stability is reached at 15 messages per second. The buffer occupation time is relatively low and nearly constant, due to the buffer release by the transport confirmation for US. The slight increase for high load results from waiting times within the MAP station. This effect could not be observed in the opposite direction due to the simpler AP station which therefore does not represent a bottleneck.

In Figure 9 CS has been considered with station–to–station confirmation. This confirmation is produced by the user in the receiving station of each network. Therefore, the buffer occupation time is significantly higher than in Figure 8. For the opposite direction it would still be smaller than the total transfer time because of the missing layers 5, 6 and 7a at the AP side.

Especially in the MAP station, the increased service time due to the additional confirmation leads to a lower boarder of stability. For an empty system the same total transfer time can be observed as for US, due to no influence of that additional confirmation on the total transfer time for an empty system.

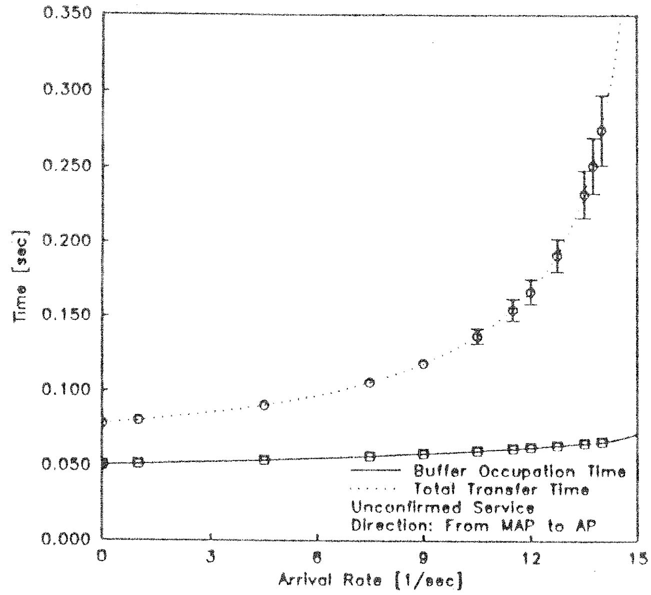


Figure 8 : Transfer and Buffer Occupation Times (US, MAP → AP)

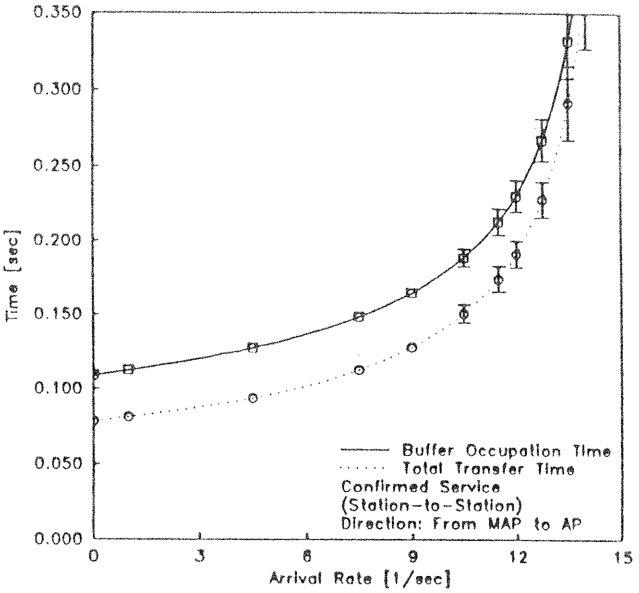


Figure 9 : Transfer and Buffer Occupation Times (CS, S-t-S, MAP → AP)

Usually data from the receiving station are expected in confirmations. Therefore, the acknowledgement mechanism has to work with an end–to–end significance. This is depicted in Figure 10, leading to a further increased buffer occupation time.

After the comparison of service types, only CS will be considered in the following results.

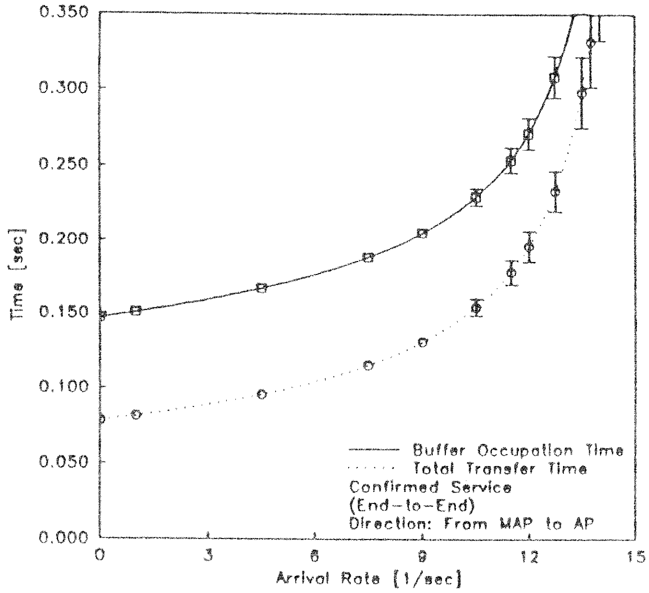


Figure 10 : Transfer and Buffer Occupation Times (CS, E-t-E, MAP → AP)

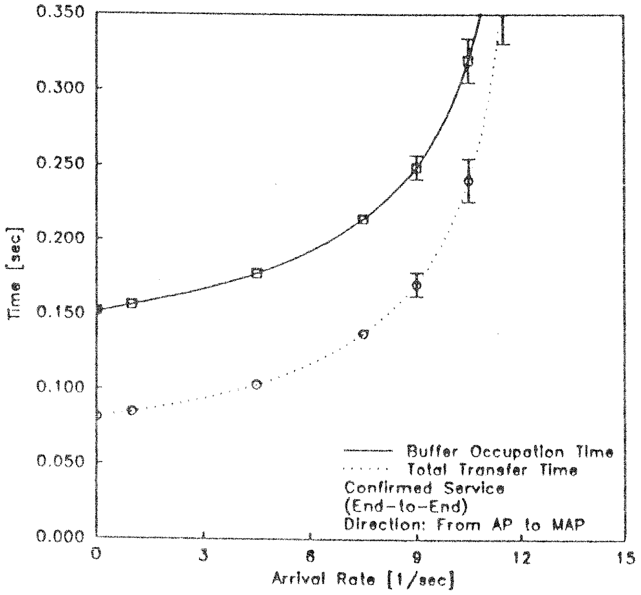


Figure 11 : Transfer and Buffer Occupation Times (CS, E-t-E, AP → MAP)

Figure 11 shows the results of Figure 10 for the opposite direction. The starting values are the same due to an identical set of phases to be passed by the messages and no occuring waiting times. The boarder of stability has decreased due to the bigger overhead in the MAP–Gateway caused by the transport confirmation at the MAP side. This could also be observed for the opposite direction for Figure 9.

In Figure 12 the MAP–Gateway has been simulated containing three processors, one for the SCP protocol stack, one for the CNMA protocol stack, and one for the protocol conversion software. The residual parameters have been chosen as for Figure 10. The bottleneck is the processor of the MAP station here, which has to handle the sending part of the CNMA protocol stack. Total transfer time and buffer occupation time are relatively constant in the order of 100 ms and 150 ms, respectively.

In Figure 13 a scenario of the class 1:k has been considered for the direction from AP to MAP. A realistic example for this class is the data transfer in both directions for k=2. Whereas in AP this can be done with a single service, in MMS the services *Write* and *Read* have to be used sequentially. Due to the more complex scenario, the buffer occupation time has increased significantly compared to Figure 11. The boarder of stability is reached earlier due to the doubled load at the MAP side. The performance would slightly improve, if the higher load were at the AP side.

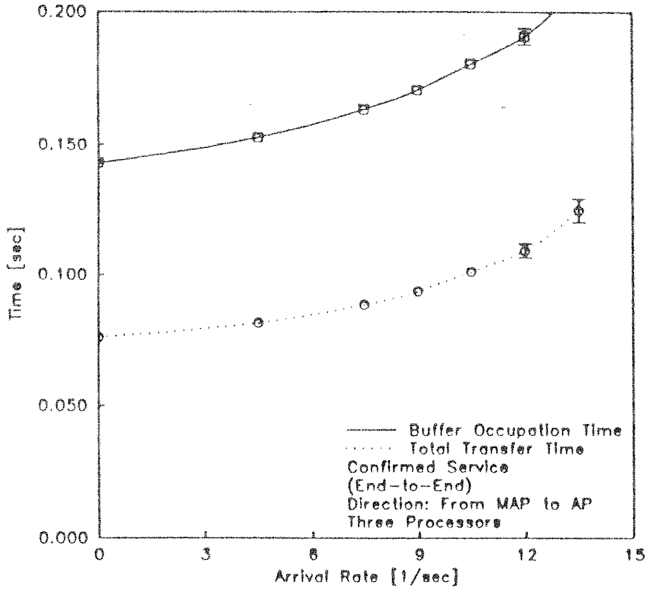


Figure 12 : Transfer and Buffer Occupation Times (Multiple Processors)

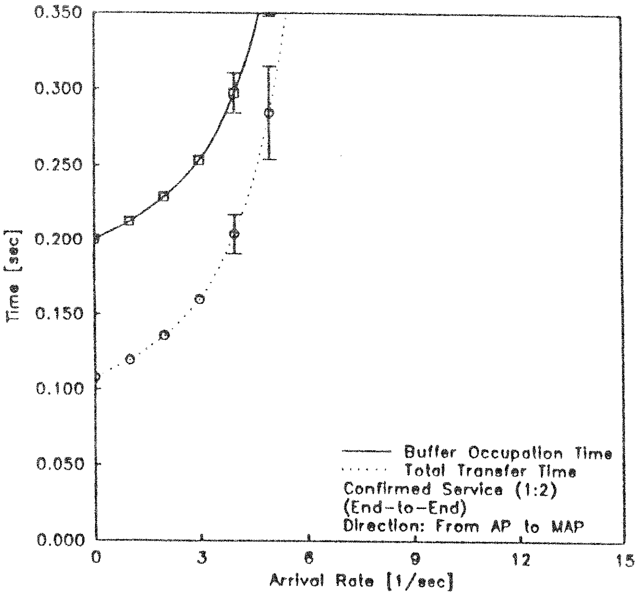


Figure 13 : Transfer and Buffer Occupation Times (1:2, AP → MAP)

Another scenario class is the k:1 mapping in the MAP–Gateway. Figure 14 shows a result for k=5 with a window size of w=3 for the segments of the segmented service used. As a realistic example the transmitting of a big amount of data from AP to MAP may be considered. Whereas in AP the message size is fixed and a big amount of data must be segmented, the buffer size in MMS is more flexible in our implementation. The data of all intermediate segments are stored in the MAP–Gateway until the last segment has arrived. Then they are mapped onto one message for the MAP side. Due to the reduced load at the MAP side, the boarder of stability is increased significantly. The buffer occupation time is a mean value of the roundtrip delays for intermediate segments (CS, station-to-station) and the last segment (CS, end-to-end).

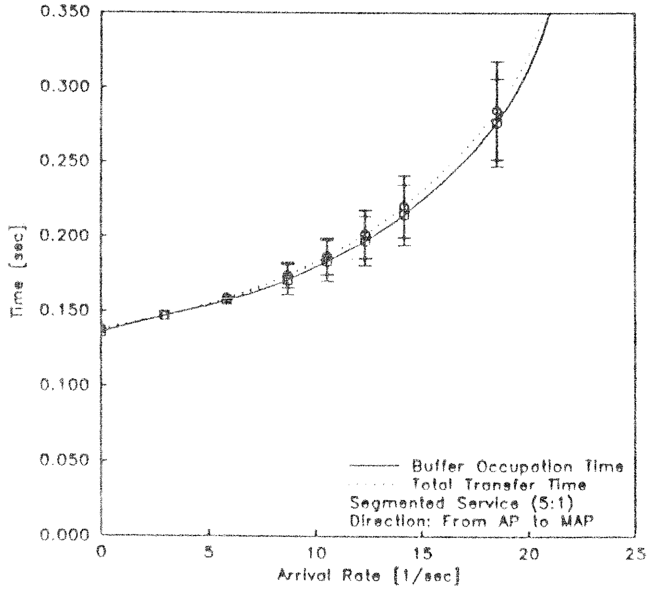


Figure 14 : Transfer and Buffer Occupation Times (5:1, AP → MAP)

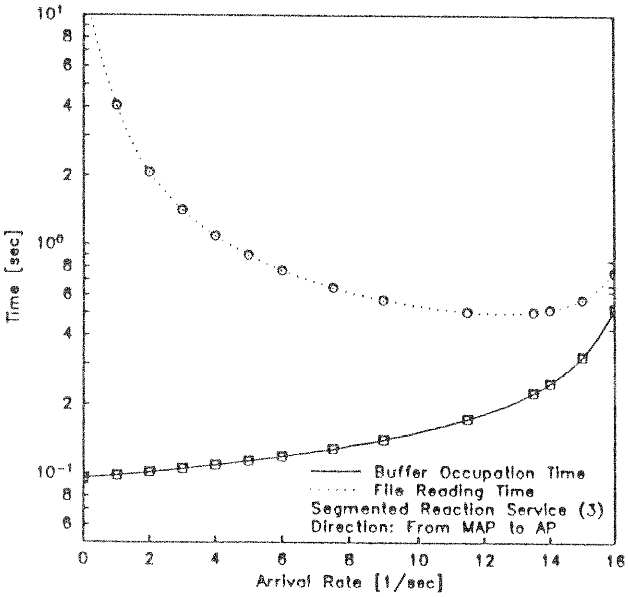


Figure 15 : Reading and Buffer Occupation Times (File: AP → MAP)

In Figure 15 the file transfer has been considered. Here, a significant difference can be found for the two application layer protocols AP and MMS. In MMS (appendix in [15]) the file transfer is controlled by the reading user. He has to submit a sequence of the services *FileOpen*, multiple *FileReads* and *FileClose*. In the original version of AP the file transfer is usually controlled by the owner of the file. After receiving a reading command, the file owner sends the file as a segmented reaction service (window size w=1). All segments have to be confirmed to the file owner. In Figure 15 results are depicted for a MAP station which wants to read files consisting of three segments. After a *FileOpen* service, the MAP-Gateway plays a passive role only. From the MAP station *FileRead* services arrive and from the AP station the file segments. These segments have been modelled as credits of a flow control, controlling the *FileRead* commands and thus realizing a synchronization mechanism. The file reading time has been defined as the time from the generation of a *FileOpen* service until the confirmation to the corresponding *FileClose* service has arrived at the MAP station. For a low arrival rate this file reading time is very long due to the big gap between the *FileRead* services of the same file. With an increasing arrival rate this gap will grow smaller and the file reading time decreases. At a high load the MAP-Gateway becomes congested and the file reading time increases again due to increased waiting times in front of processor phases in the MAP-Gateway.

Figures 16 and 17 show instationary results to Figures 10 and 15, respectively. The horizontal straight lines repeat the stationary results for the low and high arrival rates. The first vertical straight line marks the time of a jump from the low to the high arrival rate and the second vice versa. The fundamental behaviour of the buffer occupation and total transfer times over the simulated time has found for all scenario classes to be as depicted in Figure 16. A smooth increase of these times at the first jump and a quick decrease at the second could be observed. Immediately after the jump from the high to the low arrival rate, the high stationary value can be exceeded up to 50 % due to an increased ratio of acknowledgements to messages at that time, which increases the total transfer time. The buffer occupation time shows the same behaviour due to the increased total transfer time and a nearly constant acknowledgement time.

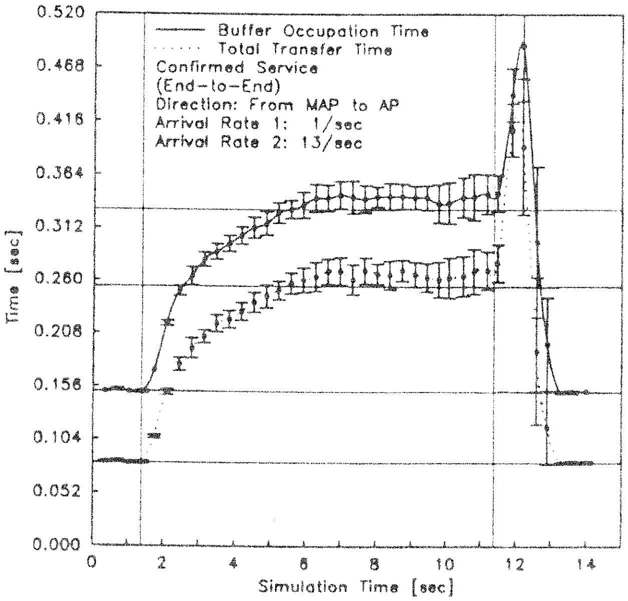


Figure 16 : Transfer and Buffer Occupation Times (Instationary)

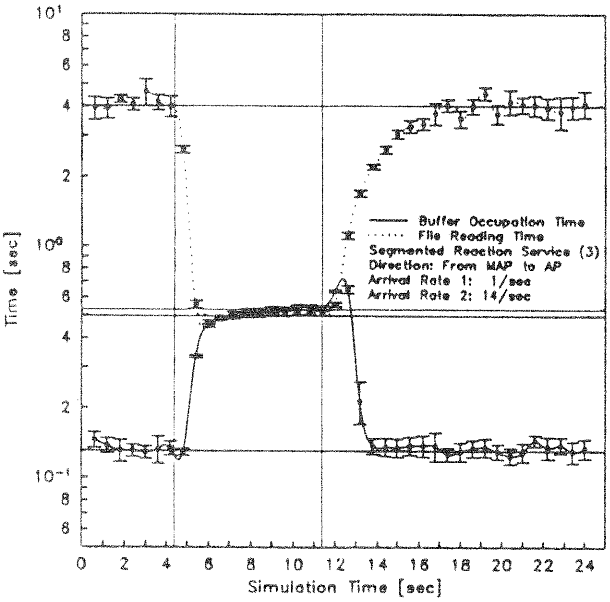


Figure 17 : Reading and Buffer Occupation Times (Instationary)

Figure 17 shows the same effects for the buffer occupation time, whereas the file reading time decreases for high arrival rates, see also Figure 15. Immediately after the first jump, the file reading time reaches a minimum due to decreased interarrival time of *FileRead* services and a nevertheless relatively empty system.

7. Conclusions

Design and implementation of two versions of a MAP-Gateway have been described. This MAP-Gateway will further be developed to a product version by Siemens. It will be presented in the framework of a CNMA pilot installation in Stuttgart this year. Furthermore the modelling and simulation of that MAP-Gateway has been explained, accompanied by a parameter study. Simulation model and program have been developed using two existing simulation models and the corresponding programs for two protocol profiles for factory automation. Various scenario classes within the MAP-Gateway could be investigated to predict its performance for a stationary and instationary behaviour of the generator. Currently we are extending this MAP-Gateway to allow an overall network management by central manager stations.

Acknowledgements

The author would like to thank Georg Rößler and all the students who have been involved in the implementation and the simulation of the MAP-Gateway.

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] M. Bosch, O. Gihr; "Simulation of Application Layer Protocols for Factory Automation", International Conference on Computer Communication (ICCC) 1988, Tel Aviv, October 30 – November 4, 1988

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

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

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

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

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

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

[9] ISO DIS 8473; "Protocol for Providing the Connectionless–mode Network Service", May 1987

[10] ISO 8571; "File Transfer, Access and Management", September 1987

[11] 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

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

[13] ISO DIS 8822; "Connection Oriented Presentation Service Definition", June 1986
ISO DIS 8823; "Connection Oriented Presentation Protocol Specification", June 1986

[14] ISO 8824; "Specification of Abstract Syntax Notation One (ASN.1)", May 1987
ISO 8825; "Specification of Basic Encoding Rules for Abstract Syntax Notation One (ASN.1)", May 1987

[15] ISO 9506; "Manufacturing Message Specification, Part 1: Service Definition", January 1989
ISO 9506; "Manufacturing Message Specification, Part 2: Protocol Specification", January 1989

[16] ISO DIS 9595; "Common Management Information Service", September 1988
ISO DIS 9596; "Common Management Information Protocol", September 1988

[17] ISO DP 10028.2; "Information Processing Systems – Definition of the Relaying Functions of a Network Layer Intermediate System", August 1989

[18] ISO TR 10029; "Information Technology – Telecommunications and Information Exchange between Systems – Operation of an X.25 Interworking Unit", March 1989

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

[20] SINEC; "SINEC AP 1.0, Teil 1: Protokollspezifikation", Siemens AG, 30. Juni 1987
SINEC; "SINEC AP 1.0, Teil 2: SINEC AP Technologische Funktionen, Teil A: Dienstbeschreibung", Siemens AG, November 1988
SINEC; "SINEC AP 1.0, Teil 2: SINEC AP Technologische Funktionen, Teil B: PDU Beschreibung", Siemens AG, November 1988
SINEC; "SINEC AP 1.0, Teil 3: AP–Protokollabwickler", Siemens AG, Januar 1988

[21] SINEC; "SCP1–Software Entwicklungsunterlage", Siemens AG, 23. Januar 1989

[22] G. Wood; "Current Fieldbus activities", Computer Communications 11, June 1988, pp. 118 – 123