

Performance Modelling of a Highly Modularized Packet Switching Node

Manfred N. Huber, Wikhard M. Kiesel, Paul J. Kuehn, Thomas Raith
Institute of Communications Switching and Data Technics
University of Stuttgart
Fed. Rep. of Germany

Gunther W. Kerschner
Siemens AG
Fed. Rep. of Germany

A highly modularized, high speed network node for wide area packet networks is considered. The control is subdivided into four levels: Termination Unit (TU) containing Line Terminator Unit (LTU) and Terminator Group Controller (TGC) and the Switching/Routing Unit (SU) containing Switching Processor Controller (SPC) and Switching Processor Unit (SPU). The TUs and SUs are interconnected through a pipelined ring controlled by a special access protocol using a ring empty indicator. The exchange of packets between the LTU, TGC, SPC and SPU modules is controlled by internal protocols which provide for buffer reservation, acknowledging and proper retransmission in case of packet loss due to buffer overflow. The modular concept allows a gradual extension up to 255 TU/SU system units. The network node has been modelled by an extensive queueing network reflecting all important resources, protocol features and statistical aspects. The model has been analyzed by means of simulation as well as analytically. For the simulation of such a complex problem an aggregation technique has been developed and implemented where some modules and the central ring are represented in full detail, whereas the residual modules are replaced by simpler models, however with the identical load generation. The analytical performance evaluation is based on the decomposition of the total model into submodels of the type of a multiple-level processor with feedback and priorities; these models are analyzed in isolation and the global results are found from the various submodel results. The performance evaluation of the network node yields results on the maximum nodal throughput and the cross-switch packet delay under different load conditions and allows the proper sizing of components and the evaluation of cross-network packet delays.

1. PACKET SWITCHING NODE

1.1. Switch Structure

The increasing need for packet communications leads to a second generation of packet switching nodes to be introduced in wide area packet networks c.f. [1,2]. This second generation of nodes is characterized by a multiplicity of modules which are interconnected by high-speed internal busses or rings. Fig. 1 shows the basic structure of such a node consisting of 4 global functional blocks:

Line Termination Unit	LTU	} Termination Unit TU
Terminator Group Controller	TGC	
Ring Unit	RU	
Switching Unit	SU	

The LTUs interface the switching node to the subscriber lines and internodal trunks; they perform basically the functions of levels 1 and 2 of the ISO-Basic Reference Model. The traffic of several LTUs is multiplexed onto one TGC. The TGCs communicate with the more centralized SUs through the central RU. The SU consists of two parts, the Switching Processor Unit SPU and the Switching Processor Controller SPC. The SPCs interface the more centralized SPU with the RU. The SPU performs the basic packet switching functions whereas the SPCs offload the SPU from time critical transmission procedures.

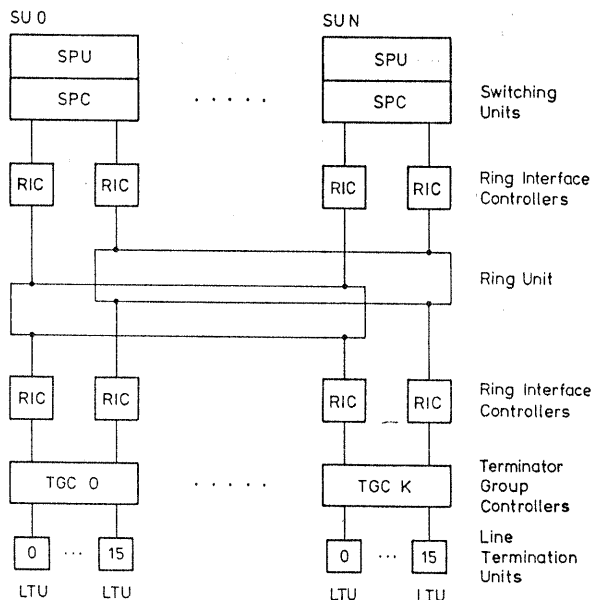


Fig. 1 Basic structure of a modularized packet switching node

The whole packet switching node consists of many SUs, TGCs and LTUs. Up to 16 LTUs may be connected to one TGC. Each SU handles the traffic

of several LTUs which are logically assigned to that SU. The logical assignment may be changed in case of breakdown of a SU.

Virtual connections through the switch are established by a setup procedure involving two (originating/destinating) LTUs and their corresponding SUs. Once a virtual connection is established, succeeding data packets are routed through the switch from the incoming LRU to the outgoing LRU via one of the SUs being assigned to the connection setup phase.

1.2. Switch Operation

Fig. 2 shows the basic internal routing scenarios for a virtual connection setup and a subsequent data transfer.

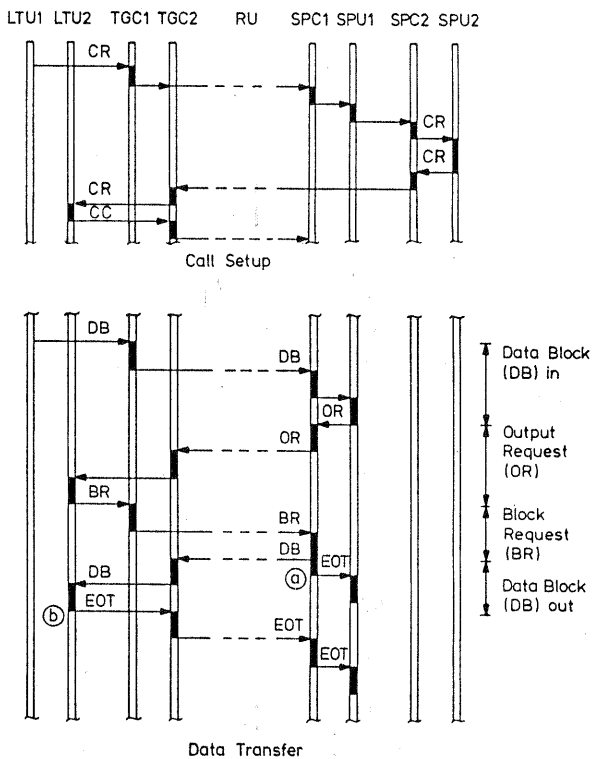


Fig. 2 Connection management

The virtual connection setup starts with a Call Request (CR-) Packet (being carried within a CR-Message) from the originating LRU 1 which is forwarded to the corresponding SPU 1 through TGC 1 and SPC 1. The SPU 1 performs the routing function and forwards the CR-Packet to the destinating SPU 2 through SPC 2 within a corresponding CR-Message. SPU 2 sends the CR-Message via SPC 2 and TGC 2 to LRU 2. The Call Confirmation (CC-) Packet runs as a CC-Message from LRU 2 the opposite way back across the system.

Within the data transfer phase (c.f. Fig. 2) a data packet is transferred from LRU 1 to SPU 1 as a Data Block (DB-) Message through TGC 1 and SPC 1. The SPU 1 buffers the data packet in the Main Memory (MM) and sends an Output Request (OR-) Message to LRU 2 which is acknowledged by LRU 2 with a Block Request (BR-) Message. Upon reception of the BR-Message the SPC 1 transfers out the data packet from the Main Memory of the SPU within a DB-Message. Finally, there are two

different options for the internal acknowledgement of the data packets transferred out: In mode a, (LT-buffered Mode) the data packet is cleared within the SPU 1 directly after successful sending of the DB-Message across the RU, whereas in mode b, (CP-buffered Mode) the data packet is cleared after reception of an End of Transmission (EOT-) Message from LRU 2.

1.3. Objectives

The operation of the highly modularized packet switching node is characterized by parallel packet processing within the various units and a pipelined packet transmission over the central RU. The performance of the switch, i.e. the throughput and delay characteristics, depends largely on the individual processing and transmission times as well as on the various queueing delays caused by the resource sharing principle. The estimation of the throughput capabilities and cross-switch delays has therefore been subjected to an extensive performance evaluation study through modelling, simulation and analytical calculations. The results of the performance evaluation may be used for a proper

- configuration management
- sizing of buffer and processor capacities
- implementation of load-adaptive schedules
- throughput estimation
- cross-switch and cross-network delay estimation.

2. MODELLING

2.1. Derivation of the Switch Model

In this section a model of the central part, i.e. Switching Processor Unit (SPU), Switching Processor Controller (SPC), Ring Unit (RU) and Terminator Group Controller (TGC) of the high-speed packet node will be derived. The different units are modelled by their processing and scheduling phases. The data and control packets may wait for processing in the queues corresponding to these phases. All queues will be served according to the FIFO service discipline. Furthermore, an arbitrary scheduling cycle is defined for each particular unit to serve the waiting packet requests by taking the various load situations into account.

In the following, the various component submodels are shortly motivated. The global switch model is then composed from these component submodels.

2.2. Terminator Group Controller

The TGC is essentially made up from one microprocessor single server with 4 processing and 1 scheduling phase, see Fig. 3. The phases correspond to the following functions:

- phase 1 : transfer from TGC-ring receive queue to LRU-send queue (DMA)
- phase 2 : transfer from LRU-send queue to LRU (DMA)
- phase 3 : transfer from TGC-transmit queue to TGC-ring send queue (DMA)
- phase 4 : transfer from LRU-receive queue to TGC-transmit queue
- phase 5 : scheduling.

To account for a proper emptying of the high-speed

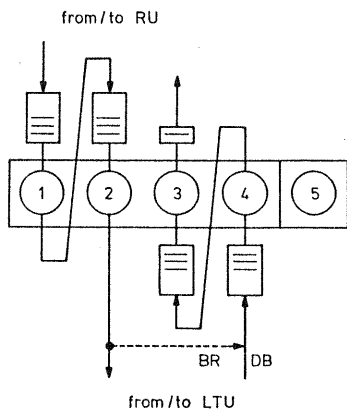


Fig. 3 Submodel of the Terminator Group Controller (TGC-Submodel)

ring receive and send queues, phases 1 and 3 have nonpreemptive priority, whereas phases 2 and 4 are scheduled for service cyclically. Phase 5 represents the operating system overhead. The scheduling strategy can be given arbitrarily. In case of nonempty queues, the cycle may be 1 3 5 1 3 2 1 3 5 1 3 4 ...

The phase service time distributions can be arbitrarily fixed e.g., according to measured data of the implemented switch. Typically, the service time is composed of a fixed part (corresponding to DMA initialisation) and a variable part corresponding to the actual packet length.

2.3. Switching Processor Controller

The SPC consists of two microprocessors corresponding to the receive and transmit functions with respect to the high-speed ring. Both modules are shown in Figs. 4a,b.

The Receive-SPC (RSPC) submodel (Fig. 4a) is a single server model with 6 service phases corresponding to

- phase 1 : EOT-transfer from RSPC-EOT queue to SPU-EOT queue
- phase 2 : BR-transfer from RSPC-BR queue to TSPC-BR queue
- phase 3 : EOT-transfer from RSPC-EOT-CP queue to RSPC-EOT queue
- phase 4 : DB-transfer from RSPC-DB queue to SPU-queue and main memory
- phase 5 : transfer from RSPC-ring receive queue to the queue belonging to the message
- phase 6 : scheduling.

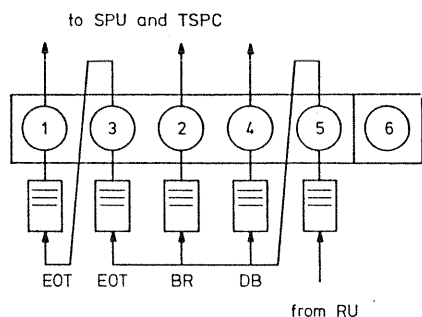


Fig. 4a Submodel of the Switching Processor Controller (RSPC-Submodel)

Similarly to the TGC-Submodel, phase 5 is granted nonpreemptive priority. Scheduling and service time distributions may be fixed analogously as described in Section 2.2.

The Transmit-SPC (TSPC) submodel (Fig. 4b) is also a microprocessor single server with 5 phases:

- phase 1 : transfer from TSPC-transmit queue to TSPC-ring send queue (DMA)
- phase 2 : OR-transfer from TSPC-OR queue to TSPC-transmit queue
- phase 3 : BR-transfer from TSPC-queue to TSPC-transmit queue
- phase 4 : BR-transfer from TSPC-BR queue to TSPC-queue
- phase 5 : scheduling.

Phase 1 has nonpreemptive priority. All other details are similar as described above.

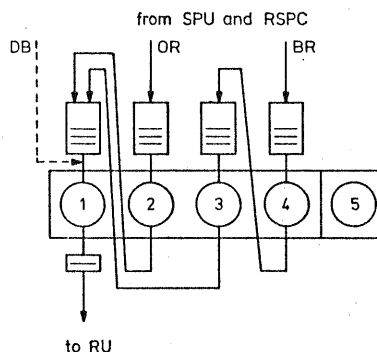


Fig. 4b Submodel of the Switching Processor Controller (TSPC-Submodel)

2.4. Switching Processor Unit

The SPU performs the packet switching functions and is modelled by a microprocessor single server system with 2 processing phases, see Fig. 5. The phases correspond to

- phase 1 : Routing of CR-Messages
Switching of DB-Messages
OR-transfer to the TSPC-OR queue
- phase 2 : Processing of EOT-Messages

In addition to the phase-specific input queues a third buffer models the intermediate storage of packets within the Main Memory. Each packet buffered in the Main Memory will be released upon processing of the corresponding EOT-Message. The scheduling is organized according to a clocked mode by which phase 2 is initiated periodically after a specified time.

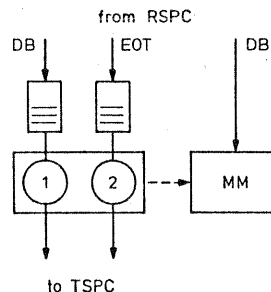


Fig. 5 Submodel of the Switching Processor Unit (SPU-Submodel)

2.5. Ring Unit

The RU operates according to a pipelined ring system. As such, a modified single server model is used where the scheme for the initiation of a message transmission is modelled by a polling mechanism and where the pipelined transmission process is modelled by scheduled arrival times at the various receive queues. This submodel is schematically depicted within the global switch model (Fig. 6).

2.6. Global Switch Model

The global model of the switch is shown in Fig. 6 representing explicitly only one TGC-Submodel, one SPU-Submodel and one pair of RSPC- and TSPC-Submodels. In the real model, an arbitrary number of these representatives can be given, e.g., 50 SUs, 50 RSPCs, 50 TSPCs and 200 TGCs.

The individual packet flow through the global switch model follows principally according to the scenario of Fig. 2, i.e., there are two types of messages corresponding to connection setup and data transfer. The resulting total packet traffic flow follows from a given origination - destination traffic matrix and can be arbitrarily unbalanced.

The packet traffic is generated by external traffic generators which are simply modelled as Poisson processes as a good approximation for the superimposed traffics of the various LTUs (these are not included in the central switch model).

3. PERFORMANCE EVALUATION TECHNIQUES

3.1. Analytical Performance Evaluation

The analytical performance evaluation is based on decomposition techniques where the submodels of Section 2.1 are analyzed in isolation under certain "environment" conditions, i.e., under simplified interface traffic assumptions. For the typical assumption of Poisson (Markovian) arrivals pure priority models and pure polling models can be analyzed exactly or approximately with respect to mean delay based on renewal theory and Little's theorem [3-7]. The models of Fig. 3-5 are somewhat more complicated; they can approximately be analyzed by an aggregation technique for all equally treated processing phases. These methods are currently under study and will not be described here explicitly.

3.2. Simulative Performance Evaluation

Opposite to the analytical approach, simulation is an experimental performance evaluation technique. The simulation model is implemented by a computer simulation program where the system state is represented by a complex data structure. Messages are represented by requests (jobs, customers) which require transmission or processing from service units (servers). They are generated by some traffic generators with individual inter-arrival time distributions. The generation of new requests and the service terminations are represented by discrete-time system events. The simulation is organized according to a time sequenced event list which is processed sequentially by the simulation program (time true simulation, event-by-event simulation). Within a simulation run, usually hundreds of thousands of

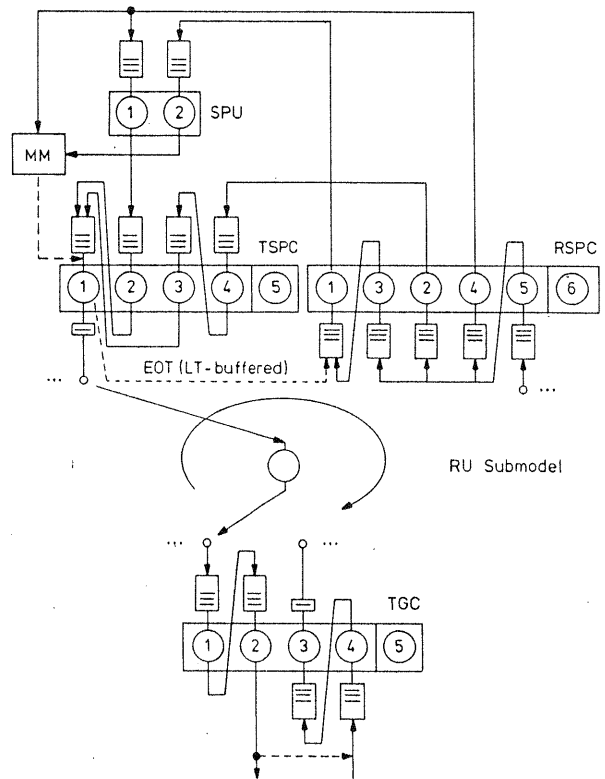


Fig. 6 Global Switch Model

events are executed and measurements are taken simultaneously. By averaging over the various sample measurements representative results on the throughput, delay or occupation of servers and queues are found.

Simulation, however, has significant drawbacks when it is applied to models as the example of Fig. 6. The simulation execution time depends linearly on the number of executed events. In order to obtain reliable results for a highly modular system as that of Fig. 6, many millions of events have to be executed resulting in excessive CPU-times. Therefore, various simplified simulation techniques which are less time-consuming have been applied:

a) Subsystem Simulation

In this method, only part of the global model is simulated as, e.g., RU-Submodel, TGC-Submodel, SPC-Submodel, or SPU-Submodel. This method allows a correct representation of the subsystem-scheduling and service mechanisms, but suffers from the lack of inter-message dependencies which are caused by the node-internal routing scenarios, see Fig. 2.

b) Global System Simulation with Aggregated Submodels

In order to correctly represent the node-internal routing scenarios, the submodels of Fig. 3-5 may be aggregated to simple single server models with an aggregated total service time. Fig. 7 shows this simplified global model where the aggregated submodels consist of a single server queueing model in the receiving direction and a send queue in the sending direction. The aggregated submodels are

interconnected by the RU-Submodel as introduced in Section 2.5. The drawback of this method is that the specific priority mechanisms are neglected.

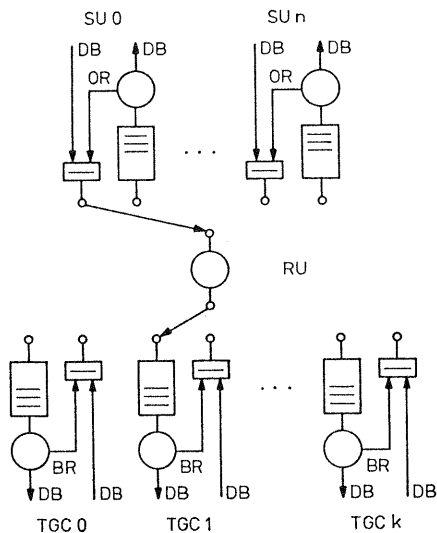


Fig. 7 Simplified Global Model

c) Combined Method

Methods a) and b) suggest themselves a combined method where one or a few complex submodels are explicitly introduced in the global model whereas the mass of the residual submodels are of the aggregated type. This method maintains the submodel specific characteristics, the inter-submodel scenarios as well as the global traffic flows correctly. Trans-nodal characteristics as, e.g., the cross-switch packet transfer delays, can be measured accurately by selecting the particular paths including the explicit submodels, only. In conclusion, the combined method allows the implementation of a much less complex system model and reduces the number of events to be executed drastically.

4. SIMULATION RESULTS

In the following results obtained by the combined simulation method will be presented and discussed for the case of a symmetrically loaded packet switching node. The simulation results will be depicted with their 95% confidence intervals. For the results presented the node is considered to consist of 6 SUs and 8 TGCs. Packets transferred via the RU will be acknowledged according to mode a), i.e. LT-buffered mode. In the simulation the following message lengths are assumed: DB-Message = 40 words, OR-Message = 10 words and BR-Message = 10 words. All queues and the main memory have finite storage capacity. Fig. 8 shows the mean packet transfer delay versus the offered data packet rate of each TGC. As expected due to the processing and waiting times in the SPU the mean transfer delay for data blocks via the ring to the Main Memory (DB in) is much larger compared to the delay of data blocks transferred out (DB out). BR-Messages will not be processed by the SPU, this results in a smaller transfer delay. In Fig. 9 the carried load of the different submodels is shown as a function of the offered packet rate per TGC. The graphs show, that with an increasing packet rate the load increases linearly.

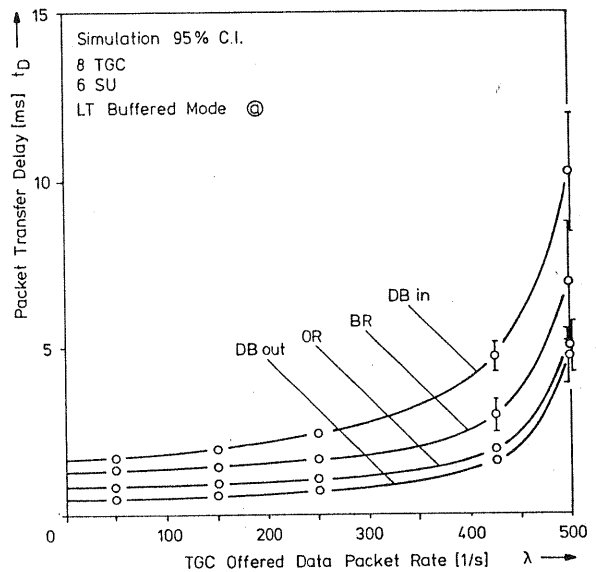


Fig. 8 Mean packet transfer delay vs TGC offered data packet rate

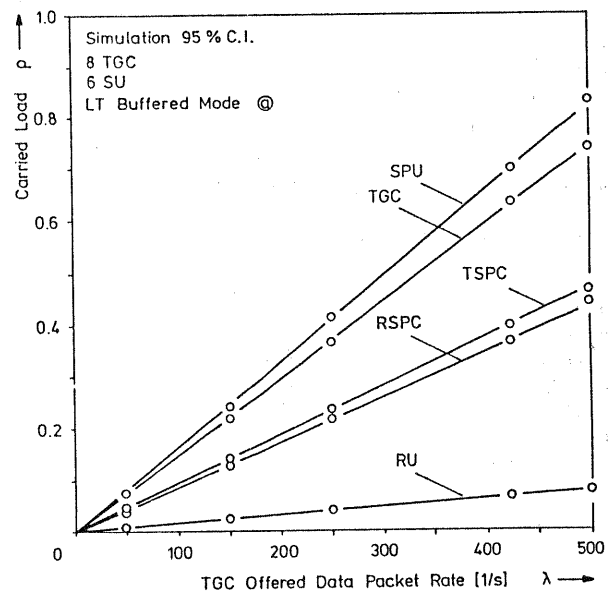


Fig. 9 Carried load vs TGC offered data packet rate

5. CONCLUSION AND OUTLOOK

In the paper, modelling of a highly modularized node for packet switching networks is outlined. The obtained extensive queueing network reflects all important resources, protocol features and statistical aspects of the network node. To evaluate the performance in terms of mean packet transfer delay or carried load of the different units, the queueing model of the packet node is simulated by a special technique where the whole model of the network node is subdivided into detailed subsystem models. By aggregation of the subsystems to simple single server models a global system simulation can be performed in order to correctly represent the node-internal routing

scenarios. Finally a combined simulation applying aggregation techniques and subsystem considerations provides performance measures and may be a basis for the decision process in the development and engineering of such complex packet switching nodes. Currently, the simulation of the LTU is being implemented where some aspects of the data link protocol are also taken into account as well as an analytical performance evaluation of the whole packet switch.

ACKNOWLEDGEMENT

The authors would like to thank Mr. T. Denzel for his programming efforts.

REFERENCES

- [1] E. Mair, H. Hausmann, R. Naessl, "EWSP - A High-performance Packet Switching System", ICC '86, Munich, paper B7-2.
- [2] J.F. Huber, A. v.Kienlin, "Functional and Performance Requirements in Future Packet Switching Networks", ISS '84, Florence, paper 43A-2.
- [3] A. Cobham, "Priority Assignment in Waiting Line Problems", Operations Res. 2 (1954), pp. 70-76.
- [4] U. Herzog, "Priority Models for Communication Processors Including System Overhead", 8. ITC, Melbourne (1976), paper 62-3.
- [5] P.J. Kuehn, "Approximate Analysis of General Queueing Networks by Decomposition", IEEE Trans. Commun., vol. COM-27, pp. 113-126, Jan. 1979.
- [6] H. Takagi, L. Kleinrock, "Analysis of a Polling System with Priorities", Japan Science Institute Research Report, 1985.
- [7] P.J. Kuehn, "Multiqueue Systems with Nonexhaustive Cyclic Service", Bell Syst. Tech. J. 58 (1979), pp. 671-699.



Paul J. Kuehn received the Dipl.-Ing. and Dr.-Ing. degrees in EE from the University of Stuttgart, Germany in 1967 and 1972, respectively. From 1973 to 1977 he was Head of a group for traffic research in computer and communications systems at the University of Stuttgart. In 1977 he joined Bell Laboratories in Holmdel, NJ, where he was working in the field of computer communications.

In 1978 he was appointed full professor for Communications Switching and Transmission at the University of Siegen, Germany. Since 1982 he holds the chair of Communications Switching and Data Technics at the University of Stuttgart, Germany.

Prof Kuehn is member of IEEE, ACM, NTG (German Communications Society), and GI (German Informatics Society). He has been appointed as Member of the Communications Switching Committee of NTG, the Computer Communications and Computer Performance Committees of GI/NTG, the IFIP W.G. 7.3., and Vice

Chairman of the Int. Advisory Council of the Int. Teletraffic Congress (ITC).

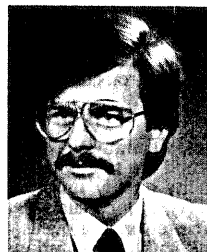


Manfred N. Huber was born in 1959. He studied electrical engineering at the University of Stuttgart, F.R.G., where he received the degree of 'Diplom-Ingenieur' in 1984. Since this time he has been with the Institute of Communications Switching and Data Technics at the University of Stuttgart. Actually, he works in the field of integrated in-house communication, performance of packet switching and broadband switching.



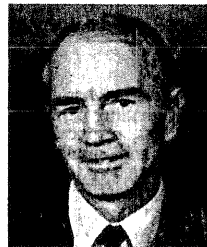
Wikhard M. Kiesel was born in Schwaebisch Hall, West-Germany in 1954. He received the Dipl.-Ing. degree in electrical engineering from the University of Siegen, Siegen, West-Germany in 1980. From 1980 to 1983 he has been a Research Associate at the Department of Communications, University of Siegen, working in the field of local area networks. From 1983 to 1985

he has been with the Institute of Communications Switching and Data Technics, University of Stuttgart. In 1985 he joined the Division for Energy and Automation Technology of Siemens AG in Erlangen, West-Germany, where he is currently with the Systems Engineering Development Department. His main interests are now communication for factory automation and network/system management.



Thomas Raith received the M.S. degree (Dipl.-Ing.) in Electrical Engineering from the University of Stuttgart, Germany, in 1981. Since 1982 he has been a scientific staff member at the Institute of Communications Switching and Data Technics at the Univ. of Stuttgart, where he is involved in the performance investigation of computer communication systems.

T. Raith is member of NTG (German Communications Society), and GI (German Informatics Society).



Gunther W. Kerschner received an Ing (Eng) degree from the Hoehere Technische Lehranstalt of Vienna, Austria. He joined Siemens AG in 1963 and first worked on hardware and test-software development for the electronic data switching system EDS. Later he was responsible for testphilosophie and led a group developing testsoftware for board- and systemtests. Presently he is

involved in the development of a packet switching system and there he is responsible for hardware and software.