Die Abwicklung von Realzeit-Aufträgen in MAP-Netzen

Prof. Dr. Helmut Rzehak

Universität der Bundeswehr München
Institut für Systemorientierte Informatik
Werner-Heisenberg-Weg 39, D-8014 Neubiberg
Telefon: (089) 6004-3397 oder -2542

Zusammenfassung:

Das Manufacturing Automation Protocol (MAP) ist ein mit beträchtlichen Investitionen verbundener Vorstoß von General Motors, den Bereich der Datenkommunikation in der Fertigung zu standardisieren. Der Datenaustausch zwischen Geräten und Rechnern verschiedener Hersteller, die den MAP-Definitionen genügen, soll ohne besondere Anpassungen möglich sein. MAP baut weitgehend auf Standardisierungsvorschläge der ISO für die Kommunikation in offenen Rechnernetzen auf. Die Anwendbarkeit ist auch außerhalb der Automobilindustrie in weiten Bereichen gegeben. Zum allgemeinen Verständnis wird kurz auf die Prinzipien bei der Abwicklung dieser Protokolle eingegangen. Es werden die grundsätzlichen Probleme bei der Abwicklung von Realzeitaufträgen behandelt, und die Konzepte vorgestellt, die bei der z.Zt. durchgeführten überarbeitung des MAP-Standards berücksichtigt werden! Der aktuelle Stand dieser Arbeiten wird berichtet und eine Analyse gegeben, in welcher Größenordnung Zeitbedingungen eingehalten werden können.

1. Datenkommunikation in der Fertigungsautomatisierung

1.1 Die Ausgangssituation

Der generelle Trend zur Verwendung von immer mehr und immer komplexeren Rechensystemen und Steuerungseinrichtungen mit Mikroprozessoren in der Fertigungsautomatisierung ist durch einige Besonderheiten gekennzeichnet:

- Die verwendeten Rechensysteme bzw. Automatisierungskomponenten lassen sich in der Regel nicht auf eine Systemfamilie bzw. einen Hersteller beschränken. Die Vielfalt ist tendentiell um so größer, je größer die Fertigungseinrichtung ist.

- Die Aufgabenstellungen werden immer umfassender und zwingen zur Integration von Lösungen für einzelne Teilaufgaben (Schlagwort CIM).

Zusammengefaßt bedeutet dies, daß in der Fertigungsautomatisierung zunehmend heterogene Rechnernetze verwendet werden und daß der Anschluß von Automatisierungseinrichtungen an lokale Rechnernetze immer wichtiger wird. Dies wird jedoch durch eine Vielzahl nicht übereinstimmender Schnittstellen bei Hardware und Software erheblich erschwert und kann bei immer größeren Netzen einen beträchtlichen technischen und finanziellen Aufwand bedeuten, wenn es nicht gelingt diese Schnittstellen zu vereinheitlichen.

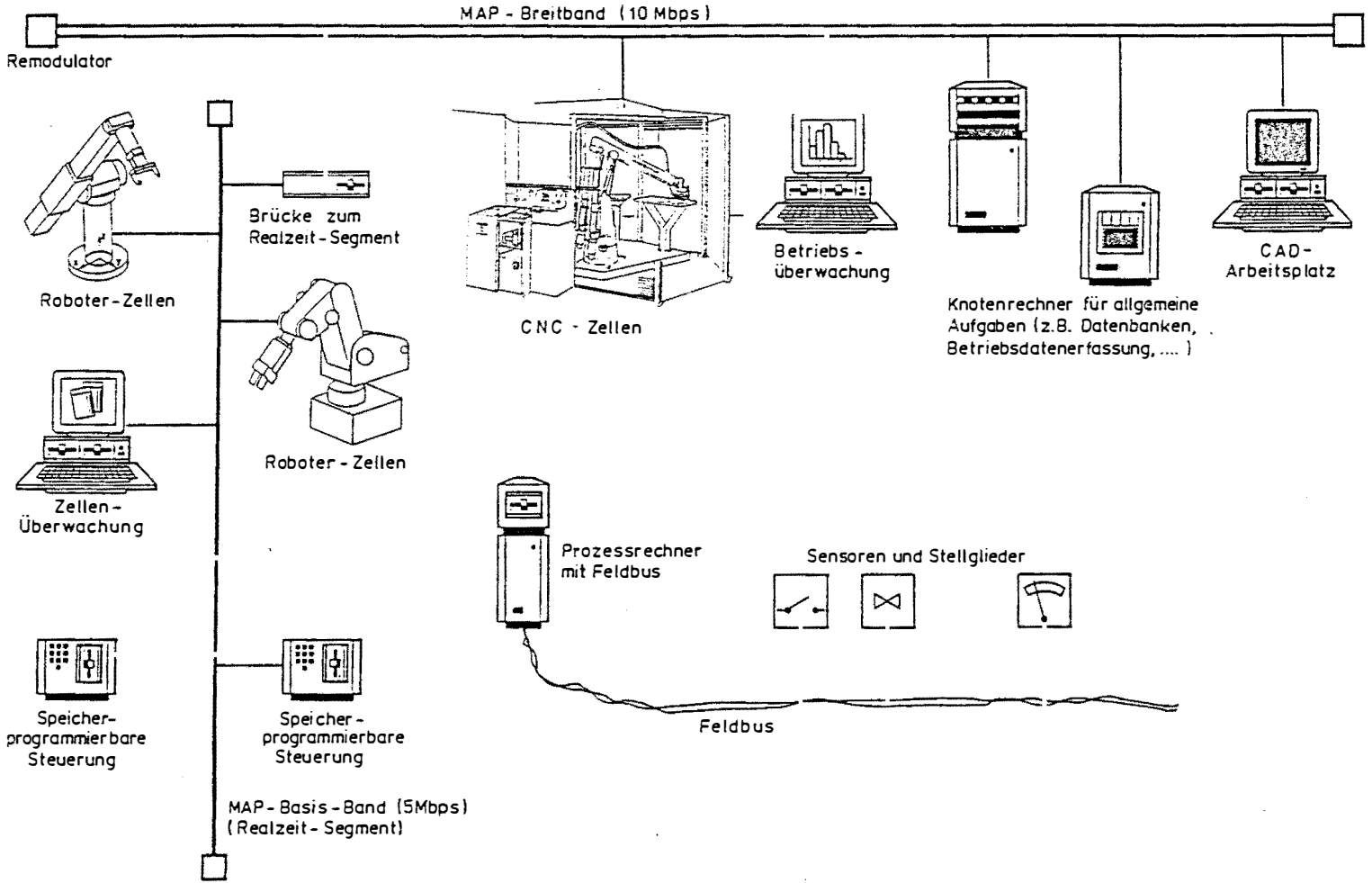
In dieser Situation wurde auf Initiative von Gerneral Motors mit beträchtlichem Aufwand versucht, durch Definition des Manufacturing Automation Protocol (MAP) einen Netzstandard für die Datenkommunikation zwischen Automatisierungseinrichtungen und Rechnern zu schaffen. MAP baut auf ISO-Normen auf, soweit diese zur Verfügung stehen. Es werden jedoch Verfahren und Parameter festgelegt, wenn auf Grund der Normen eine Auswahl oder Festlegung von Parametern oder Optio-

nen möglich oder auch erforderlich ist. Trotz großer Fortschritte der Open Systems Interconnection (OSI) Arbeitsgruppen in der ISO gibt es derzeit noch Lücken im Normungswerk, so daß für MAP Ergänzungen notwendig waren. Teilweise wurden diese Ergänzungen auch bereits in die ISO-Normen aufgenommen. Derzeit veröffentlicht ist die MAP-Version 2.1; eine neue Version 3.0 liegt als internes Arbeitspapier bereits vor und soll im Frühjahr 1988 veröffentlicht werden. In diesem Bericht sind die bekannt gewordenen wesentlichen Änderungen in MAP 3.0 bereits berücksichtigt.

1.2 Verwendungsbereich von MAP-Netzen

Eine wichtige Vorentscheidung für das zu definierende Netzprotokoll wird durch die Festlegung der verfügbaren "Knotenintelligenz" getroffen, d.h. welche Verarbeitungsleistung für die Protokoll-Abwicklung verfügbar sein soll. Die hierdurch verursachten Kosten könnnen sinnvoll nur einen Bruchteil der Kosten des ganzen Knoten darstellen. Die für MAP getroffenen Festlegungen gehen davon aus, daß einfache Sensoren als selbständige Knoten nicht in Betracht kommen. Andererseits sollen neben frei programmierbaren Prozessrechnern auch speicherprogrammierbare Steuerungen, numerisch gesteuerte Werkzeugmaschinen, Handhabungsgeräte u.ä., selbständige Knoten bilden können. Für diese Gruppe von Geräten ist es nicht ausreichend, daß MAP ein allgemeines Botschaftsformat festlegt und die Interpretation der Botschaft den Vereinbarungen zwischen den Anwenderprogrammen überläßt. Es bedarf vielmehr der Festlegung von Kommandos, die von den Geräten einheitlich interpretiert werden.

Der Trend zur Ingeration der gesamten Produktionssteuerung einschließlich Entwurf und Fertigungsvorbereitung bringt es mit sich, daß MAP-Netze aus dem Bereich der Fertigung herausführen und sich auch für die übertragung größerer Datenmengen unter weniger harten Zeitforderungen eignen müssen. Damit ergibt sich die in Bild 1.1 gezeigte typische Architektur eines MAP-Netzes.


MAP - Breitband (10 Mbps)
Remodulator
Roboter-Zellen
Brücke zum Realzeit-Segment
Roboter-Zellen
Zellen-Überwachung
CNC - Zellen
Betriebs-überwachung
Knotenrechner für allgemeine Aufgaben (z.B. Datenbanken, Betriebsdatenerfassung, ...)
CAD-Arbeitsplatz
Prozessrechner mit Feldbus
Sensoren und Stellglieder
Feldbus
Speicher-programmierbare Steuerung
Speicher-programmierbare Steuerung
MAP-Basis-Band (5Mbps) (Realzeit-Segment)

1.3 Erwartungen an die Standardisierung

Der Anwender erwartet von der Standardisierung vor allem die Interoperabilität der Systeme von verschiedenen Herstellern. Normenkonformität alleine genügt nicht, wenn nicht die Norm selbst alle wesentlichen Aspekte der Interoperabilität berücksichtgt. Die Netze müssen offene Systeme sein, d.h. es müssen während des Betriebes neue Knoten hinzufügbar und zusätzliche Programmkomponenten integrierbar sein. Die Protokolle sollen natürlich nur wenig übertragungs- und Verarbeitungsleistung absorbieren. Von großer Bedeutung ist auch die Möglichkeit vorhandene Netze schrittweise so umzugestalten, daß sie nach einigen Umrüstungen normkonform sind (Migrationswege).

2. Das ISO-Referenzmodell für die Kommunikation in Offenen Systemen

2.1 Grundlgegende Betrachtungen

Die ISO-Normen für Offene Systeme orientieren sich an einem Referenzmodell, das in der für lokale Netze gebräuchlichen Variante in Bild 2.1 dargestellt ist. Ein Kommunikationsauftrag aus einem Anwendungsprogramm (oberhalb Schicht 7) wandert durch die Schichten von oben nach unten, wobei die Schicht i-1 der nächst höheren Schicht i verschiedene Dienste zur Verfügung stellt. Der Aufruf dieser Dienste muß bestimmten Schnittstellenkonventionen genügen. Beim Zielknoten wandert der Auftrag nun durch alle Schichten von unten nach oben, bis er das Anwendungsprogramm erreicht hat, für das er bestimmt ist. Damit eine Schicht die ihr zugewiesene Aufgabe erfüllen kann, müssen bestimmte Vereinbarungen zwischen den beteiligten Knoten für diese Schicht getroffen werden (Protokoll).

Knoten A			Node B
7	Anwendung	Anwendungsbezogene Dienste	Application
6	Darstellung	Wandlung von Datenformaten	Presentation
5	Kommunikations-steuerung	Transformation von Namen in Adressen; Prüfen von Zugriffsrechten; Synchronisation	Session
4	Transport	Herstellen und Überwachen einer gesicherten End-zu-End-Verbindung	Transport
3	Netzwerk	Wegelenkung für Botschaften zwischen nicht benachbarten Knoten	Network
2	Verbindungskontrolle	Erkennen von Übertragungsfehlern	Logical Link Control
	Mediumzugriff	Abwickeln des Zugriffsverfahrens zum Übertragungsmedium	Medium Access Control
1	Phys. Signal	Aufprägen und Übertragen der Information entsprechend der gewählten physikalischen Signaldarstellung	Physical

Bild 2.1 Das ISO-Referenzmodell für lokale Netze

Kommunikation ist eine gerichtete Beziehung, d.h. es gibt einen Sender einer Nachricht und einen Empfänger. Dem entspricht das Wandern der Aufträge von oben nach unten (Sender) bzw. von unten nach oben (Empfänger) durch die einzelnen Schichten. Dabei werden jeweils Daten einer bestimmten Struktur übergeben, die nachfolgend als Protokoll-Dateneinheit (protocoll data unit, pdu) bezeichnet werden. Die Übergabe einer solchen Protokoll-Daten-Einheit von oben nach unten (Sender) wird als Anforderung (request) und die Übergabe von unten nach oben als Anzeige (indication) bezeichnet. Zur Abwicklung der Protokolle fügt jede Schicht bei einer Anforderung bestimmte Steuerinformation vorne an die Protokoll-Daten-Einheit an. Diese Steuerinformation wird von der entsprechenden Schicht beim Empfänger ausgewertet und entfernt, wenn sie die Protokoll-Daten-Einheit bei einer Anzeige erhält. Es entsteht die in Bild 2.2 gezeigte typische Anordnung von Protokoll-Daten-Einheiten. Man sieht, daß die zu übertragende Information durch die Protokolle der einzelnen Schichten beträchtlich anwachsen kann.

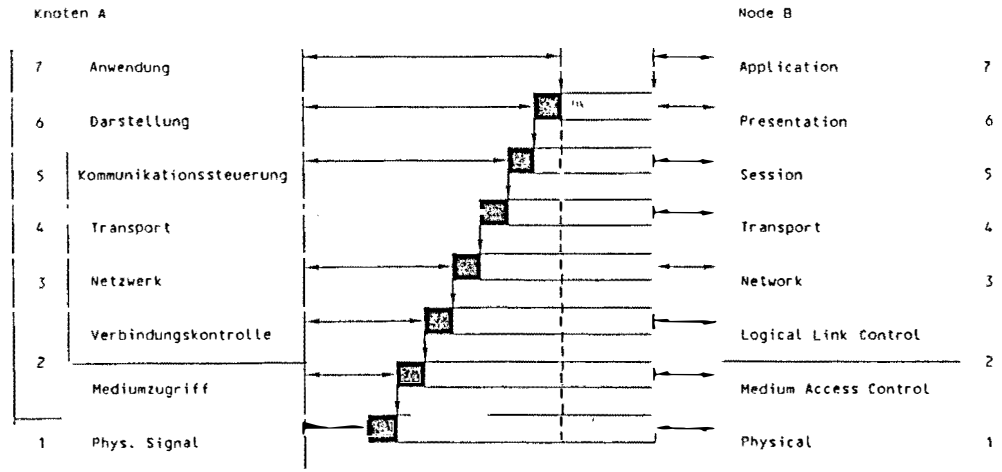


Bild 2.2 Zusammensetzung der Protokoll-Dateneinheiten

Verschiedene Anwendungs-Instanzen eines Knoten können gleichzeitig und unabhängig voneinander Kommunikationsbeziehungen unterhalten. Die Transport-Schicht stellt hierzu verschiedene Zugänge zur Verfügung. Zur Identifizierung einer Kommunikationsbeziehung wird daher neben den Knotenadressen der beteiligten Knoten noch die Nummer des zugehörigen Dienstzugangspunktes (service access point, sap) benötigt.

2.2 Verbindungslose und verbindungsorientierte Dienste

Aus historischen Gründen assoziieren die Postverwaltungen mit einer Kommunikationsbeziehung gerne eine "Verbindung" (Leitung, Kanal), die auf- und abgebaut werden muß. Dieses Denken hat auch lange Zeit die Normung für Datennetze stark geprägt, obwohl z.B. in einer Paketvermittlung kein technisches Äquivalent zu einer Verbindung zu finden ist. In den ersten Normen für offene Rechnernetze wurden daher verbindungsorientierte Dienste für alle Schichten des Referenzmodells vorgesehen. Dies hat den unbestreitbaren Vorteil, daß

nach dem erfolgten Verbindungsaufbau die Kommunikationspartner dem Vermittlungsnetz bekannt sind und die ausgetauschten Daten intern quittiert werden können. Man erreicht somit eine stabile, weitgehend gesicherte Kommunikation. In lokalen Netzen führt dies zu einem aufwendigen Protokoll für die Verbindungsaufnahme, obwohl die Nachrichten an alle angeschlossenen Stationen gelangen und dort auf Grund der mitgesendeten Adresse ausgewählt werden. Besonders störend wird dies, wenn während einer Kommunikationsbeziehung nur wenig Daten ausgetauscht werden.

Es wurden daher für die unteren Ebenen Protokolle für verbindungslose Dienste (connectionless mode) ausgearbeitet. Für eine gesicherte übertragung ist dann allerdings in der Transportschicht ein aufwendigeres Protokoll erforderlich. In MAP ist nun festgelegt, daß in den unteren Schichten bis einschließlich der Netzwerk-Schicht verbindungslose Dienste benutzt werden, während in der Transportschicht das verbindungsorientierte Protokoll für ungesicherte Transportwege (Klasse 4) verwendet wird. Dies ergbit den in Bild 2.3 dargestellten typischen Ablauf einer Verbindung zwischen zwei Instanzen der Schicht 4.

2.3 Zusammenstellung der in MAP verwendeten ISO-Normen

Anwendungs-Schicht

File Transfer, Access and Management (FTAM)
ISO DIS 8571

Manufacturing Message Service (MMS) ISO DP 9506 (entspricht RS 511); Verwendung in MAP 3.0

Darstellungs-Schicht

Die Darstellungs-Schicht ist in MAP 2.1 leer. Es ist zu erwarten, daß in MAP 3.0 eine Protokoll-Festlegung entsprechend der in ISO DIS 8822 "Connection Oriented Presentation Service Definition" oder einer Untermenge davon verwendet wird.

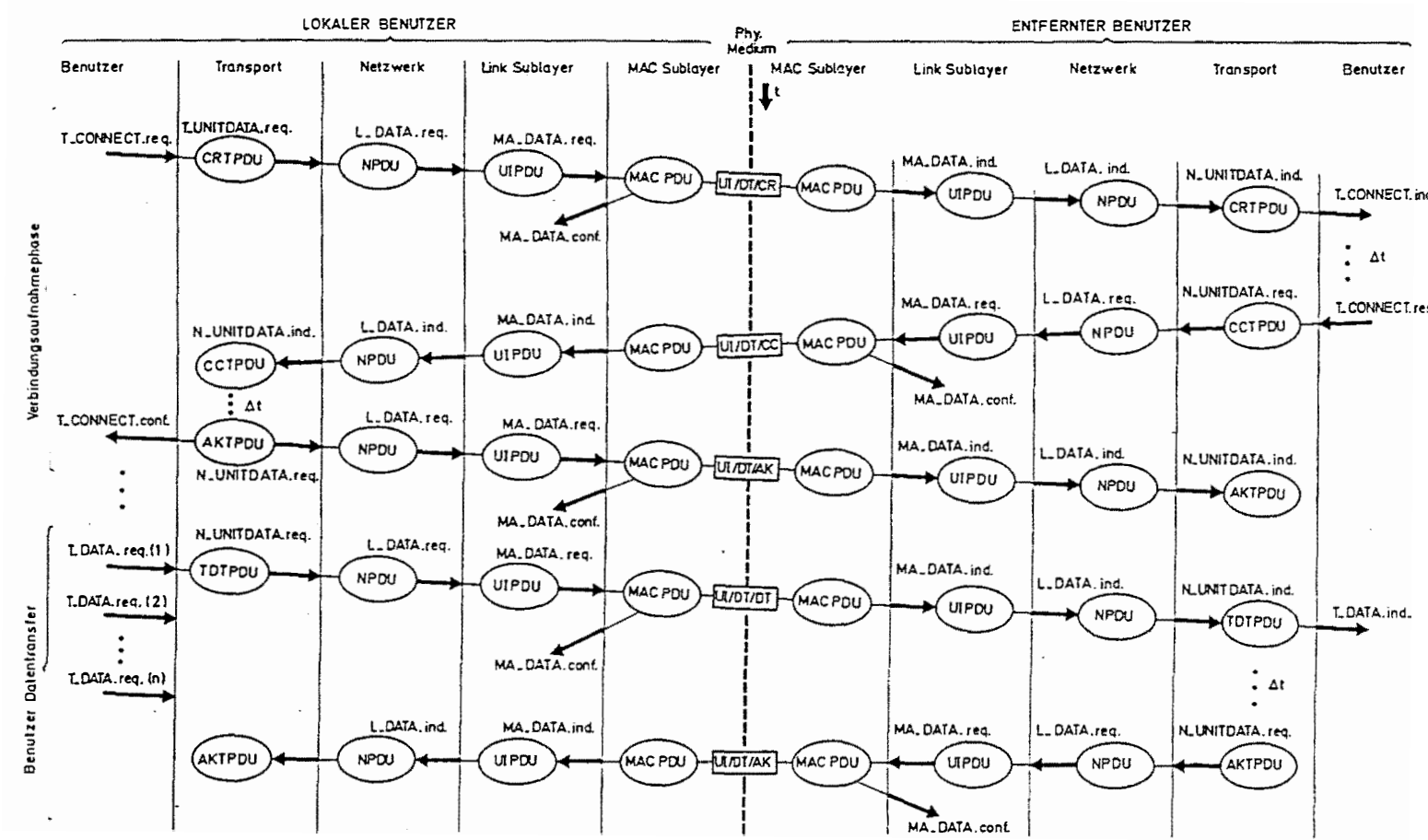


Bild 2.3 Verbindungsablauf zwischen zwei Instanzen der Transport-Schicht

Kommunikationssteuerungs-Schicht:

ISO-Session-Kernel entsprechend ISO IS 8326 "Connection-Mode Session Service Definition" (entsp. X.215)

Transport-Schicht:

Transport Service Definition ISO IS 8072 (entsp. X.214)

Verwendet wird das Klasse 4-Transport-Protokoll für ungesicherte Transportwege gemäß ISO IS 8075 "Connection oriented Transport Protocol Spezification"

Netzwerk-Schicht:

Network Service Definition ISO IS 8348 (entsp. X.213)

Verwendet wird das Connectionless Mode Protocol für Kommunikation innerhalb eines LAN mit Benutzung des Inactiv Network Layer Protocol, falls keine Auftrennung der Dateneinheit erforderlich ist und der Zielknoten mit dem dabei vereinfachten Kontrollblock adressierbar ist.

Verbindungskontroll-Schicht:

LAN-Logical Link Control ISO DIS 8802/2 (entspricht IEEE 802.2) mit LLC-Typ 1 und LLC-Typ 3 (in MAP 3.0)

Mediumzugriffs-Schicht:

LAN-Token-passing Bus Access Method ISO DIS 8802/4 (entspricht IEEE 802.4)

Phys. Signal-Schicht:

Breitband-Netz mit 10 Mbps gemäß ISO DIS 8802/4 (2-Kanal Option)

Basisband-Netz mit 5 Mbps gemäß ISO DIS 8802/4

Es kann grundsätzlich jedes Medium verwendet werden, wenn die in ISO DIS 8802/4 beschriebene Schnittstelle zur Übergabe der MAC-Symbole eingehalten wird. Normen zur Verwendung von Lichtleitern sind in Vorbereitung.

3. Zeitkritische Aufträge in lokalen Netzen

3.1 Grundsatzforderungen

Unter Realzeitbetrieb eines Rechensystems versteht man im allgemeinen eine Betriebsweise, bei der die Aufträge im Rechensystem schritthaltend mit externen Vorgängen abgearbeitet werden, d.h. daß die Verarbeitung simultan zur Datenentstehung durchgeführt wird. Dies bedeutet, daß die Verarbeitungsergebnisse innerhalb einer vorgegebenen Zeitspanne verfügbar sein müssen (garantierte Antwortzeiten), und daß mehrere Aufträge gleichzeitig zu bearbeiten sind (Mehrprogrammbetrieb). Die Größenordnung der Antwortzeiten ist problemabhängig. Da die einzelnen Aufträge im allgemeinen nicht völlig unabhängig voneinander bearbeitet werden können, ist eine gewisse Kooperation zwischen den beteiligten Rechenprozessen erforderlich.

Die Forderung nach garantierten Antwortzeiten impliziert, daß ein Auftrag nicht beliebig lange von der Bearbeitung ausgeschlossen werden darf. Abhängig von der Art der Kooperation kann es darüber hinaus erforderlich sein, daß Aufträge relativ zueinander einen bestimmten Bearbeitungsfortschritt machen. Es muß also ausgeschlossen werden, daß sich Rechenprozesse gegenseitig blockieren (deadlock) oder aber, daß einem einzelnen Rechenprozess Betriebsmittel auf Dauer vorenthalten werden, weil sie von anderen arbeitsfähigen Rechenprozessen nicht freigegeben werden (starvation). Da in Rechnernetzen kooperierende Prozesse auf verschiedenen Knoten ablaufen können, muß die Kommunikation so gestaltet werden, daß die oben gestellten Forderungen erfüllbar sind. Dabei stellt sich die Frage, welches die kleinstmögliche Antwortzeit ist. Da hier nur die Kommunikationsprobleme behandelt werden, sollen nur die Zeitverhältnisse bei der Kommunikation betrachtet werden. Bei diesen Betrachtungen spielt

die Zugriffsmethode auf das Kommunikationsmedium eine erhebliche Rolle.

3.2 Deterministischer oder nichtdeterministischer Zugriff zum Übertragungsmedium

Für die Regelung des Zugriffs auf das Übertragungsmedium stehen zwei Verfahren zur Diskussion. Bei dem nichtdeterministischen Verfahren "Carrier Sense Multiple Access with Collision Detection" (CSMA/CD) stellt ein sendewilliger Knoten fest, ob das Übertragungsmedium belegt ist. Ist dies nicht der Fall, so beginnt er zu senden. Insbesondere auch wegen der Kabellaufzeiten kann es vorkommen, daß mehrere Knoten zu senden beginnen. Dies führt zu Überlagerungen (Collision) und damit zu einer gegenseitigen Störung. Da die sendenden Knoten ihre eigene Sendung mitempfangen, wird diese Situation erkannt und allen Knoten übermittelt. Die sendenden Knoten müssen nach einer individuellen zufälligen Wartezeit ihre Sendung wiederholen, damit es hierbei möglichst nicht wieder zu Kollisionen kommt.

Bei dem deterministischen Verfahren mit Weitergabe der Sendeberechtigung von Knoten zu Knoten (Token-passing) wird die Sendeberechtigung in Form eines besonderen Zeichens (Token) weitergegeben. Nur wer im Besitz des Zeichens ist kann senden. Kollisionen sind ausgeschlossen. Ein Knoten darf erst wieder senden, wenn das Token einmal bei allen Knoten umgelaufen ist. Das Verfahren garantiert, daß innerhalb der Token-Umlaufzeit jede Station einmal senden konnte.

CSMA/CD ist nicht sehr komplex; allerdings treten bei zunehmendem Verkehr immer häufiger Kollisionen - auch bei der Wiederholung einer Sendung - auf, so daß eine bestimmte Maximalzeit für den Zugriff nicht garantiert werden kann. Das Token-Weitergabe-Verfahren ist komplex, da besondere Vorkehrungen für das Einschleusen neuer Knoten und gegen Verlust oder Duplizierung des Tokens bei übertragungsfehlern getroffen werden müssen.

Beide Verfahren sind in einer Reihe von Arbeiten ausführlich untersucht und verglichen worden. Eine ausführliche Darstellung findet sich z.B. in Schwartz, Telekommunication Networks. In Bild 3.1 ist das typische Verhalten der Verfahren bei unterschiedlichem Verkehr im Netz gegenübergestellt. Es ist darin die mittlere normierte Übertragungszeit - d.h. die Summe aus Zugriffs- und Transferzeit, bezogen auf die Zeit zum Senden des Paketes - in Abhängigkeit vom normierten Durchsatz - d.h. des Durchsatzes der Nutzdaten, bezogen auf die Übertragungsrate - dargestellt. Man erkennt, daß bei CSMA/CD die Mittelwerte der Zugriffszeiten immer größer werden, je größer der normierte Durchsatz ist. Die rechnerisch mögliche Übertragungskapazität (entspricht dem normierten Durchsatz 1) kann nicht erreicht werden, da wegen der immer häufiger werdenden Kollisionen ab einem bestimmten Auslastungsgrad die Nutzdatenrate sinkt. Die Zugriffszeiten können dann beliebig groß werden. Wegen dieses typischen Verhaltens wird in MAP das Token-Weitergabe-Verfahren vorgeschrieben. Die Gründe hierfür sind:

- Bei großer Auslastung des Netzes ist das Token-Weitergabe-Verfahren effizienter.
- Bei Fehlern im Fertigungsbetrieb und anderen Ausnahmesituationen wird ein höheres Kommunikationsbedürfnis erwartet. Gerade dann dürfen die Zugriffszeiten nicht steigen.
- Wenn das Netz besonderen Belastungssituationen gewachsen ist, und hierbei die Realzeitforderungen erfüllt sind, so gilt dies bei weniger kritischen Belastungssituationen erst recht. Effizienzbetrachtungen spielen dann höchstens eine untergeordnete Rolle.

Wie noch gezeigt wird, lassen sich auch beim Token-Weitergabe-Verfahren garantierte Zugriffszeiten nur unter besonderen Bedingungen angeben. Dieses Verfahren ist jedoch bei großer Netzbelastung grundsätzlich günstiger.

Parameter:

10 Mbps übertragungszeit
2 km Segmentlänge
50 Stationen
1 Bit Verzögerung pro Station
24 Bit Steuerinformation
1000 Bit mittlere Datenfeldlänge
bei exponentieller Verteilung

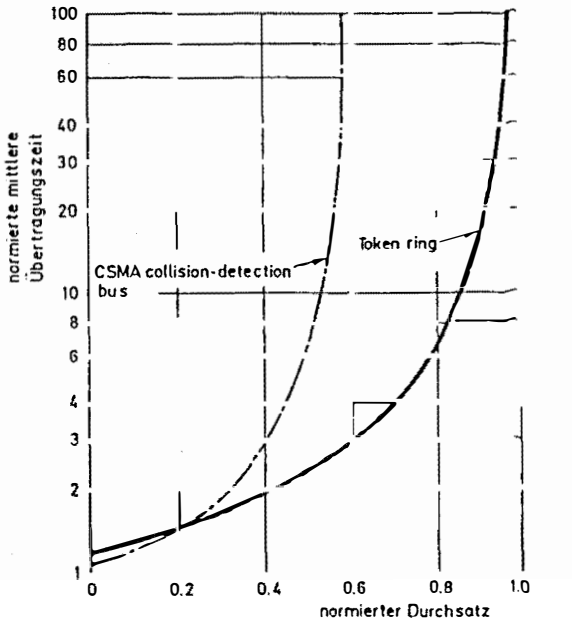


Bild 3.1 Lastverhalten der Zugriffsverfahren

3.3 Charakteristische Zeiten im übertragungsablauf

Ein typischer Kommunikationsablauf besteht darin, daß ein Knoten i eine Botschaft an einen Knoten k sendet und hierauf eine Antwort erwartet, z.B. Daten, die nur der Zielknoten k liefern kann. Hierbei ergibt sich der in Bild 3.2 dargestellte Kommunikationsablauf. Das Anwendungsprogramm richtet eine Anforderung an die Schicht 7, die entsprechend den Protokollen bearbeitet wird und nach der Zeit T_p an die Schicht 2 übergeben wird. T_p hängt von der Komplexität der Protokolle und der gewählten Implementierung ab. Nach Bearbeitung durch die LLC-Schicht (Zeit T_{LLC}) wird die Botschaft von der MAC-Schicht in einen Sendepuffer eingetragen. Wird nun der Knoten i sendeberechtigt, so überträgt er die im Sendepuffer abgelegten Botschaften (in der Reihenfolge ihres Eintreffens). Die Zeit, die dem Knoten zum Senden zur Verfügung steht, wird durch die Token Hold Time T_{HT} bestimmt. Damit

ist die Zugriffszeit T_Z, die von der MAC-Schicht zur Übergabe der Botschaft an das Medium benötigt wird, situationsabhängig. Für den Fall, daß der Sendepuffer bei jedem Token-Durchlauf vollständig geleert wird, ist der Maximalwert für T_V durch die Zeit für einen Token-Umlauf bestimmt (Token Rotation Time T_R). Zwischen T_R und $T_{HT,i}$ der einzelnen Stationen besteht ein Zusammenhang:

$$T_R = T_{Rmin} + \sum_{i=1}^{N} T_{HT,i} \qquad (3.1)$$

Dabei ist T_{Rmin} der Minimalwert für T_R, der sich bei einem Token-Umlauf ohne den Austausch von Nutzdaten ergibt. N ist die Anzahl der Stationen im Netz. Kann der Sendepuffer bei einem Token-Durchlauf nicht geleert werden, so lassen sich auch keine garantierten Zugriffszeiten angeben.

Die Zeit T_{PHY} für die Übertragung auf dem Medium ist durch die Übertragungsrate und die Länge der Botschaft bestimmt. Beim Knoten k (Empfänger) wird die Botschaft von der MAC-Schicht in einen Empfangspuffer eingetragen. Die Verweilzeit T^*_V in der MAC-Schicht des Empfangsknoten hängt von der Belastung dieses Knoten ab. Die Bearbeitungszeiten in den verschiedenen Protokollschichten sind mit T^*_{LLC} (LLC-Schicht) und T^*_p (Schicht 3 bis 7) bezeichnet. Nach Bearbeitung durch ein Anwendungsprogramm (Zeit T_B) steht das Ergebnis zur Rückübertragung bereit. Im 2. Halbzyklus wiederholt sich der beschriebene Ablauf, nur daß jetzt der Knoten k der Sendeknoten und Knoten i der Empfangsknoten ist. Es bleibt festzuhalten, daß der garantierte Zeitbedarf für beide Halbzyklen zusammen nicht kleiner als die zweifache Token-Umlaufzeit gehalten werden kann, und daß dieser Wert ohne besondere Maßnahmen auch nicht garantiert werden kann.

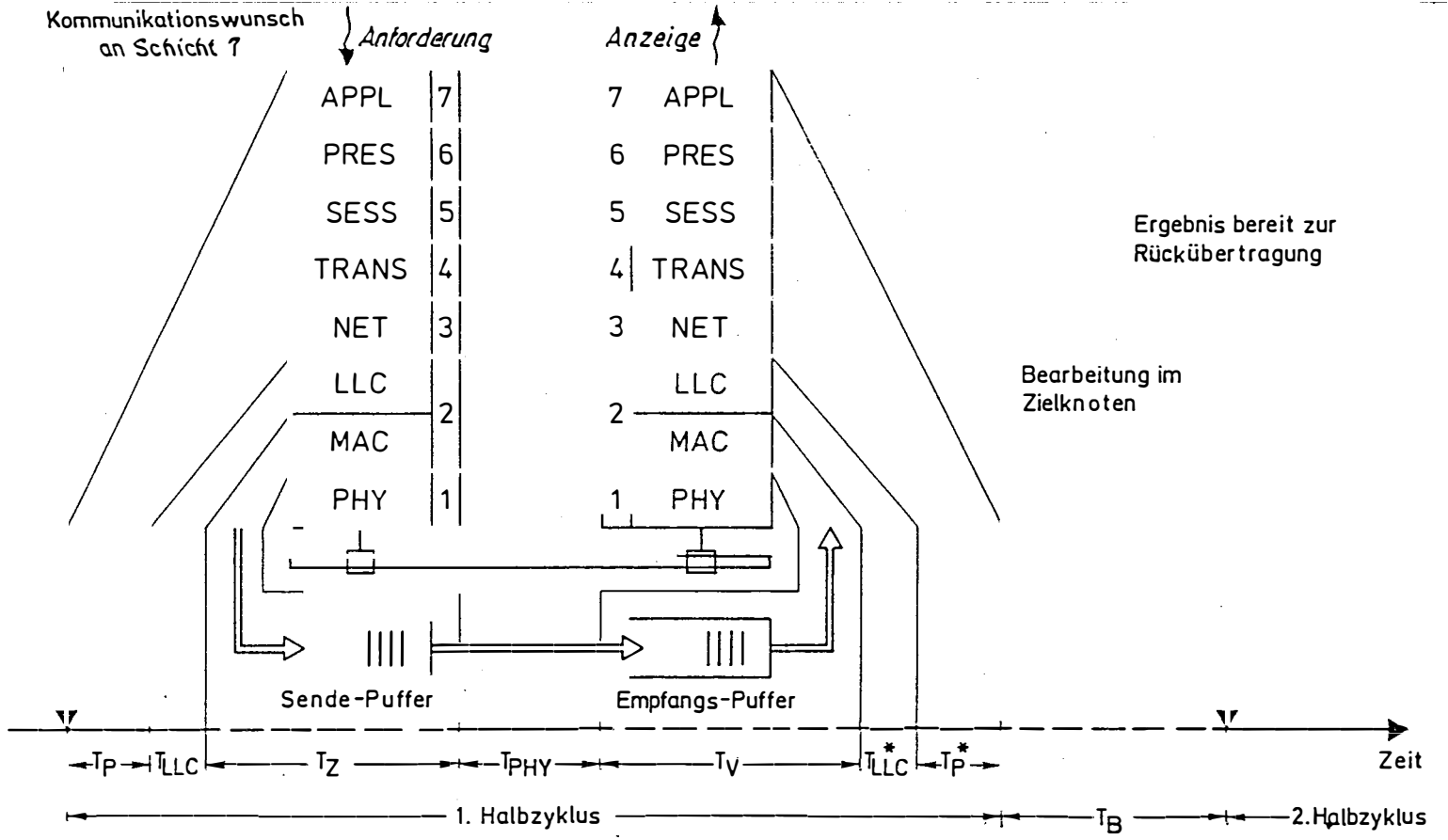


Bild 3.2 Typischer Übertragungsablauf

4. Realzeiteigenschaften der MAP-Protokolle

4.1 Realistische Erwartungen

In Abschnitt 3.3 sind bereits grundsätzliche Betrachtungen angestellt worden. Zusammenfassend gilt:

- Die garantierte Zugriffszeit T_{Vmax} kann nicht kleiner als die Token-Umlaufzeit T_R sein.

- Voraussetzung für eine garantierte Zugriffszeit ist, daß die Sendepuffer vollständig geleert werden, wenn ein Knoten sendeberechtigt wird.

- Für einen Kommunikationszyklus mit Rückübertragung des Ergebnisses können auch unter sonst optimalen Verhältnissen bis zu zwei Token-Umläufe vergehen.

Die Frage stellt sich also, welche obere Grenze der Token-Umlaufzeit für den Verwendungsbereich erforderlich und bei der verfügbaren Technologie auch erreichbar ist. Geht man einmal davon aus, daß in einem MAP-Segment für Realzeitaufgaben nicht mehr als 100 Knoten angeschlossen sind ($N \leq 100$) und verteilen sich die Kommunkationsbedürfnisse gleichmäßig, so kann man bei T_R = 100 ms mit einer Token Hold Time $T_{TH} \approx 1$ ms rechnen. Dies scheint technologische Randbedingungen nicht zu sprengen. In den PROWAY-Anforderungen erscheint jedoch die magische Zahl von 20 ms für die garantierte Zugriffszeit T_{Vmax}. Dies bedeutet, daß $T_R \leq 20$ ms sein muß. Für unser Beispiel ($N \leq 100$) ergibt dies eine Token Hold Time $T_{HT} \approx 0{,}2$ ms und es ist fraglich, ob das Leeren der Sendepuffer in dieser Zeit garantiert werden kann. Man kann also feststellen, daß bei der heute verfügbaren Technologie Realzeiteigenschaften, die als notwendig erachtet werden, mit dem für MAP zunächst vorgesehenen Verfahren nicht bzw. nur bei kleinen Netzen erreicht werden können. Es wurden da-

her Modifikationen ausgearbeitet, die zu einer erheblichen Verbesserung der Realzeiteigenschaften führen. Diese Modifikationen werden in MAP 3.0 als Optionen enthalten sein.

Die in MAP 3.0 vorgesehenen Verbesserungen der Realzeit-Eigenschaften haben zwei Zielrichtungen. Zum einen wird durch Modifikationen im Zugriffsverfahren erreicht, daß für gewisse Klassen von Botschaften die zeitliche Abwicklung verbessert wird. Zum anderen wird ein direkter Zugang aus dem Anwendungsprogramm zur Schicht 2 geschaffen. Hierdurch können die Verzögerungen durch die Protokollabwicklung erheblich reduziert werden. Diese Maßnahmen haben allerdings nicht nur positive Aspekte, wie im nachfolgenden dargestellt wird.

4.2 Modifikationen im Zugriffsverfahren

Die erste Modifikation betrifft das Einführen von Prioritäten beim Zugriff auf das Übertragungsmedium. Es wird dann nur für bevorzugte Botschaften im Sendepuffer garantiert, daß sie bei einem Token-Umlauf tatsächlich gesendet werden. Hierbei verwaltet die MAC-Schicht getrennte Sendepuffer für 4 Prioritäten, die in der Norm als Zugriffsklasse 6, 4, 2 und 0 bezeichnet werden. Es ist etwa an die folgende Verwendung dieser Prioritäten gedacht:

Priorität 6 : dringende Botschaften; nur für diese Priorität wird eine maximale Zugriffszeit garantiert.

Priorität 4 : normale Botschaften zur Prozesssteuerung

Priorität 2 : Betriebsdatenerfassung und Anzeige

Priorität 0 : Datei- und Programmtransfer

Die Sendepuffer für jede Priorität bilden eine selbständige Warteschlange und die Sendeberechtigung (Token) wird knoteninternen von der Priorität 6 zur nächst niedrigeren weitergereicht. Dabei arbeitet die Prioritätensteuerung etwa folgendermaßen:

Die Token Hold Time wird zunächst auf den Wert T_{HTE} (high priority token hold time) gesetzt. Es wird unterstellt, daß diese Zeit für eine Anwendung so gewählt ist, daß alle im Sendepuffer der Priorität 6 befindlichen Botschaften gesendet werden können. Für jede Priorität wird eine gewünschte Token-Umlaufzeit (T_{RT4}, T_{RT2}, T_{RT0}) definiert. Für die Prioritäten 4, 2 und 0 wird jeweils die Zeit festgehalten, die seit der letzten Weitergabe des Tokens vergangen ist (T_{RC4}, T_{RC2}, T_{RC0}). Steht nun eine Priorität $i < 6$ zur Bedienung an und ist für diese die gewünschte Token-Umlaufzeit größer als die Zeit seit der letzten Token-Weitergabe, so wird die Token-Hold-Time gleich der Differenz gesetzt:

$$T_{HT} = T_{RTi} - T_{RCi} \qquad (3.2)$$

Andernfalls wird das Token sofort weitergegeben. Damit man für die verschiedenen Prioritäten tatsächlich verschiedene Bedienqualitäten erhält, sollte folgende Regel gelten:

$$T_{RT4} \geq T_{RT2} \geq T_{RT0} \qquad (3.3)$$

Man könnte nun zu dem Schluß gelangen, daß die garantierte Zugriffszeit für die Priorität 6 durch

$$T_{V6,max} = \max (N*T_{HT6}, T_{RTi}) \qquad (3.4)$$

gegeben ist. Dies ist jedoch nicht ganz korrekt, weil die Zeitverhältnisse nur beim Beginn der Übertragung einer Botschaft geprüft werden. So können also z.B. lange Botschaften zu Abweichungen von (3.4) führen.

Eine zweite Modifikation im Zugriffsverfahren trägt dem Umstand Rechnung, daß ein Knoten erst dann eine Antwort auf eine empfangene Botschaft senden kann, wenn er sendeberechtigt wird. Will man z.B. nur sicher sein, daß eine Botschaft angekommen ist, so ist die hiermit verbundene Wartezeit lästig. Es wird daher die Möglichkeit einer unmittelbaren Antwort (immediate response, LLC-Type 3) eingeführt. Hierbei antwortet ein Knoten auf eine eingehende Botschaft mit Aufforderung zur Antwort sofort, während der ursprünglich sendende Knoten seine Sendeberechtigung behält. Der Zeitbedarf für die Antwort muß also bei der Token Hold Time berücksichtigt werden und kann zu einer Verlängerung der Token-Umlaufzeit führen.

Die zurückzusendende Antwort muß in der LLC-Schicht des angesprochenen Knotens bereits vorrätig sein. Das Verfahren eignet sich also zur Spiegelung (Echo) der gerade eingegangenen Botschaft als Empfangsquittung oder aber zur Abfrage hinterlegter Botschaften und könnte auch Commit-Verfahren für verteilte Transaktionen beschleunigen.

4.3 Der direkte Zugang zur Schicht 2

Aus der Sicht des Anwendungsprogrammes besteht der Zeitbedarf für den Botschaftsaustausch nicht nur aus der Zeit für den Zugriff auf das übertragungsmedium in der MAC-Schicht, sondern auch aus der Durchlaufzeit durch die anderen Protokollschichten (vgl. Bild 2.3). Diese Zeiten sind keineswegs vernachlässigbar klein. Es ist daher vorgeschlagen worden, daß Anwendungsprogramme für besonders zeitkritische Aufgaben einen direkten Zugang zur Schicht 2 erhalten. Dies setzt allerdings voraus, daß die beteiligten Knoten am selben Netzsegment liegen, da wegen der fehlenden Netzwerk-Schicht Knoten über die Segmentgrenze hinaus nicht adressierbar sind. Es werden nicht nur die Druchlaufzeiten durch die Schichten 3 bis 7 gespart, sondern es ergibt sich auch

eine Reduktion der zu übertragenden Daten, da keine Daten zur Steuerung der Protokollabwicklung übertragen werden müssen.

Der direkte Zugang zur Schicht 2 bringt zwei Problembereiche mit sich. Zum einen wird man wegen der fehlenden Schichten 3 bis 7 dem Anwendungsprogramm nicht die gleichen Funktionen anbieten können wie bei einem vollen Protokoll. Der Anwender wird also unterscheiden müssen, ob er auf einer 2-Schichten-Architektur oder einer 7-Schichten-Architektur arbeitet. Der andere Problembereich entsteht durch die wünschenswerte Koexistenz von beiden Architektur-Modellen auf dem gleichen Netzwerk-Segment.

In MAP 3.0 sind speziell für Realzeitaufgaben Basis-Band-Segmente vorgesehen. Ein Mini-MAP-Knoten (vgl. Bild 4.1) enthält nur die Schichten 1 und 2 sowie eine eingeschränkte Anwendungs-Schicht (Schicht 7). Daneben gibt es Knoten mit erweiterter Funktionalität (Enhanced Performance Architecture, EPA). Diese haben mehrere Zugriffspunkte zur LLC-Schicht. Genau einer davon wird für die volle Architektur (Schicht 3 bis 7) zur Verfügung gestellt. Die anderen dienen zum direkten Zugriff der Anwendungsprogramme über eine eingeschränkte Schicht 7. Ein Mini-MAP-Knoten kann nur mit einem Mini-MAP-Knoten oder einem EPA-Knoten des selben Segmentes Botschaften austauschen. Ein EPA-Knoten dagegen kann dank seiner vollen Architektur auch mit anderen Netz-Segmenten kommunizieren.

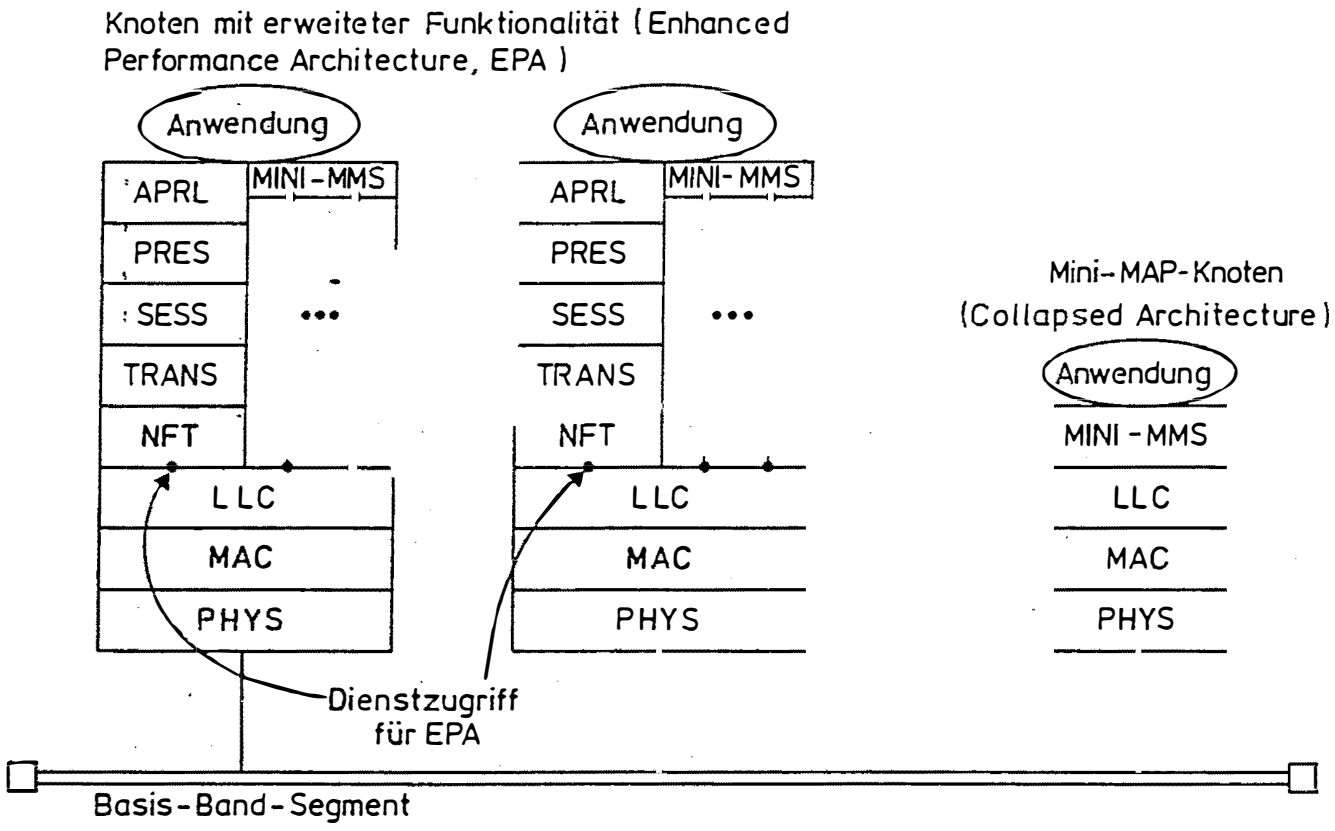


Bild 4.1 Mini-MAP und Knoten mit erweiteter Funktionalität (EPA-Knoten)

5. Die Verwendung von MAP-Komponenten

5.1 Verfügbarkeit und Anwendungsbereiche

MAP-Netze auf der Basis der seitherigen Verison 2.1 sind bisher nur als Studien- bzw. Pilotprojekte realisiert worden. Die Anwender warten im wesentlichen auf die Version 3.0, die bis zum Enterprise Network Event im Juni 1988 in Baltimore verfügbar sein wird. MAP bietet dem Anwender eine Reihe von anwendungsorientierten Funktionen, Dateizugriffsdienste und Netzwerkmanagement-Dienst an, auf die hier wegen des Schwerpunktes auf Realzeiteigenschaften nicht näher eingegangen werden kann. MAP 3.0 wird die Bedürfnisse zur Kommunikation im Fertigungsbereich entsprechend dem heutigen Stand der Technologie abdecken. Seine Druchsetzung am Markt wird neben den Kosten wesentlich von der tatsächlich erreichten Interoperabilität der Komponenten verschiedener Hersteller abhängen. Um dies zu gewährleisten werden neutrale Testzentren aufgebaut, die MAP-Komponenten auf ihre Normenkonformität prüfen. Die zur Demonstration beim Enterprise Network Event verwendeten Komponenten werden bereits gemäß MAP 3.0 getestet sein.

Die für den Anwender hauptsächlich wichtige Interoperabilität hängt jedoch nicht nur von der Güte der Konformitätsprüfungen ab sondern auch davon, ob in der Norm selbst alle wesentlichen Punkte für die Interoperabilität ausreichend geregelt sind. Man kann davon ausgehen, daß der gefundene Konsens unter den Fachleuten in den Normungsgremien und die gesammelten Erfahrungen mit früheren Versionen die Gewähr dafür gibt, daß alle essentiellen Punkte in MAP 3.0 berücksichtigt sind.

5.2 MAP und PEARL

Der im Mehrrechner-PEARL beschriebene Botschaften-Mechanismus ist in einem MAP-Netz (auch Mini-MAP) direkt implementierbar. Für das Blocking-Send-Protokoll kann z.B. der LLC-Typ 3 (vgl. Abschnitt 4.2) mit Vorteil verwendet werden. Auch läßt sich eine 1-zu-n-Kommunikation durch geeignete Gruppen-Adressierung realisieren.

MAP stellt jedoch eine Reihe von anwendungsorientierten Funktionen zur Verfügung, die über den eigentlichen Botschaftenaustausch hinausgeht. Es sollte eine einheitliche Darstellung des Aufrufs dieser Dienste mit Hilfe von Sprachelementen von PEARL (z.B. Datentypen, Prozeduraufrufe u.a.) gefunden werden (Language Binding). PEARL-Programme könnten dann auch bei unterschiedlichen Kompilierern in MAP-Netzen portabel sein.

Literatur:

Giese, E.; K. Görgen; E. Hirsch; G. Schulze; K. Truol:
Dienste und Protokolle in Kommunikationssystemen - Die Dienst- und Protokollschnittstellen der ISO-Architektur, Springer-Verlag 1985

Janetzky, D.; K.S. Watson:
Token Bus Performance in MAP and RPOWAY;
IFAC Workshop on Distributed Computer Control Systems, Mayschoss (FRG), Sept., 30 - Oct.,2,1986

Kaminski, M.:
Protocols for Communicating in the Factory;
IEEE Spectrum, April 1986

MAP 2.1:
General Motors Corporation, General Motors Technical Center, Warren, Michigan

MAP User's Group Meeting Summary:
MAP user's group meeting Sept. 10 - 11,1985, Anaheim, California; North-Holland, 1986

PROWAY C; IEC Process Data Highway:
IEC Publication 955, 1986

Schwartz, M.:
Telecommunication Networks;
Addison-Wesley, 1987

Suppan-Borowka, J.; T. Simon:
MAP-Datenkommunikation in der automatisierten Fertigung;
Datacom-Verlag, 1986

Zwoll, K.; H. Halling:
MAP-Automation Manufacturing Protocol;
Proceedings of the IFAC/IFIP-Symposium on Software for Computer Control, Graz, May, 20-23, 1986