

# Diagnostik und Wartung an Multimikrorechnersystemen

Dipl.-Ing. WOLFGANG MATTHES

Mit diesem zweiteiligen Beitrag veröffentlichen wir Hinweise zum Betrieb von Multimikrorechnersystemen, die sich inhaltlich an unsere im H. 12 (1985) abgeschlossene Reihe „Multimikrorechnersysteme“ anschließen. Hier werden Fehlersuchstrategien und Testmittel, aber auch rechnerinterne Wartungssysteme beschrieben.

Ohne Vorkehrungen für Diagnostik und Wartung können komplexe elektronische Systeme weder wirtschaftlich gefertigt noch sinnvoll eingesetzt werden. Um die entsprechenden Erfordernisse bei Multimikrorechnersystemen deutlich veranschaulichen zu können, sei zunächst die traditionelle Vorgehensweise kurz dargestellt.

Diagnose- und Wartungsvorkehrungen haben den Zweck, die Funktionsfähigkeit des Systems überprüfen und im Fehlerfall möglichst schnell wiederherstellen zu können. Diese Aufgaben sollen mit Verifizierung (Überprüfung der Funktionsfähigkeit) und Lokalisierung (Auffinden der defekten Leiterplatten, Bauelemente usw.) bezeichnet werden. Die Geräte sind üblicherweise in Struktur und Funktion dokumentiert (z. B. Stromlaufpläne, Impulsdigramme, Funktionsbeschreibungen). Die einfachste Form der Verifizierung besteht im Beobachten der normalen Betriebsabläufe: Wenn Abweichungen auftreten, kann auf das Vorhandensein eines Fehlers geschlossen werden. Die darauf folgende Lokalisierung besteht darin, anhand der Dokumentation, die neben der Struktur faktisch das Sollverhalten beschreibt, die Abweichungen zu ermitteln und durch Messungen, Versuche, Überlegungen usw. die Ursache des Fehlers zu finden. Dies erfordert häufig viel Zeit und stellt hohe Anforderungen an die Qualifikation des Wartungspersonals. Zur Rationalisierung der Prozesse können Testabläufe, spezielle Prüfeinrichtungen u. ä. vorgesehen werden. Damit läßt sich die Fehlerlokalisierung bis zum Auffinden der verdächtigen Funktionseinheit beträchtlich beschleunigen.

Bei Multimikrorechnersystemen ist es kaum praktikabel, bei der Fehlersuche ausschließlich von den normalen Funktionsabläufen auszugehen. Während die Struktur gut zu dokumentieren ist, lassen sich die normalen Funktionsabläufe kaum sinnvoll zur Fehlersuche einsetzen. Die zugrunde liegende Software ist zumeist kompliziert und umfangreich, und die Abläufe selbst sind mit einfachen Mitteln kaum beobachtbar. Selbst die umfangreichste Dokumentation ist keine Garantie für eine erfolgreiche Fehlersuche. Es ist somit unbedingt erforderlich, die Problematik von Wartung und Diagnose im gesamten Entwicklungsprozeß eines Multimikrorechnersystems von vornherein zu berücksichtigen. Wesentliche Aspekte sind:

- Die Betriebsfähigkeit aller Funktionseinheiten sowie deren komplexes Zusammenwirken muß mit spezifischen Mitteln verifizierbar sein. Dazu ist eine entsprechende Testsoftware zu schaffen. An kritischen Stellen sind Hardwaremaßnahmen vorzusehen (z. B. zur Nachbildung externer Einrichtungen).
- Es sollte möglich sein, mit einfachen Mitteln verdächtige Funktionseinheiten zu lokalisieren.

- Es ist von vornherein zu entscheiden, auf welche Weise defekte Bauelemente lokalisiert werden sollen. Eine Möglichkeit besteht im detaillierten Testen der verdächtigen Funktionseinheiten auf automatischen Testeinrichtungen. Die Alternative dazu besteht in Vorkehrungen, die es ermöglichen, defekte Bauelemente mit erträglichem manuellem Aufwand im Gerät selbst zu lokalisieren. Die Hardware muß von Anfang an dem gewählten Prinzip entsprechend ausgelegt werden, wobei für die Planung des Entwicklungsablaufes damit gerechnet werden muß, daß Änderungen nicht nur aus funktionellen Gründen, sondern auch zur Gewährleistung der Prüfbarkeit erforderlich werden können.
- Es sind entsprechende Ressourcen vorzusehen, also Mittel zum Auslösen von Testabläufen und zum Anzeigen von Testergebnissen, Speicherplatz für Testroutinen u. ä.
- Es ist zu erwägen, welche zusätzlichen Testhilfen bereitgestellt werden sollen. Dies betrifft zum einen Hilfsmittel zum Auffinden von Fehlern in der Software und zum anderen Vorkehrungen zur Wartung und Diagnose eingeschlossener Einrichtungen, um besondere Prüfmittel einzusparen. Beispielsweise gibt es Prüfeinrichtungen für Floppy-Disk-Laufwerke, die Stimulussignale für Fehlersuch- und Justagearbeiten liefern, z. B. zyklische Positionierimpulse, einen Datenstrom mit definiertem Bitmuster usw. Alternativ dazu kann man in einem Multimikrorechnersystem, das mit solchen Laufwerken ausgerüstet ist, einen Satz ROM-residenter Testroutinen vorsehen, die die erforderlichen Funktionen ausführen, so daß keine besonderen Prüfeinrichtungen benötigt werden.
- Die Hardware- und Softwaremaßnahmen sind ihrerseits mit der Dokumentation in Übereinstimmung zu bringen, so daß Übersichtlichkeit und Folgerichtigkeit gewährleistet sind. Stromlaufpläne u. ä. werden zum Auffinden von Bauelementen, Anschlüssen, Kabelverbindungen usw. stets erforderlich sein. Sonst sollte die gesamte Konzeption darauf ausgelegt werden, daß Verifizierung sowie die Lokalisierung verdächtiger Funktionseinheiten neben einer relativ knappen Hardware- und Funktionsbeschreibung lediglich ein Wartungshandbuch erfordern, d. h. eine Dokumentation, die im wesentlichen folgendes enthält:
  - Übersicht über die vorhandenen Diagnose- und Wartungsvorkehrungen
  - Beschreibung des Einsatzes und der Bedienung dieser Mittel
  - Interpretation von Fehleranzeigen
  - Empfehlungen zu Handlungsweisen bei Funktionsstörungen

- Prozeduren zur Fehlersuche (in Form von Tafeln, Flußbildern o. ä., die für bestimmte Fehlerbilder eine systematische Vorgehensweise aufzeigen)
  - zusätzliche Information, z. B. Kodetabellen, Belegung von Steckverbindern, Hinweise zum Einstellen von DIL-Schaltern auf den Leiterplatten usw.
- Zum Suchen von Hardwarefehlern sollte es nicht notwendig sein, auf Assemblerausdrücke oder dergleichen zurückgreifen zu müssen.

## Softwareorganisation

Die Testsoftware muß so organisiert sein, daß sowohl die Überprüfung des gesamten Systems auf Funktionsfähigkeit (Verifizierung) als auch die Fehlerlokalisierung in angemessener Zeit und mit erträglichem Arbeitsaufwand ausführbar sind.

Eine bekannte Organisationsform ist die des aufbauenden Tests (engl. Bootstrap Diagnostics). Dabei wird zunächst die Betriebsfähigkeit eines relativ kleinen Funktionskomplexes (engl. Hardcore) geprüft. Ist dieser in Ordnung, wird mit dessen Hilfe die nächste Funktionseinheit getestet. So geht es weiter. Schließlich wird das Zusammenwirken der Funktionseinheiten kontrolliert. Der Test wird mit einer Prüfung der komplexen Betriebsfähigkeit aller Funktionseinheiten abgeschlossen.

Durch dieses Schema ist von vornherein eine gewisse Lokalisierung verdächtiger Funktionseinheiten gegeben. Der Zeitbedarf kann jedoch recht hoch werden. So muß z. B. für das Überprüfen eines RAM von 32 Kbyte mit einer Größenordnung von 30 s bis zu mehreren Minuten gerechnet werden.

Damit wird es sinnvoll, gelegentlich mit einem Komplextest zu beginnen, um schnell Aufschluß über die Betriebsfähigkeit des Systems zu erlangen. Der länger dauernde Bootstraptest wird nur dann ausgeführt, wenn der Komplextest Fehler signalisiert hat oder wenn während des normalen Betriebs Störungen festgestellt wurden.

Weitere Aspekte für die Gestaltung der einzelnen Testabläufe und für die Ablauforganisation:

- Möglichkeiten zum selektiven Aufruf einzelner Testabläufe für die gezielte Fehlersuche
- Möglichkeiten zum zyklischen Abarbeiten sowohl der einzelnen Tests als auch komplexer Testabläufe, um die Zuverlässigkeit des Systems bzw. einzelner Funktionseinheiten prüfen und transiente Fehler lokalisieren zu können. Weiterhin haben sich bewährt:
  - Betrieb unter Grenzbedingungen der Versorgungsspannungen
  - Veränderung der Taktfrequenz
  - Erwärmen bzw. Abkühlen verdächtiger Bauelemente.

- Unterscheidung zwischen automatisch ablaufenden Tests und solchen, die manuelle Eingriffe (z. B. das Stecken von Kurzschlußsteckern) erfordern. Letztere sollten stets separat aufgerufen werden und in Komplextests entweder gar nicht enthalten bzw. durch Bedienhandlungen ausschließbar sein. Dies gilt auch für Testabläufe, die periphere Einrichtungen betreffen (z. B. Beschreiben von Spuren auf Floppy Disks, Druckertest mit eingelegtem Papier).
- Bereitstellung von Fehlerinformation. Die einfachste Lösung, bei Feststellung eines Fehlers den Testablauf anzuhalten und diesen Zustand anzuzeigen, ist für Lokalisierungszwecke nicht besonders wirksam. Besser ist es, den Testablauf zu Ende zu führen und für alle Fehlersituationen die näheren Umstände aufzuzeichnen. Manchmal ist dies von vornherein nicht möglich. Dann hat es sich bewährt, folgende Informationen zu erfassen:

- Anzahl der Fehler
- detaillierte Fehlerinformation für eine gewisse Zahl von Fehlern (z. B. für die ersten vier Fehler).

Beispiel: Wird bei einem RAM-Test nur angezeigt, daß ein Fehler aufgetreten ist, so liefert dies keine weitere Information für dessen Lokalisierung. Die Anzeige von Adresse, Sollwert und Istwert des ersten Fehlers ermöglicht eine etwas bessere Fehlerzuordnung, genauere Aussagen sind aber nicht immer möglich (z. B. die Entscheidung, ob der Fehler im Adressen- oder im Datenpfad zu suchen ist). Bei Aufzeichnung mehrerer Fehlersituationen sind solche Aussagen eher möglich (z. B. wenn Fehler nur in einer Datenbitposition auftreten, wenn sie nur dann auftreten, wenn ein bestimmtes Adressenbit einen bestimmten Wert hat usw.).

- Ausschluß verdächtiger Funktionseinheiten von der Testung, physisches Entfernen und Untereinander-Tauschen von Funktionseinheiten. Die Testabläufe sollten so organisiert sein, daß es möglich ist, diese einfachen, aber recht wirkungsvollen Fehlersuchmethoden anzuwenden. Die Hardware muß natürlich ebenfalls dafür eingerichtet sein (z. B. Betriebsfähigkeit des Bussystems nach Entfernen von Funktionseinheiten). Als besonders günstig erweist sich eine weitgehende Standardisierung. Geeignet wäre z. B. ein einziger Mikrorechner mit standardisiertem ROM-Inhalt, wobei alle funktionsbestimmenden Programme aus RAM-Bereichen abgearbeitet werden, so daß jeder Mikrorechner für jede Funktion einsetzbar ist. Durch das Ausschließen verdächtiger Funktionseinheiten läßt sich ermitteln, ob der Fehler tatsächlich von der betreffenden Funktionseinheit verursacht wurde. Das physische Entfernen empfiehlt sich besonders dann, wenn der Verdacht besteht, daß die betreffende Funktionseinheit durch einen Defekt den Testablauf indirekt beeinflußt oder das Starten von Testabläufen überhaupt verhindert (etwa durch einen defekten Buskoppelbaustein).
- Erreichbarkeit der zu testenden Bau-

gruppen in den Funktionseinheiten. In vielen Funktionseinheiten sind die zu testenden Baugruppen nicht direkt für die Diagnose zugänglich, d. h., es gibt keine Möglichkeit, Signale direkt einzuspeisen und die jeweilige Reaktion abzufragen. Somit kann die betreffende Baugruppe nur indirekt (mit passend gewählten Funktionsabläufen) geprüft werden, so daß nur in seltenen Fällen die Testsoftware eine exakte Fehlerlokalisierung (etwa bis zum verdächtigen Signal) gewährleisten kann. Die Erarbeitung der Abläufe für indirekte Prüfungen erfordert zudem einen beachtlichen Arbeitsaufwand. Bild 1 zeigt als Beispiel für einen indirekten Test die Überprüfung externer Wartebedingungen einschließlich der Möglichkeit, diese Bedingungen programmtechnisch abzuweisen.

### Hardwaremaßnahmen

Elementare Hardwaremaßnahmen bestehen darin, Speicherplatz sowie Bedien- und Anzeigemittel für die Testsoftware bereitzustellen. Weiterhin ist zu gewährleisten, daß ein angemessener Teil des Systems in sich, d. h. ohne Rückgriff auf die angeschlossenen Einrichtungen, automatisch geprüft werden kann. Damit sind die Voraussetzungen für die Verifizierung des Systems und die Lokalisierung verdächtiger Funktionseinheiten im allgemeinen gegeben.

Eine weitergehende Automatisierung hängt davon ab, in welchem Maße die einzelnen Baugruppen für die detaillierte Prüfung zugänglich sind. Eine übliche Maßnahme besteht darin, zu gewährleisten, sämtliche Flip-Flops einer Funktionseinheit einstellen und abfragen zu können. Dazu hat es sich bewährt, alle Flip-Flops zu Master-Slave-Flip-Flops zu erweitern und diese in Form von Schieberegistern mit der prüfenden Einrichtung zu verbinden. Das Prüfproblem

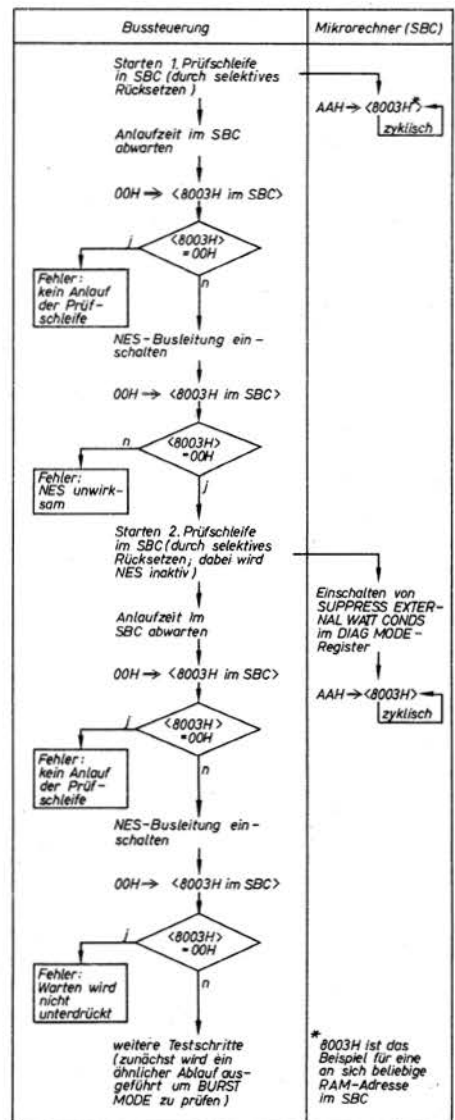


Bild 1: Prüfung der Reaktion auf externe Wartebedingungen

Tafel 1: Vor- und Nachteile der Lokalisierungsarten für verdächtige Bauelemente

Methode	Vorteile	Nachteile	Anwendungsgebiete
1.	Austauschbare Einheit (Leiterplatte) ist einfach und billig. Testvorkehrungen erfordern relativ geringen Aufwand, Dokumentation unproblematisch. (Stromlaufpläne und Prüf-anweisungen). Es kann darauf vertraut werden, daß der Techniker die jeweilige Teilfunktion durchdenken kann, so daß Mängel in der Dokumentation bzw. außergewöhnliche Fehlersituationen nicht zu besonderen Problemen führen. Die Prüfung kann weitgehend außerhalb des Systems erfolgen.	Große Anzahl an Steckverbindern und IS, System ist physisch ausgedehnt, systemspezifische Prüfmittel zumeist unumgänglich. Die Prüfung der Leiterplatten ist kein Funktionstest, für Leiterplatten mit hochintegrierten IS (z. B. dynamische RAMs) fehlt Teststimulus unter Echtzeitbedingungen.	industrielle Steuerungen
2.	Höhere Zuverlässigkeit und geringere Hardwarekosten durch minimale Zahl von Steckverbindern, IS und Leiterplatten, kompakter Aufbau, keine spezifischen Prüfmittel erforderlich, Prüfung unter Echtzeitbedingungen (realistischer Stimulus für LSI-IS).	Dokumentation muß exakt und detailliert sein, wegen der Komplexität kann nicht darauf vertraut werden, daß sich der Techniker auch in problematischen Fällen stets behelfen kann (damit höhere Anforderungen an Schulung bzw. Kundendienst-Organisation), Prüfung außerhalb des Systems nur unvollständig möglich, für die Prüfung in der Fertigung ist ein Mustergerät o. ä. oft unumgänglich.	komplexe und kompakte Meßgeräte, Computer, EDV-Peripherie usw.

Tafel 2: Vergleich der Signaturanalyse mit anderen Prüfverfahren

Prinzip	Meßgeräte-kosten	Zeitbedarf für Messung	Kriterien für Impulsfolgen	gleichzeitig zu prüfende Signale	Dokumentation	Fehleraussage	Anforderungen an Stimulus
Oszilloskop	mittel bis hoch	gering (wenige Sekunden je Signal) (SYNC-Signal anschließen, Oszilloskop einstellen, zu prüfende Signale nacheinander antasten)	zyklisch, geringe Anzahl von Impulsen im Zyklus	1 bis 4	Auslösung, Meßpunkte, Sollabläufe; bei Änderungen ist exaktes Nachführen nicht in allen Details notwendig	sehr genau auf elektrischer Ebene (Flanken, Impulsformen, Pegel usw.) oft wenig detailliert hinsichtlich komplexer logischer Zusammenhänge	zyklisch, wenige Impulse; Fehler im Stimulus direkt erkennbar
Logikanalyzer	hoch	mittel bis hoch (einige Minuten) (alle zu prüfenden Signale anschließen, Analyzer einstellen)	Impulsanzahl durch Speichertiefe des Analyzers begrenzt, sonst keine Einschränkungen	8 bis 32 (oder mehr)	Auslösung, Meßpunkte, Sollabläufe; bei Änderungen ist exaktes Nachführen nicht in allen Details notwendig	sehr detailliert hinsichtlich komplexer logischer Zusammenhänge, Rückschlüsse auf elektrische Ebene nur bedingt möglich (z. B. Auftreten von Störimpulsen)	keine besonderen Forderungen, Erkennen von Fehlern im Stimulus möglich, erfordert aber detaillierte Kenntnisse
Signaturanalyse	gering	gering (wenige Sekunden je Signal) (Start-, Stopp- und Taktsignale anschließen, Analyzer einstellen, zu prüfende Signale nacheinander antasten)	zyklisch, viele Impulse (mehrere tausend) je Zyklus möglich	1	Auslösung, Meßpunkte, eine Signatur (vierstellige Angabe) je Signal; bei Änderungen ist exaktes Nachführen aller Sollsicherungen notwendig	Ja-Nein-Aussage bezüglich der logischen Funktion, keine Rückschlüsse auf funktionelle Zusammenhänge möglich	zyklisch; ist der Stimulus selbst fehlerhaft, kann das Verfahren nicht angewendet werden

reduziert sich dabei auf die Prüfung der Schiebewege selbst und auf das Testen kombinatorischer Schaltungen. Mikrorechnersysteme haben allerdings einige Besonderheiten, die die Einführung rigoroser Prüfprinzipien außerordentlich erschweren: Sie basieren auf hochintegrierten Schaltkreisen, die sehr komplexe Funktionen realisieren und nicht explizit für Testzwecke zugänglich sind, sowie auf Speicherschaltkreisen, für die generell spezifische Prüfprinzipien (Prüfsummen bei ROMs, spezielle Testmuster bei RAMs) verwendet werden müssen. Die meisten der SSI- bzw. MSI-IS realisieren an sich nur elementare Verknüpfungsfunktionen. Sequentielle Schaltungen sind zumeist sehr einfach aufgebaut, und im Interesse einer geringen Anzahl von Bauelementen sind deren funktionelle Möglichkeiten zum großen Teil ausgenutzt. Eine weitgehende Zugänglichkeit würde einen enormen Zusatzaufwand erfordern. Für die Lokalisierung verdächtiger Bauelemente werden stets manuelle Messungen oder der Einsatz von Testeinrichtungen notwendig sein.

**Lokalisierung verdächtiger Bauelemente**

Ist der Einsatz hochentwickelter automatischer Testeinrichtungen nicht praktikabel, so verbleiben zwei Alternativen (s. Tafel 1):

1. Die betreffende Funktionseinheit wird auf eine größere Zahl von Leiterplatten aufgeteilt, deren Inhalt so strukturiert wird, daß die Prüfung mit einfachen Steckeinheitenprüfgeräten oder mit Betriebsmeßmitteln möglich ist.
2. Maßnahmen werden vorgesehen, die eine Fehlerlokalisierung mit handelsüblichen Meßmitteln auch in relativ komplexen Funktionseinheiten ermöglichen.

Im folgenden soll die zweite Alternative näher betrachtet werden. Dabei sind die verdächtigen Bauelemente durch Messungen zu ermitteln, wobei die erforderlichen Stimulussignale durch die Funktionseinheit selbst oder im Rahmen des gesamten Systems erzeugt werden. Typische Meßmittel

sind z. B. Oszilloskope oder auch Logikanalyzer. Ein weiteres Verfahren, das speziell für die hier interessierenden Probleme entwickelt wurde, ist die Signaturanalyse. Sie beruht darauf, daß die jeweils zu kontrollierende zyklische Impulsfolge über ein rückgekoppeltes Schieberegister geführt wird, wobei der resultierende Inhalt des Schieberegisters als Signatur angezeigt wird. Das Verfahren hat offensichtliche Vorteile, die aus der Gegenüberstellung zu mehr traditionellen Prinzipien besonders deutlich werden (Tafel 2).

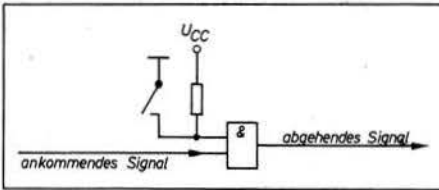
Andererseits ist ersichtlich, daß sich das Verfahren kaum als ausschließliche Grundlage der Fehlerlokalisierung eignet. Um zu bestimmen, welche Verfahren in welchen Fällen einzusetzen sind, können folgende Überlegungen herangezogen werden:

- Die Fehlersuchmaßnahmen dienen zum Suchen von Hardware-Ausfällen. Es ist realistisch anzunehmen, daß bei Auslieferung des Systems (eine sorgfältige Erprobung vorausgesetzt) keine Hardwarefunktionsfehler mehr existieren, da alle komplexen Funktionen durch Software bzw. Firmware realisiert sind.
- Problematische Defekte in LSI-Schaltkreisen (z. B. gelegentliche Datenverluste in dynamischen RAMs) können oft durch Testsoftware lokalisiert werden. Da in Multimikrorechnersystemen sicherlich wenigstens ein Mikrorechner als funktionsfähig vorausgesetzt werden kann, ist stets gewährleistet, daß die Prüfprogramme auch abgearbeitet werden können. Aufwendungen zur Schaffung einer entsprechenden Teststrategie und Ablauforganisation betreffen nur den Entwicklungsprozeß, und zusätzliche Aufwendungen (im wesentlichen Speicherplatz für die Testprogramme) fallen kaum ins Gewicht.
- Die meisten Fehler sind Ausfälle (Stuck-at-0 bzw. Stuck-at-1 einzelner Signale). Diese sind relativ leicht zu finden, ohne daß komplexe Abläufe im Detail betrachtet werden müssen.
- Bei entsprechender Gestaltung von

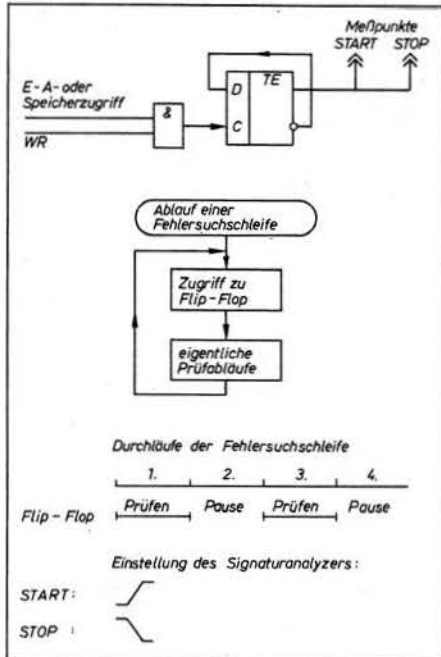
Hardware und Testsoftware ist eine elementare Vorprüfung mit dem Oszilloskop leicht zu organisieren. Damit ist prüfbar, ob ein geeigneter Stimulus für die Signaturanalyse verfügbar ist oder nicht (in diesem Fall ist der Fehler direkt durch oszilloskopische Messungen zu lokalisieren). Wird der jeweilige Stimulus korrekt erzeugt, so kann für das Überprüfen der weitaus meisten Signale die Signaturanalyse verwendet werden.

- Die Testsoftware kann weiterhin so strukturiert werden, daß sich durch systematisches Anwenden der einzelnen Tests bereits eine gewisse Lokalisierung verdächtiger Baugruppen (z. B. der CPU, der Speichervermittlung, der Masterzugriffssteuerung usw.) ergibt. Die genaue Überprüfung mit Oszilloskop bzw. Signaturanalyse kann sich dann auf diese Baugruppen konzentrieren, so daß es nicht notwendig ist, bei jedem Fehler die gesamte Funktionseinheit von Grund auf im Detail zu untersuchen.
- Der Einsatz eines Logikanalyzers kann speziellen Problemen vorbehalten bleiben, die mit den anderen Methoden nicht zu bewältigen sind.

Ein praktisch brauchbares Fehlersuchprinzip kann deshalb auf oszilloskopische Messungen und Signaturanalyse gestützt werden. Oszilloskopische Messungen dienen im wesentlichen zur Überprüfung elementarer Grundfunktionen (Taktversorgung, elementare Signalfolgen von CPUs, Funktionsfähigkeit von Zählern und Steuerketten). Die weitaus meisten Überprüfungen beruhen auf der Signaturanalyse (Datenwege, Adressenwege, Register). Damit bleibt der Umfang der beschreibenden und in Bildform darzustellenden Dokumentation (z. B. Oszillogramme mit Sollabläufen) in Grenzen. Sollsicherungen können in einfachen Listen zusammengefaßt werden. Wird ein Fehler mit der Signaturanalyse festgestellt, so empfiehlt sich nachfolgend die Benutzung des Oszilloskops, um das elek-



**Bild 2: Auftrennung von Rückführungen mit DIL-Schalter**



**Bild 3: Erzeugung von Start-, Stopp- bzw. SYNC-Impulsen mit einem programmtechnisch stellbaren Flip-Flop**

trische Verhalten der betreffenden Signale beurteilen zu können (z. B. Stuck-at-0, undefinierter Pegel, Störspitzen, verschliffene Flanken). Damit kann der Techniker in vielen Fällen die Fehlerursache bestimmen, ohne daß eine exakte Kenntnis der funktionellen Abläufe erforderlich ist. Der Techniker braucht also kaum über komplexe Funktionsabläufe nachzudenken, um herauszufinden, weshalb recht umfangreiche

Funktionseinheiten nicht funktionieren, sondern er wird durch die Kombination aus Testsoftware und Dokumentation bis zu Komplexen geführt, die aus wenigen Bauelementen bestehen, so daß sich beispielsweise das Problem ergibt, zu überlegen, weshalb ein Dekoder keine Ausgangssignale abgibt oder ein Zähler nicht zurückgesetzt wird.

Das Fehlersuchprinzip erfordert einige Vorkehrungen, die allerdings nicht sehr aufwendig sind. Beispiele:

- Gewährleistung eines zyklischen Stimulus auch für sehr elementare Abläufe. So kann in einem Mikrorechner der Übertragsausgang des Refreshadressenzählers auf das allgemeine Rücksetzsignal durchschaltbar sein (mit DIL-Schaltern). Da der Zähler eine vollkommen autonome Baugruppe darstellt, können Fehler dort leicht gefunden werden. Für die Fehlersuche in Folgesteuern bietet das zyklische Zurücksetzen günstige Möglichkeiten.
- Gewährleistung des Freilaufs von Mikroprozessoren. Dazu wird über DIL-Schalter der Datenbus konstant mit 00H beschaltet. So führt der Mikroprozessor zyklisch NOP-Befehle im gesamten Adressenbereich aus. Damit ist dessen elementare Betriebsfähigkeit prüfbar.
- Auftrennung von Rückführungen. Dies ist aus Aufwandsgründen sicherlich nicht überall möglich, sollte aber für signifikante Rückführungen vorgesehen werden. Das erfolgt über DIL-Schalter gemäß Bild 2, so daß im Normalbetrieb alle Kontakte geöffnet sind und sich Kontaktunzuverlässigkeiten nicht auswirken.
- Verlangsamung bzw. Synchronisation von Taktimpulsen. Bei sehr hohen Taktfrequenzen sind manche Prüfgeräte nicht mehr einsetzbar. Beim Zusammenwirken verschiedener Funktionseinheiten mit jeweils eigener Takterzeugung ist das Messen schwierig bzw. (bei Signaturanalyse) nicht möglich. Zu empfehlen

**Tafel 3: Anforderungen an den Teststimulus für oszilloskopische Messungen und Signaturanalyse**

Prüfprinzip	Kriterien
Oszilloskop	wenige Taktzyklen, Dauer des zu prüfenden Signals ist nicht zu kurz im Vergleich zur Zykluszeit (z. B. sind in einem Intervall von 1 ms drei Impulse von 200 ns kaum beobachtbar), relative Lage von SYNC-Impulsen und zu prüfenden Signalen weitgehend unkritisch
Signaturanalyse	Anzahl der Taktimpulse unproblematisch, Lage der zu prüfenden Signale relativ zum Prüftakt muß beachtet werden (in bezug auf die Setup-Zeit des Analyzers usw.). Prüftakt muß praktisch ständig anliegen, da Start- und Stoppbedingungen vom Analyzer getaktet übernommen werden. Die Anzahl der Prüftakte zwischen Start und Stopp (d. h. im Meßintervall) muß stets konstant sein.

sind deshalb Vorkehrungen, um die Takte in den Funktionseinheiten – zu verlangsamen (z. B. durch einen einschaltbaren Verteiler) – einen externen Takt einzuspeisen.

- Vorkehrungen für softwaregesteuerte Fehlersuchschleifen. Für viele Baugruppen kann der Testimpuls durch Software erzeugt werden (in der Funktionseinheit selbst oder durch andere Funktionseinheiten, die z. B. Zugriffe über den Systembus ausführen). Derartige Routinen müssen lediglich zyklische Stimulusimpulse liefern (z. B. für Zugriffe zu bestimmten Speicherbereichen oder Registern). Sie führen keine Fehlerkontrollen aus und sind deshalb wesentlich einfacher als übliche Testprogramme. Es wird allerdings eine größere Zahl derartiger Abläufe benötigt. So ist zu empfehlen, für die Überprüfung von Speicherzugriffen zwei Ausführungen vorzusehen, nämlich eine, die nur Schreibzugriffe und eine, die Schreib- und Lesezugriffe ausführt. Da alle Daten auf einem einzigen bidirektionalen Datenbus transportiert werden, ist es wesentlich zu unterscheiden, ob Fehler bereits beim Belegen des Datenbusses oder beim Lesen aus dem Speicher verursacht werden. Mit der ersten Prüfschleife kann die Datenbusbelegung geprüft werden. Wurden keine Fehler festgestellt, so ist mit der zweiten Schleife das Lesen des Speichers prüfbar. Treten dabei Fehler auf dem Datenbus auf, so ist der Speicher selbst verdächtig.

Derartige Schleifen sind sowohl für oszilloskopische Messungen als auch für die Signaturanalyse brauchbar. Nähere Einzelheiten veranschaulicht Tafel 3.

- Anordnung spezifischer Meßpunkte für Synchronisation (oszilloskopische Messungen) bzw. Start, Stopp und Takt (Signaturanalyse), um die Rüstzeiten für die einzelnen Messungen zu vermindern. So können alle Start-, Stopp- bzw. SYNC-Impulse bei softwaregestützten Fehlersuchschleifen von einem einzigen programmtechnisch stellbaren Flip-Flop geliefert werden (Bild 3).

## Bedienung und Anzeige

Zum Auslösen der Testfunktionen und zum Anzeigen der Testresultate sind entsprechende Mittel erforderlich. Deren Ausgestaltung und die Organisation der Testsoftware hängen wechselseitig voneinander ab: Eine umfangreiche Testsoftware, die es ermöglicht, eine Vielzahl von Testroutinen, Fehlersuchschleifen usw. flexibel einzusetzen und detaillierte Information über die Fehlersituation zu ermitteln, benötigt entsprechende Einrichtungen für Bedienung und Anzeige. Andererseits hat es kaum Sinn, eine flexible Testauswahl- und Ablaufsteuerung vorzusehen oder Fehlersituationen im Detail programmtechnisch zu analysieren, wenn die Bedien- und Anzeigemittel dazu fehlen. Wesentliche Gesichtspunkte für die entsprechenden konzeptionellen Überlegungen sind:

- Bedien- und Anzeigemittel erhöhen die Kosten, sind selbst fehleranfällig und müssen ihrerseits überprüft werden. Andererseits treten in üblichen Multimikrorechnersystemen (500 bis 2000 IS) Hardwarefehler relativ selten auf (Ausfallabstand mindestens einige hundert Stunden). Deshalb sollte der Aufwand für solche Einrichtungen, die ausschließlich zu Diagnosezwecken vorgesehen sind, nicht übermäßig erhöht werden.
- Für normale Betriebsfunktionen vorhandene Einrichtungen (Tasten, LEDs, Bildschirme, Drucker usw.) können oft zu Diagnosezwecken benutzt werden.
- Da zur Fehlersuche ohnehin die einschlägige Dokumentation benötigt wird, können zur Testauswahl bzw. zur Anzeige von Testergebnissen unbedenklich beliebige und u. U. ungewöhnliche Kombinationen von Bedienhandlungen usw. darzustellen, reicht stets eine Liste, Tabelle o. dergl. von wenigen Seiten Umfang aus. Andererseits kann man eher damit rechnen, daß sich die Fehlersuchzeiten verkürzen und die aktuelle Fehlersituation detailliert und gut auswertbar angezeigt wird.
- Für die normale Benutzung ist die Funktionsfähigkeit bedeutsam, nicht der Fehlerzustand. Es ist aber wesentlich, zwischen beiden zu unterscheiden. Deshalb sollte für eine eindeutige Anzeige der Betriebsfähigkeit gesorgt werden. Ebenso muß sich eine Überprüfung auf Funktionsfähigkeit (Verifizierung) einfach auslösen lassen. Beispielsweise kann mit einem Schlüsselschalter zwi-

schen Normal- und Wartungsbetrieb umgestellt werden. Nach dem Einschalten bzw. dem Betätigen einer Testtaste wird die Funktionsfähigkeit des Systems überprüft (Komplextest). Ist diese gewährleistet, so leuchtet eine Anzeige O. K. auf, und es wird zum normalen Betrieb übergegangen. Ist das nicht der Fall, so leuchtet eine Fehleranzeige. Mit anderen Anzeigeeinrichtungen wird gleichzeitig ein spezieller Fehlercode dargestellt.

- Reichen die normalerweise vorhandenen Bedien- und Anzeigemittel nicht aus, so können für Testzwecke zusätzliche vorgesehen werden. Es hat sich dabei bewährt, diese entweder gerätespezifisch und so einfach wie möglich auszulagern oder auf standardisierte wie Drucker, Tastaturen, Bildschirmgeräte zurückzugreifen.

Mikrorechner können mit einem Anschluß für einen Diagnoseadapter nach Bild 4 ausgerüstet werden. Dazu ist der interne Datenbus an einen speziellen Steckverbinder geführt. Zusätzlich werden in der jeweiligen Einrichtung ein Signal für Schreibzugriffe und eines für Lesezugriffe dekodiert, wodurch es möglich ist, die Stellungen von acht DIL-Schaltern abzufragen und zwei Siebensegmentanzeigen anzuzeigen. Der Diagnoseadapter liefert außerdem ein Rücksetzsignal, das z. B. durch ein Sensorelement ausgelöst werden kann, sowie Steuersignale für den Freilauf der jeweiligen CPU.

Weiterhin ist noch von Bedeutung, daß die

betreffende CPU abfragen kann, ob der Diagnoseadapter gesteckt ist oder nicht (Signal DIAG PLUG PRESENCE; z. B. über eine PIO abfragbar). Das Abfragen ist stets eine der ersten Operationen der CPU nach dem Hardwarerücksetzen. Mit dem Diagnoseadapter lassen sich Tests bzw. Fehlersuchschleifen auslösen und Fehlerzustände anzeigen. Dies ermöglicht ein automatisches Überprüfen der Funktionseinheiten sowie das Auslösen elementarer Abläufe für die Diagnose des Zusammenwirkens verschiedener Einrichtungen (Zugriffstests und Fehlersuchschleifen für den Systembus usw.).

Ist eine gewisse elementare Funktionsfähigkeit gewährleistet, so können die vorhandenen Bedien- und Anzeigemittel benutzt werden, um den Bedienkomfort zu verbessern sowie Fehlerzustände detaillierter darzustellen.

Beispielsweise sind bei einem EDV-Gerät eine Tastatur, ein Drucker sowie zwei Bildschirme vorhanden (s. Bild 5 [3]).

Die Bedienung erfolgt über Tastatur und Bildschirm, zur Fehlerausgabe werden Bildschirm und Drucker benutzt. Dazu ist der Drucker über einen Doppeladapter sowohl an die Bussteuerung (für die Ausgabe von Fehlermeldungen) als auch an einen anderen Mikrorechner (für die Druckfunktionen des normalen Betriebes) angeschlossen, wobei die Bussteuerung den Zugriff jederzeit erzwingen kann. Bei Defekten im Bussystem bzw. in den Schaltmitteln für die Bilddarstellung kann somit die Fehlerinformation über den Drucker ausgegeben werden, bei Defekten des Druckers

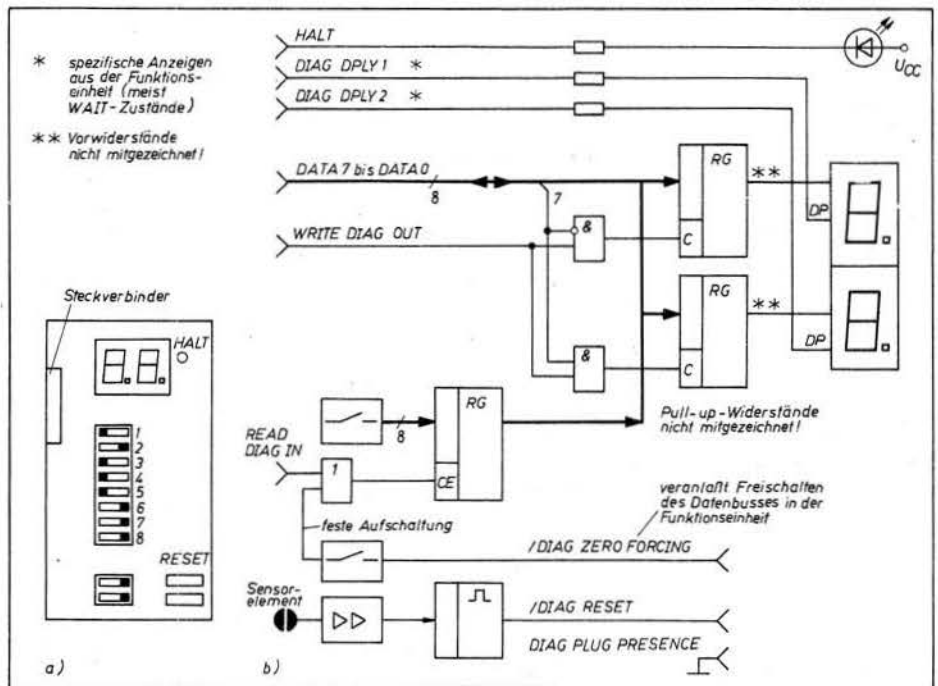
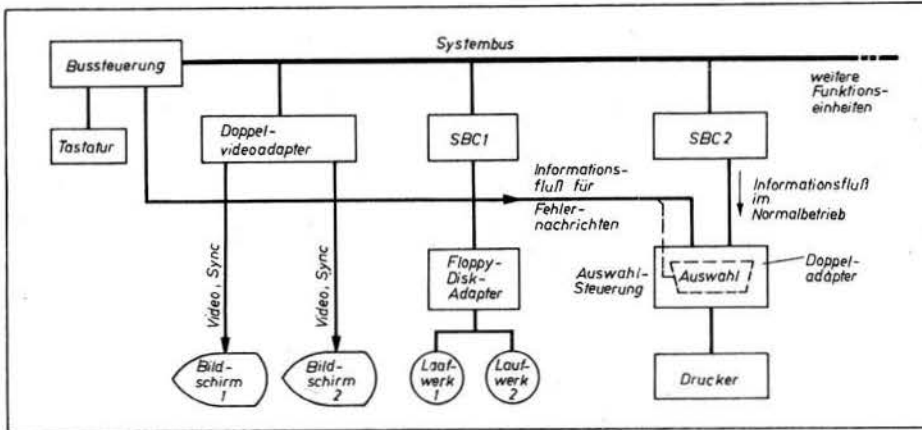


Bild 4: Diagnoseadapter. a) Darstellung; b) Prinzipschaltung



**Bild 5: Anordnung peripherer Einrichtungen zur Ausgabe von Fehlerzuständen und Bedienung eines Wartungssystems**

steht der Bildschirm zur Verfügung. Die übliche Form einer Fehlermeldung ist ein fest formatierter Block, dessen erstes Byte die Fehlermeldung identifiziert (engl. Message Code; für jeden dieser Codes enthält die Dokumentation die Interpretation der nachfolgenden Bytes). Zur Bedienung wird nur ein Teil der Tastatur benutzt, die Bedienung erfolgt über den Bildschirm. Für die verschiedenen Handlungen sind fest formatierte Bilder vorgesehen. Zur Funktionsauswahl werden die Cursorsteuertasten benutzt.

#### Struktur eines Wartungssystems

Bild 6 veranschaulicht die Struktur eines komplexen Wartungssystems, das aufbauende Tests für elementare Prüfungen, Komplextests, Fehlersuchschleifen sowie Mittel zum Suchen von Softwarefehlern umfaßt. Die Bedienung ist wie vorstehend beschrieben organisiert.

#### Besondere Merkmale:

- Programme für komplexere Funktionen stehen nicht in ROMs, sondern sie werden geladen (z. B. von Floppy Disks). Um trotzdem akzeptable Reaktionszeiten (z. B. bei Tastenbetätigungen) zu gewährleisten, wird nach der elementaren Prüfung der Funktionseinheiten (mit ROM-residenten Testroutinen) zunächst das Speichervermögen des gesamten Gerätes dazu benutzt, das komplette Wartungssystem (mit Ausnahme der HELP-Bilder) im RAM zu halten. Von den Mikrorechnern und anderen Funktionseinheiten wird zunächst nur erwartet, daß sie in der Lage sind, Information zu speichern. Diese Fähigkeit wurde zuvor mit ROM-residenten Tests geprüft.
- Es ist möglich, sowohl Einzeltests (gezielt für verdächtige Funktionskomplexe) als auch umfassende Testabläufe mit

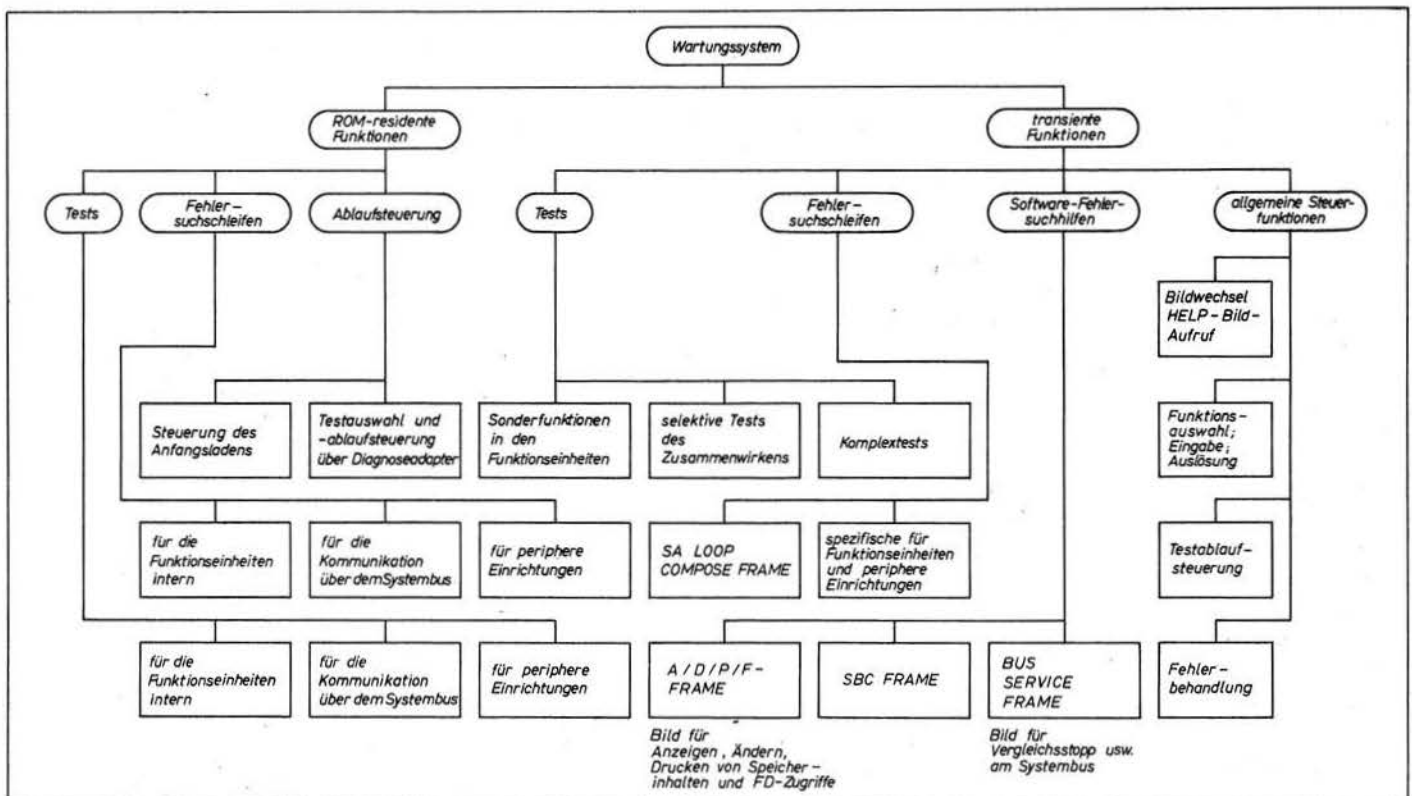
automatischer Fehlerprotokollierung (etwa zur Untersuchung der Betriebszuverlässigkeit) auszulösen.

- Für Wartungsarbeiten an peripheren Einrichtungen stehen Hilfsfunktionen zur Verfügung, die externe Testmittel erübrigen. Weiterhin ist für die interne Fehlersuche die Möglichkeit vorgesehen, über ein spezielles Bildformat (SA LOOP COMPOSE FRAME) Fehlersuchschleifen für Buszugriffe selbst zusammenzustellen, wobei jeder Zyklus bis zu vier an sich beliebige Zugriffe umfassen kann.
- Die Bildschirmanzeigen sind weitgehend selbsterklärend ausgeführt, es werden kaum Abkürzungen benutzt, und zur Bedienung werden zusätzliche Darstellungen eingeblendet, z. B., ob die aktuelle Eingabe hexadezimal oder binär erfolgen soll, wie ein laufender Test abgebrochen werden kann usw. Zusätzlich ist eine größere Zahl von HELP-Bildern aufrufbar, so daß Teile der Dokumentation direkt über den Bildschirm wiedergegeben werden können.

Der Umfang eines solchen Systems ist allerdings beachtlich. Er betrug bei einer realisierten Ausführung mehr als 40 Kbyte ohne HELP-Bilder. Mit Aufwendungen dieser Größenordnung muß bei komplexen Multimikrorechnersystemen stets gerechnet werden.

#### Literatur

- [1] Gottschalk, H.: Verbindungsprogrammierte und speicherprogrammierbare Steuereinrichtungen. RA 209. Berlin: VEB Verlag Technik 1984
- [2] Will, G.: Anwendung der Signaturanalyse in Mikrorechnersystemen. radio fernsehen elektronik, Berlin 34 (1985) 1, S. 22-24, 54
- [3] WP G 06 F/2296 345: Bedien- und Service-Processor, vorzugsweise für EDV-Anlagen



**Bild 6: Struktur eines komplexen Wartungssystems**