

Auf ähnliche Weise, wie in den skizzierten Szenarien dargestellt, lassen sich eine Vielzahl anderer bedeutender Messungen durchführen. Die beschriebenen Beispiele sollten zur Abrundung der gegebenen Übersicht jedoch ausreichen.

4. Zusammenfassung

In den vorangehenden Abschnitten wurde anhand einer Reihe von Merkmalen und Beispielen der LAN Protocol Tester B5100 charakterisiert. Nach der Beschreibung einer typischen Meßkonfiguration, der Systemarchitektur, der Funktionsgruppen, alternativer Vorgehensweisen bei der Durchführung einer Messung, der Bedienoberfläche und einiger Besonderheiten des LAN Protocol Testers B5100, wurden anhand dreier Meßszenarien die Einsatzmöglichkeiten skizziert.

Als Fazit bleibt festzuhalten: Mit seinem umfangreichen Funktionsspektrum ist der LAN Protocol Tester B5100 nicht nur zur Unterstützung des Aufbaus und der Inbetriebnahme von LANs geeignet, sondern auch für die Überwachung, Wartung, Fehlerdiagnose und zur Unterstützung der Engpaßanalyse/Ausbauplanung während des Betriebes.

Modulare Simulationstechnik für komplexe Anwendungsprotokolle in der Fertigungsautomatisierung

Martin Bosh, Ottmar Gühr
Universität Stuttgart
Institut für Nachrichtentechnik und Datenverarbeitung
Seidenstraße 36, 7000 Stuttgart 1

Wilhard Kiesl
Siemens AG, Unternehmensbereich Energie- und Automatisierungstechnik
Hauptbereich Systemtechnische Entwicklung (B STE 366)
Günther-Scharowsky-Straße 2, 8520 Ehlangen

Für die Kommunikation in der Fertigungsautomatisierung wird die Leistungsfähigkeit der MAP-Protokollarchitektur mit einer weiteren Protokollarchitektur für heterogene Automatisierungsverbunde verglichen. Nach der Modellbildung wird jeweils ein Simulationsprogramm, mit Unterstützung einer Simulationsbibliothek, die modular aufgebaut ist, implementiert. Für typische Anwendungsszenarien und Lastverhältnisse wird die Leistungsfähigkeit der beiden Protokollarchitekturen dargestellt.

1. Einleitung

Für die computerintegrierte Fertigung (CIM, Computer Integrated Manufacturing) ist der Informationsfluß in einer heterogenen Umgebung verschiedenster Rechensysteme eine wesentliche Voraussetzung. Die Funktionalität der Kommunikation von Rechensystemen ist nach dem ISO-Referenzmodell (International Organization for Standardization) für offene Systeme [2] in 7 Schichten eingeteilt und wird durch Protokolle geregelt. Für die Kommunikation von Rechensystemen und Steuerungen verschiedener Hersteller bedarf es der Standardisierung dieser Protokolle durch ein internationales Gremium und der Anerkennung und Anwendung dieser Standards durch die Hersteller. Nachdem die Protokolle standardisiert sind, werden für die einzelnen Anwendungsgebiete die Optionen und Alternativen festgelegt, die bei den Protokollen auf den Schichten verwendet werden sollen. Bei der Fertigungsautomatisierung hat sich auf die Initiative von General Motors seit dem Jahre 1980 das "Manufacturing Automation Protocol" (MAP) herausgebildet, nachdem die Notwendigkeit einer einheitlichen Kommunikation durch schnell wachsende Kosten für Protokollumsetzungen erkennbar wurden. Die Spezifikation der MAP-Protokollarchitektur liegt inzwischen in der momentan gültigen Version 3.0 [19] vor. Die vorhergehende Version 2.1 wurde für einige Zeit eingefroren, um eine gewisse Kontinuität zu sichern.

Da die internationale Standardisierung noch nicht in allen Bereichen einen stabilen Zustand erreicht hat, werden derzeit weitere Protokollarchitekturen mit dem Ziel entwickelt, als Übergangslösung zu dienen, um zu gegebener Zeit in einen Standard einzumünden oder mit in die Standardisierung einzufließen.

Bild 1 zeigt die MAP-Protokollarchitektur und eine weitere Protokollarchitektur für heterogene Automatisierungsverbände (AP, Automation Protocol). Auf den Schichten 1 (Physical) und 2a (MAC, Media Access Control) wird das Token Passing Bus [12] und das CSMA/CD (Carrier Sense Multiple Access with Collision Detection) [11] Medienzugangsverfahren eingesetzt. Die Schicht 2b (LLC, Logical Link Control) [10] verwendet den unquittierten Datagrammdienst (LLC Type 1 Procedure). Die Schicht 3 (Network) enthält bei diesen Protokollarchitekturen keine Funktionalität bezüglich der Vermittlung von Paketen, da die Kommunikation innerhalb eines "lokalen Netzes" (LAN, Local Area Network) ablaufen soll. Auf der Schicht 4 (Transport) wird die Klasse 4 des verbindungsorientierten Transportprotokolls [3,4] verwendet. Für die MAP-Protokollarchitektur enthalten die Schichten 5 (Session) [5,6] und 6 (Presentation) [13,14] bisher nur Kernfunktionen. Insbesondere ist auf der Schicht 6 der Encoder und der Decoder für die auf der Anwendungsschicht verwendete Syntax, die in ASN.1 (Abstract Syntax Notation One) [15,16] beschrieben ist, anzudeuten. Die Schicht 7 (Application) enthält je nach Einsatzfall ein oder mehrere sehr spezielle Anwendungsprotokolle. Der Teil, der allen diesen Anwendungsprotokollen gemeinsam ist, wird als Association Control Service Element (ACSE) [8,9] bezeichnet und ist separat standardisiert. Für die Fertigungsautomatisierung wird neben FTAM (File Transfer, Access and Management) [7] von der Electronic Industries Association (EIA) ein Standard für den MMS-Monitor (Manufacturing Message Specification) vorbereitet, dessen Draft 6 die ISO als "Second Draft Proposal 9506" [17,18] angenommen hat. Für die alternative Protokollarchitektur enthalten die Schichten 5 und 6 keine eigene Definition. In der Schicht 7 wird ein spezieller Automation Protocol (AP) Monitor [1,20,21] eingesetzt, der die Funktionalitäten der Schichten 5 und 6, soweit sie für LANs in einem heterogenen Automatisierungsverbund benötigt werden, mit bereitstellt.

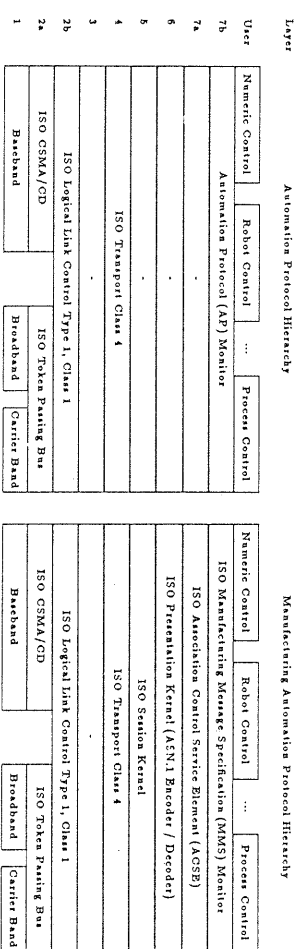


Bild 1 : Protokollarchitekturen für die Simulationen

Abweichend zu der MAP-Spezifikation wird, um die Leistungsfähigkeit der Protokollarchitekturen besser vergleichen zu können, auf den Schichten 1 und 2a für die Untersuchung vom Medienzugangsverfahren CSMA/CD ausgegangen.

Im zweiten Abschnitt wird die Modellierung der Protokollarchitekturen, hauptsächlich in der Anwendungsschicht, erläutert. Nach der Beschreibung der Simulationstechnik im dritten Abschnitt werden im vierten Abschnitt die Ergebnisse dargestellt und diskutiert.

2. Modellierung

Die Protokollarchitekturen aus Bild 1 werden für die Simulation in Modelle abgebildet. In den

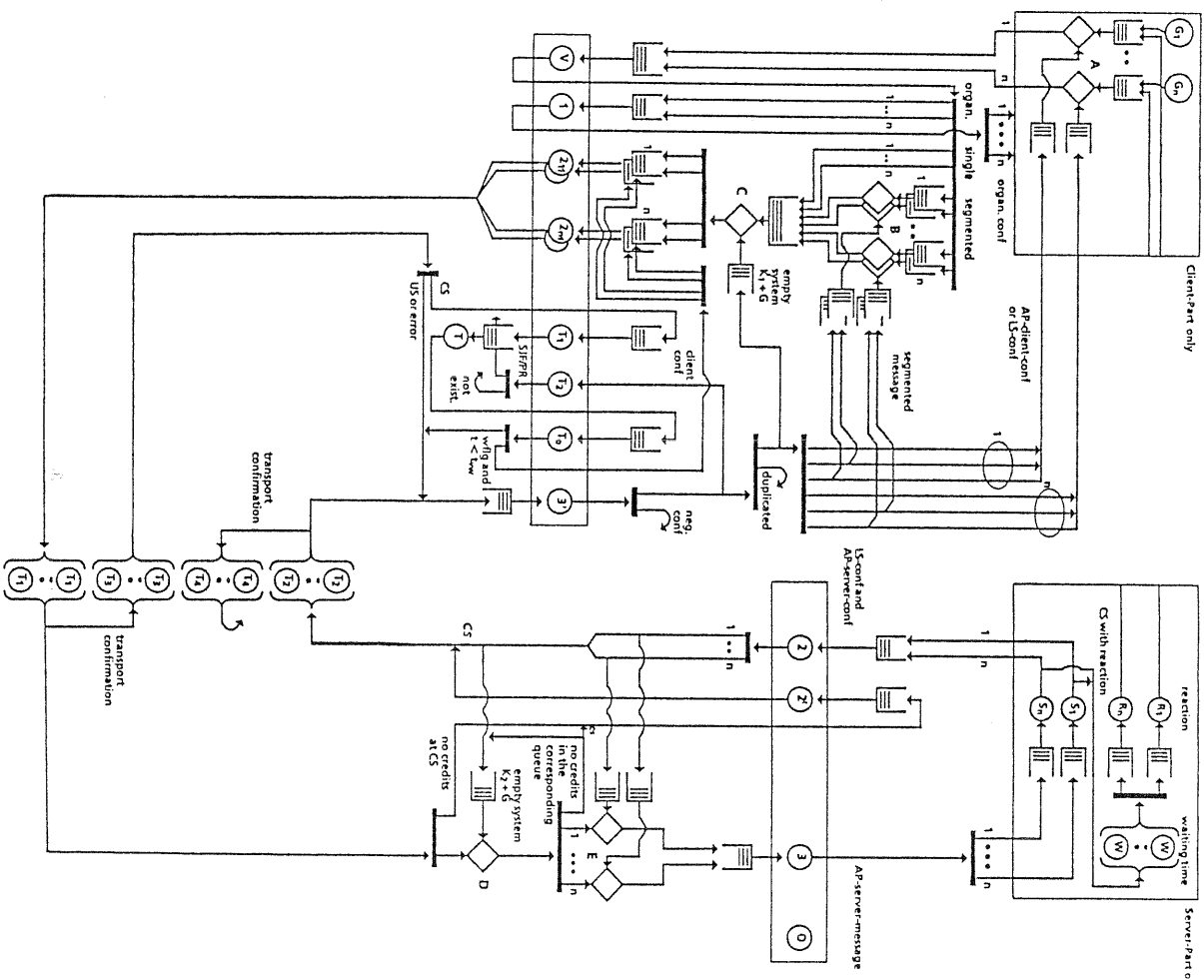


Bild 2 : Modell der AP-Protokollarchitektur

Erklärungen	Symbol	Legende	AP-Parameter	MMS-Parameter
	 	n logische Multiplex-Kanäle Entscheidung Duplizierung	1/2/10	1
Flußkontrollen	A	Kanalindividuell (Client) oder Contextindividuell (Requestor) Anzahl der Credits (für US) Anzahl der Credits (für CS) Segmentierte Aufträge (Fenster) Fenstergröße	1/2/5/10 1/2/5/10	3/10/1000 1/2/5/10
	B	Begrenzung gemeinsamer Ressourcen (Client oder Requestor) K ₁ Credits nur für C G Credits für C und D	2	
	C	Begrenzung gemeinsamer Ressourcen (Server oder Responder) K ₂ Credits nur für D	10 30	10 30
	D	Begrenzung gemeinsamer Ressourcen (Server oder Responder) G Credits für C und D (s.o.) K ₂ Credits nur für D	10 30	10 30
	E	Kanalindividuell (Server) oder Contextindividuell (Responder)	1/2/5/10	10
Abkürzungen	ACSE AP conf CS LS S/JF/PR T t tw US wflg	Association Control Service Element Automation Protocol Quittung Auftrag mit Quittung Lokale Schnittstelle Spezielle Ableitungsstrategie Hardware-Timer Zeit seit der Auftragsgenerierung Verwaltungszeit Auftrag ohne Quittung Wiederholungsflag		
Monitor-Phasen	O 1 2 2' 2 ₁ 3 3' 3'' I T ₁ T ₂ T ₀ V	Für organisatorische Aufträge oder für M-Open Für Positive Quittungen vom User Für Negative Quittungen vom User Für Aufträge vom User Für Aufträge zum User Für Quittungen zum User Für Transport-Quittungen zum User Für M-Await Initiate Zum Starten eines Timers Zum Stoppen eines Timers Für Timeout eines Timers Verteilphase	1 ms 3 ms 2,5 ms 2,5 ms 5 ms 7,5 ms 4 ms 1,5 ms 3,5 ms 3,5 ms 1,5 ms	1 ms 1 ms 3 ms 3 ms 3,5 ms 3 ms 3 ms 0,5 ms 5,5 ms
Sonstige Phasen	O A ₁ F ₁ R R R ₁ R ₂ S ₁ T ₁ T ₂ T ₃ W	Overhead-Phasen ACSE-Phasen Presentation-Phasen Responder-Phase Reaktionsauftragsphasen Server-Phasen oder Session-Phasen Transportverzögerung (Auftrag) Transportverzögerung (Quittung) Transportverzögerung (T-Quittung) Wartephase bis zum Reaktionsauftrag	5 ms 5 ms 5 ms 5 ms 5 ms 5 ms 7,5 ms 7,5 ms 500 ms	1,5 ms 1 ms 1,5 ms 5 ms 5 ms 3,5 ms 10,5 ms 7,5 ms 7,5 ms

Tabelle 1 : Parameter der Simulation und Legende zu Bild 2 und 3

Bildern 2 und 3 sind die oberen Schichten 5–7 detailliert dargestellt. Die unteren Schichten 1–4 werden zusammengefaßt (aggregiert) und durch eine Verzögerungszeit repräsentiert, die sich aus der detaillierten Modellierung und durch Analyse/Simulation dieser Schichten ergibt.

Diese Modelle sind so gegliedert, daß sich die aktive Station (Client oder Requestor) auf der linken Seite und die passive Station (Server oder Responder) auf der rechten Seite befindet. Von jeder Station ist nur der für diese Zuordnung benötigte Teil dargestellt. Zu dem Modell einer vollständigen Station kommt man, indem man die beiden Teile in Gedanken miteinander verbindet. Das Transportsystem, das alle Schichten von der Transportschicht abwärts umfassen soll, ist in der Mitte unten als ein "Infinite Server" dargestellt. Es wird hier genau eine Transportverbindung betrachtet. Die Parameter für die Phasen des "Infinite Server" können mit Hilfe einer separaten analytischen/simulativen Untersuchung oder durch Messung an einem realen Transportsystem ermittelt werden. Der Prozessor in Bild 2 gehört zum AP-Monitor, während in Bild 3 der oberste Prozessor Teil des MMS-Monitors ist. Oberhalb der Anwendungsschicht befindet sich die Anwendung (User), die Aufträge sendet oder empfängt. In Bild 2 wird jedem Generator G, ein Multiplex-Kanal zugeordnet. Die Generatoren können zu einer oder mehreren Anwendungen gehören. Dagegen ist in Bild 3 nur ein "Context" der Anwendungsschicht dargestellt. Aufträge mit und ohne Quittung werden hier von verschiedenen Generatoren generiert und separat flußkontrolliert. Implementierungsdetails sind einer Pilot-Implementierung im Rahmen des ESPRIT (European Strategic Program for Research and Development in Information Technology) Teilprojektes Communications Network for Manufacturing Applications (CNMA) [22] entnommen.

Mit den Rauten werden Flußkontrollen dargestellt. In den horizontalen Warteschlangen sind die Credits enthalten. Jeder Auftrag nimmt beim Passieren der Raute ein vorhandenes Credit mit, oder er kann die Raute nicht durchlaufen, solange die horizontale Warteschlange leer ist. Es werden folgende Flußkontrollen verwendet:

- 1) Verbindungsindividuelle Flußkontrollen A und E (bei Bild 3 ist A noch aufgeteilt für Aufträge mit und ohne Quittung),
- 2) Fenstermechanismus B für segmentierte Aufträge,
- 3) Flußkontrollen C und D wegen der Begrenzung gemeinsamer Ressourcen.

Bei Bild 2 erreichen alle Aufträge den AP-Monitor über eine Verteilphase V. Diese Phase hat die niedrigste Priorität des Prozessors, damit bei Engpässen die Last aus der Station herausgehalten werden kann. Dort wird der Auftragstyp erkannt und je nachdem, ob es sich um einen organisatorischen Auftrag, um einen einzelnen Auftrag oder um einen segmentierten Auftrag handelt, zu einer speziellen Warteschlange verzweigt. Nach der Ankunft der Transportquittung wird weiter unterschieden, ob es sich um einen Auftrag mit oder ohne Quittung handelt. Im ersten Fall muß ein Timer (Phase T₁) gestartet werden, während im zweiten Fall eine lokale Quittung erzeugt wird. Beim Server gibt es noch die Möglichkeit, daß es sich um einen Auftrag mit Reaktionsauftrag handelt. In diesem Fall wird nach einer Wartephase W ein Reaktionsauftrag erzeugt, der auch wieder segmentiert sein kann. Jetzt wird der Server zu einem Client, der diesen Auftrag aussendet.

Bei Bild 3 erreichen alle Aufträge den MMS-Monitor ebenfalls über eine Verteilphase V. Diese Phase hat wieder die niedrigste Priorität des Prozessors, damit bei Engpässen die Last aus der Station herausgehalten werden kann. Dort wird der Auftragstyp erkannt und je nachdem, ob es sich um ein M-Open handelt oder nicht, zu einer speziellen Warteschlange verzweigt. Nach der Flußkontrolle C

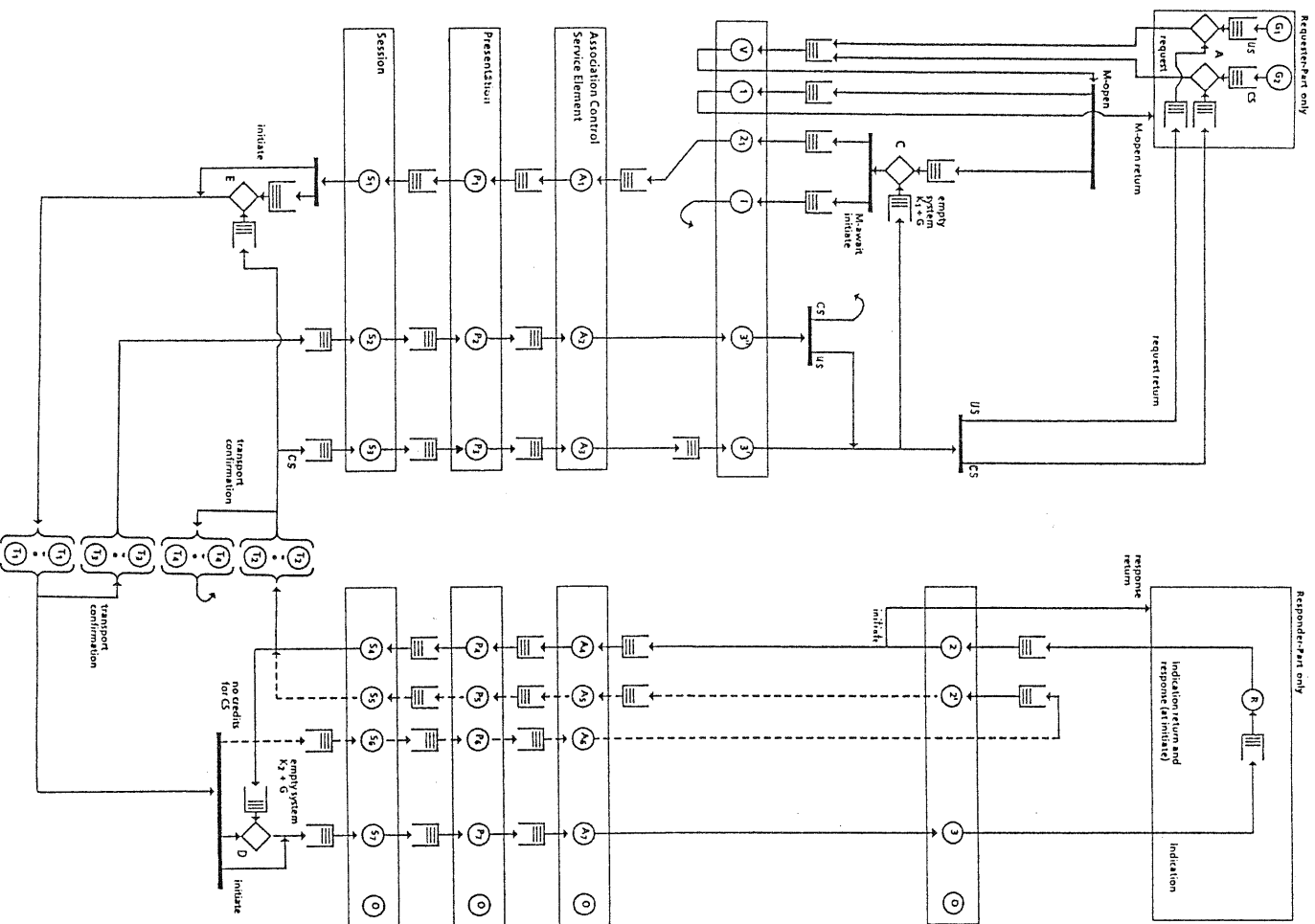


Bild 3 : Modell der MAP-Protokollarchitektur

(Belegen von Ressourcen) muß noch der Auftrag M-Await Initiate besonders behandelt werden, da er nur lokale Bedeutung hat und nicht über das Netz übertragen wird. Auf Timer wird bei MMS verzichtet, da von einer sicheren Transportverbindung ausgegangen wird, die, auch im Fehlerfall, zumindest eine negative Quittung zurückgibt. Die verbindungsindividuelle Fluktkontrolle E arbeitet hier, im Gegensatz zu Bild 2, so, daß sich die Aufträge bei geschlossenem Fenster bereits auf der Requestor-Seite stauen. Die Schichten 5, 6 und 7a sind als Prozessoren modelliert, die lediglich Verzögerungsphasen enthalten. Im Simulationsprogramm können alle Prozessoren der Schichten 5 bis 7 zu einem Prozessor zusammengefaßt werden.

3. Simulationstechnik

Es wird die ereignisgesteuerte zeitreue Simulationstechnik (Event by Event) verwendet. Die Zeit zwischen Ereignissen wird dabei übersprungen und das Ereignis zu dem Zeitpunkt des Eintretens vollständig bearbeitet. Alle Zustandsänderungen im Modell, wie Bedienungsende eines Auftrags, Timeout, neuen Auftrag durch den Generator (Client oder Requestor) erzeugen, u.s.w., werden als Ereignisse aufgefaßt, die, gesteuert über den Kalender, zu den Zeitpunkten ihres Eintretens bearbeitet werden. An dem so simulierten Modell werden die Messungen der interessierenden Leistungsgrößen durchgeführt. Diese Leistungsgrößen können sein: Transferzeiten, Prozessorauslastungen, Warteschlangenlängen oder Bedienzeiten.

Da ein Modell übergreifend über alle Schichten wegen der großen Komplexität nur mit sehr hohem Aufwand simulierbar ist, werden nur wenige (1-3) Schichten gemeinsam simuliert und die darunterliegenden Schichten in ihrem Verhalten durch eine variable Verzögerungszeit für die Paketübertragung repräsentiert. Durch diese Aggregation der unteren Schichten zu einem Ersatzmodell ist es möglich, die Einflüsse der unteren Schichten zu berücksichtigen, ohne sie gemeinsam mit den oberen Schichten simulieren zu müssen. Die variablen Verzögerungszeiten der aggregierten Schichten lassen sich durch analytische/simulative Verfahren oder durch Messungen ermitteln. Die Verzögerungszeiten sind im allgemeinen abhängig von dem Verkehrsangebot an diese Schicht und dem gesamten Angebot an das Übertragungsmedium. Dieses Angebot wird aus dem Angebot der Anwendung (User) an die oberste Schicht abgeschätzt. Damit werden die Verzögerungszeiten berechnet oder simulativ bestimmt. Bei der Simulation der oberen Schichten werden diese Angebotswerte an die darunterliegende Schicht gemessen und mit den angenommenen Verkehrswerten verglichen. Weichen die Werte zu sehr voneinander ab, so werden die Verzögerungszeiten neu abgeschätzt und die Simulation mit den neu bestimmten Werten wiederholt, bis eine hinreichend kleine Abweichung gegeben ist.

In der hier vorgestellten Untersuchung werden die Angebotswerte der Anwendung (User) an die Anwendungsschicht bestimmt, die Angebotswerte an das Transportsystem damit abgeschätzt und aus diesen Werten daraufhin die Angebotswerte an den Übertragungskanal abgeschätzt. Dies wird für alle Stationen gemacht, und damit die Simulation des GSM/CD-Übertragungskanal durchgeführt. Mit den erhaltenen Verzögerungszeiten über den Übertragungskanal wird das Transportsystem simuliert. Mit dessen Verzögerungszeiten wird wiederum die Simulation der oberen Schichten durchgeführt. Man nennt dieses Verfahren eine wiederholte Aggregation der Submodelle.

Durch die Komplexität und den Detaillierungsgrad der Modelle, die oft neuartige Modellkomponenten notwendig machen, ist die Anwendung von Simulationstools nicht oder nur eingeschränkt mit vielen Abstraktionen und Vereinfachungen möglich. Bei Simulationstools sind meist keine Erwei-

terungen der Modellkomponenten möglich und die resultierende Simulationszeit ist, bedingt durch die Unversalität und die komfortable Benutzerschnittstelle, oft nicht akzeptabel. Daher wurde für jedes Modell ein Simulationsprogramm mit Hilfe einer "Baustein-Bibliothek", wie sie am Institut für Nachrichtentechnik und Datenverarbeitung besteht, entwickelt. Die "Baustein-Bibliothek" ist ein Kompromiß zwischen einem Simulationswerkzeug und einem individuellen Simulationsprogramm. Sie enthält ein Muster der erweiterbaren dynamischen Datenstruktur und Prozeduren/Funktionen für immer wiederkehrende Aufgaben der Simulation wie Kalenderverwaltung, Zufallszahlengenerierung, Statistik, Warteschlangen-Einträge und -Ausgänge, u.s.w. Durch sorgfältiges Implementieren der dynamischen Datenstruktur und möglichst gute Lokalität der Daten im Hauptspeicher lassen sich Effizienz, auf die Probenstellung optimal angepaßte Simulationsprogramme mit vergleichsweise kurzer Programmaufzeit relativ schnell erstellen.

4. Ergebnisse

Den Ergebnissen liegt eine Konfiguration mit zwei Stationen zugrunde: einer aktiven Station, die die Aufträge generiert, und einer passiven Station, die die Aufträge empfängt und darauf reagiert. Die Untersuchung des Transportsystems, das identisch angenommen wird und auf einem LAN-Controller-Board mit eigenem Prozessor implementiert ist, ergibt konstante nur vom Pakettyp abhängige Übertragungszeiten, da die angebotene Last immer weit unter den maximal möglichen Durchsatz liegt. Es ist somit keine Iteration notwendig. Für die Simulation werden die Bedingdaumen der einzelnen Phasen konstant mit den Werten nach Tabelle 1 gewählt. Dieser Tabelle kann auch die Anzahl der Credits für die verschiedenen Flußkontrollen entnommen werden. Alle dargestellten Simulationsergebnisse sind für den Fall eines Kanals auf der Schicht 7 gewonnen worden. Die Ergebnisse für mehrere Kanäle zeigen gleiches Verhalten, wenn die Gesamtanzahl der Credits in den Flußkontrollen A identisch ist. Der Generator ist jeweils so eingestellt, daß die Ankunftsabstände der Aufträge negativ exponentiell verteilt sind (Markoffscher Ankunftsprozeß). Die Last wird jeweils bis zur Stabilitätsgrenze gesteigert. Die Prioritäten der Phasen in dem Modell nach Bild 2 nehmen in folgender Reihenfolge ab: $T_1, T_0, 3', 3, 2', 2, 1, V$. Auf jede Phase folgt eine Overheadphase, um die Anteile des Betriebssystems mit zu berücksichtigen (Prozeßwechsel). Die Phase $3'$ bildet eine Ausnahme, da auf sie direkt die Phase T_2 folgt. Timer sind so dimensioniert, daß kein Timeout auftritt. Bei beiden Protokollarchitekturen wird davon ausgegangen, daß die Protokolle der Schichten 5-7 auf einem Prozessor abgewickelt werden. Auf diesen Prozessor haben die Phasen der unteren Schichten gegenüber den Phasen der oberen Schichten Priorität. Innerhalb einer Schicht wird die Empfangsrichtung gegenüber der Senderichtung priorisiert. Die niedrigste Priorität hat also der MMS-Monitor. Die Prioritäten seiner Phasen nehmen in folgender Reihenfolge ab: $3'', 3', 3, 1, 2', 2, 1, V$.

In den Bildern 4 bis 9 sind Simulationsergebnisse zu den beiden Protokollarchitekturen dargestellt. Dabei gehören die Kurven der Bilder 4 bis 6 zu der AP-Protokollarchitektur und die Kurven der Bilder 7 bis 9 zu der MAP-Protokollarchitektur.

In den Bildern 4 und 5 ist jeweils die Speicherbelegungszeit eines Auftrags auf der Client-Seite und die Transferzeit eines Auftrags bis zum Server für den Fall *Auftrag ohne Quittung* beziehungsweise *Auftrag mit Quittung* über der Last für die AP-Protokollarchitektur aufgetragen. Die höhere Speicherbelegungszeit beim *Auftrag mit Quittung* kommt daher, daß hier auf die Quittung gewartet

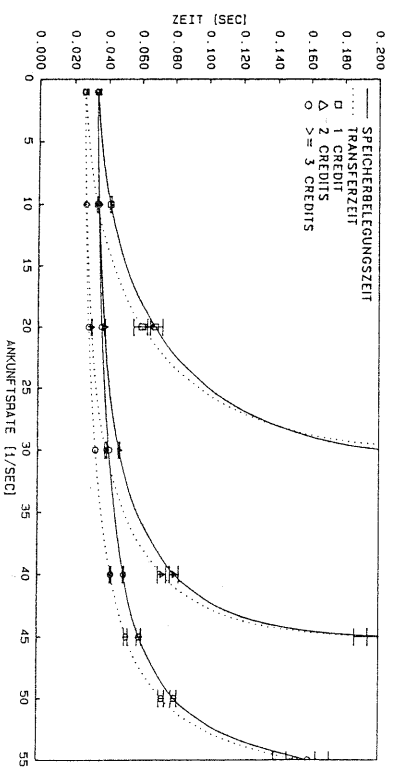


Bild 4 : Speicherbelegungszeit und Transferzeit für Auftrag ohne Quittung bei der AP-Protokollarchitektur

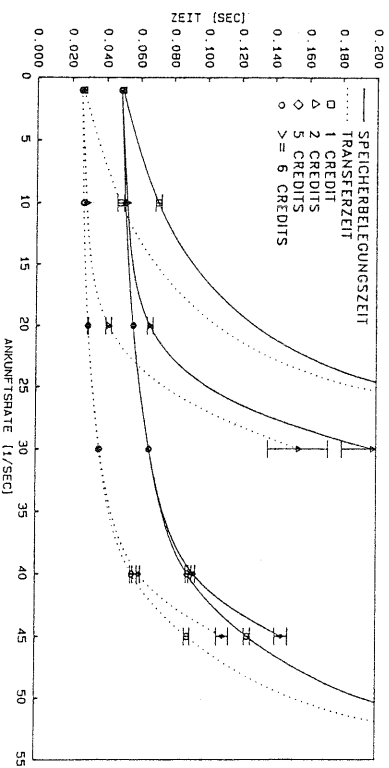


Bild 5 : Speicherbelegungszeit und Transferzeit für Auftrag mit Quittung bei der AP-Protokollarchitektur

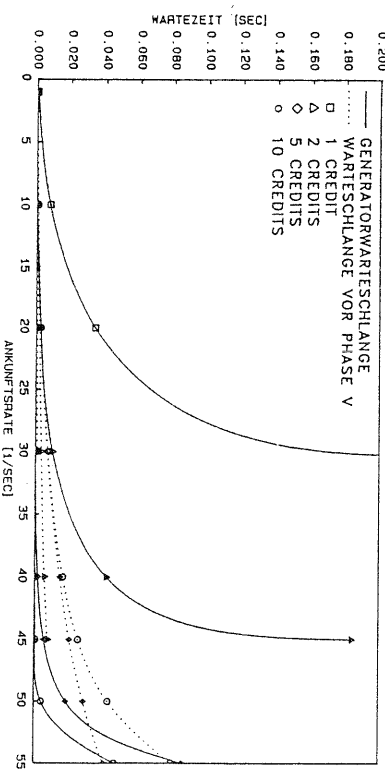


Bild 6 : Wartezeiten in der Generatorwarteschlange und vor der Phase V für Auftrag ohne Quittung bei der AP-Protokollarchitektur

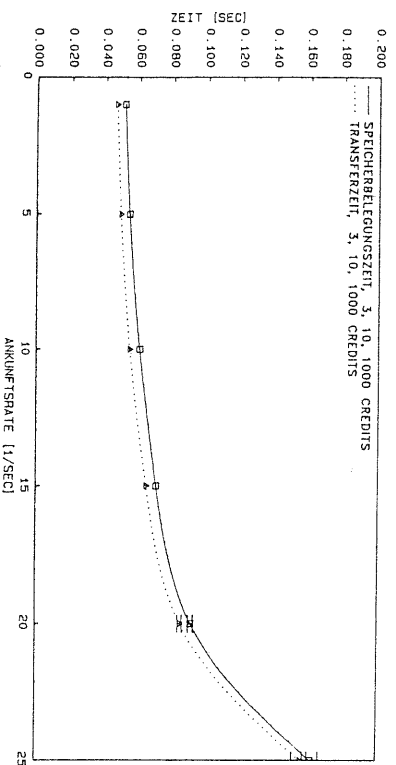


Bild 7 : Speicherbelegungszeit und Transferzeit für *Auftrag ohne Quittung* bei der MAP-Protokollarchitektur

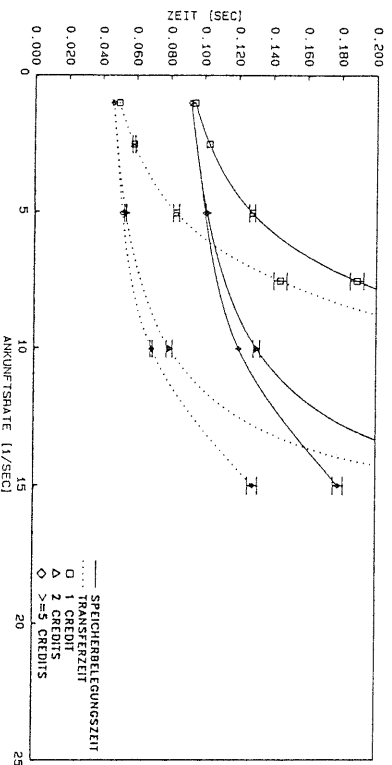


Bild 8 : Speicherbelegungszeit und Transferzeit für *Auftrag mit Quittung* bei der MAP-Protokollarchitektur

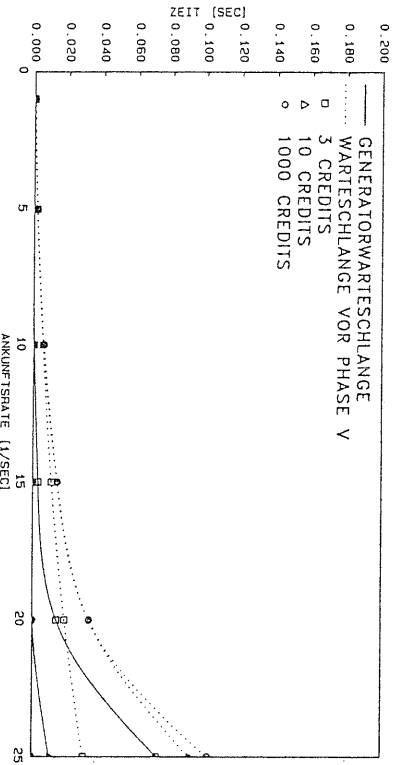


Bild 9 : Wartezeiten in der Generatorwarteschlange und vor der Phase V für *Auftrag ohne Quittung* bei der MAP-Protokollarchitektur

werden muß, während beim *Auftrag ohne Quittung* der Speicher bereits beim Empfang der Transportquittung freigegeben wird. Bild 6 ist zu entnehmen, daß bei einer harten Begrenzung der Credits in der kanalindividuellen Flußkontrolle A diese Flußkontrolle den Engpaß darstellt. Wird dieser Engpaß durch Erhöhung der Anzahl von Credits beseitigt, so verlagert sich der Engpaß zunehmend zu der Warteschlange vor der Prozessorphase mit der niedrigsten Priorität (V). In diesem Bild haben wir aus Übersichtlichkeitsgründen auf die Darstellung der (kleinen) Vertrauensintervalle verzichtet.

In den Bildern 7 und 8 ist jeweils wieder die Speicherbelegungszeit eines Auftrags auf der Requestor-Seite und die Transferzeit eines Auftrags bis zum Responder für den Fall *Auftrag ohne Quittung* beziehungsweise *Auftrag mit Quittung* über der Last für die MAP-Protokollarchitektur aufgetragen. Die höhere Speicherbelegungszeit beim *Auftrag mit Quittung* kommt wieder daher, daß hier auf die Quittung gewartet werden muß, während beim *Auftrag ohne Quittung* der Speicher bereits beim Empfang der Transportquittung freigegeben wird. Bild 9 ist zu entnehmen, daß sich hier der Engpaß in der Warteschlange vor der Prozessorphase mit der niedrigsten Priorität (V) ausbildet. Bei einer kleinen Anzahl von Credits in Flußkontrolle A und bei großer Last verschiebt sich der Engpaß zu dieser Flußkontrolle. In diesem Bild haben wir wieder aus Übersichtlichkeitsgründen auf die Darstellung der (kleinen) Vertrauensintervalle verzichtet. Interessant ist, daß sich trotz dieser Verlagerung des Engpasses an den Transferzeiten und an den Speicherbelegungszeiten nach Bild 7 nichts ändert. Wegen der wesentlich größeren Anzahl von Prozessorphasen wird die Stabilitätsgrenze bei der MAP-Protokollarchitektur wesentlich früher erreicht als bei der AP-Protokollarchitektur. Die Transferzeiten bei der MAP-Protokollarchitektur sind bei gleicher Last deutlich höher als bei der AP-Protokollarchitektur. Bei beiden Modellen kommt der größte Teil der Transferzeiten vom Durchlaufen der sendenden Station in abgehender Richtung. Alle anderen Anteile der Transferzeiten sind hier konstant über der Last und wesentlich kleiner.

5. Zusammenfassung

Ausgehend von Protokollarchitekturen wurden Modelle erstellt, die simulativ, durch wiederholte Aggregation der Submodelle, untersucht wurden. Die Untersuchung zeigt, daß die Leistungsfähigkeit der AP-Protokollarchitektur deutlich höher als bei der MAP-Protokollarchitektur ist. Dies liegt an der erhöhten Funktionalität der MAP-Protokollarchitektur und der dazu notwendigen Bearbeitungszeiten in den Schichten 5-7.

Die für eine Konfiguration exemplarisch dargestellten Kurven sind stellvertretend für die Kurven zu vielen weiteren denkbaren Konfigurationen zu sehen. Durch eine andere Wahl der Parameter, durch eine andere Reihenfolge der Phasen eines Prozessors oder durch eine andere Aufteilung der Schichten auf einen oder mehrere Prozessoren können mit Hilfe unserer Simulationsprogramme beliebig viele weitere Ergebnisse gewonnen werden.

Die Autoren möchten sich bei Hans-Hubert Hemberger und bei Waldemar Herscher, die die Simulationsprogramme implementiert und die Produktionsläufe durchgeführt haben, herzlich bedanken.

- [1] Feldmann, B. Kiesel, Meisner, "SINEC Communication Processor - SGP", Siemens AG, 1986
- [2] ISO 7498 : Information Processing Systems - Open Systems Interconnection - Basic Reference Model, November 1983
- [3] ISO 8072 : Transport Service Definition, May 1984
- [4] ISO 8073 : Transport Protocol Specification, May 1984
- [5] ISO 8326 : Basic Connection Oriented Session Service Definition, September 1984
- [6] ISO 8327 : Basic Connection Oriented Session Protocol Specification, September 1984
- [7] ISO DP 8571 : File Transfer, Access and Management, April 1985
- [8] ISO DIS 8649 : Service Definition for Common Application Service Elements - Part 2 : Association Control, April 1986
- [9] 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
- [11] ISO DIS 8802/3 : Local Area Networks - GSM/CD Access Method
- [12] ISO DIS 8802/4 : Local Area Networks - Token Passing Bus Access Method
- [13] ISO DP 8822 : Connection Oriented Presentation Service Definition, May 1986
- [14] ISO DP 8823 : Connection Oriented Presentation Protocol Specification, May 1986
- [15] ISO DIS 8824 : Specification of Abstract Syntax Notation One (ASN.1), 2nd DIS, September 1986
- [16] ISO DIS 8825 : Specification of Basic Encoding Rules for Abstract Syntax Notation One (ASN.1), 2nd DIS, September 1986
- [17] ISO DP 9506 : Manufacturing Message Specification, Part 1: Service Definition, EIA Project 1393A Draft 6, 2nd DP, May 1987
- [18] ISO DP 9506 : Manufacturing Message Specification, Part 2: Protocol Specification, EIA Project 1393A Draft 6, 2nd DP, May 1987
- [19] MAP (Manufacturing Automation Protocol) Version 3.0, General Motors, Warren/Michigan, April 7, 1987
- [20] SINEC Automation Protocol (AP 1.0), Teil 1: Protokollspezifikation, Siemens AG, 30. Juni 1987
- [21] SINEC Automation Protocol (AP 1.0), Teil 2: Technologische Funktionen, Siemens AG, 28. Mai 1986
- [22] E. Stahl, M. Veltes, "ESPRIT Project 955, CNMA, MMS Interface Specification", Version 4.1, Siemens AG, November 2, 1987

Integration von Netzleitinformationen als Basis eines CIM-Konzeptes für EVU-Netze

Hans-Jörg Haubner
Fraunhofer-Institut für Informations- und Datenverarbeitung
7500 Karlsruhe

Abstract

Integration of information has become an important factor in power utilities for control, planning and administration of electrical networks. In this paper an approach is described to fulfill these requirements. It is based on a central computer system linking the heterogeneous hosts in a wide area network. This system has two main functions. It provides a transparent transport system for data flow between the hosts and a central database for actual data of the electrical network supporting integrated data acquisition and access of the hosts. The realized system can be regarded as an approach in the sense of the concept of Computer Integrated Manufacturing (CIM).

Zusammenfassung

Der Austausch von Netzleitinformationen zwischen Netzleitstellen, Verwaltungs- und Energiewirtschaftsstellen eines EVU wird zu einem immer wichtigeren Instrument für unternehmensweite Überwachungs-, Planungs- und Abrechnungsaufgaben. Im vorliegenden Beitrag wird ein Lösungsansatz beschrieben, der die Integration von technischen und organisatorischen Funktionen von Einzelkomponenten in ein Gesamtsystem im Sinne des Begriffes CIM (Computer Integrated Manufacturing) unterstützt.

Grundidee des Lösungsansatzes ist es, die Integration über ein Koppelrechnersystem durchzuführen. Dieses System erfüllt dabei zwei Hauptaufgaben. Zum einen stellt es für die zu integrierenden Stellen eine geeignete Kommunikationsstruktur zur Verfügung, die als Transportsystem für den Datenaustausch dient. Zum zweiten beinhaltet es Datenzugriffsverfahren, die einen umfassenden und wahrfreien Austausch von Prozeß- und Planungsdaten innerhalb und zwischen den Rechnern der Prozeß- und Betriebsebenen mittels eines zentralen Datenhaltungssystems erlauben.