

Die Entwicklung des Terminalsteuerrechners MUX30A EC8404.M1

Autor: Heinz Gutbier



Inhaltsverzeichnis

1	Einleitung.....	3
2	Die technische Lösung des MUX30A.....	4
2.1	Hardwarestruktur.....	4
2.2	Schranksystem und Blockeinschübe.....	6
2.2.1	Überblick.....	6
2.2.2	Steuereinschub SE32 mit MUX20.....	7
2.2.3	Datenübertragungseinschübe DUE16 und DUE 32.....	9
2.2.4	ESER-Anschluss-Prozessor	11
2.3	Funktionsgruppen.....	13
2.3.1	Allgemeines.....	13
2.3.2	Mikrorechnerkerne K1520.....	13
2.3.3	Serieller Datenadapter SDA.....	14
2.3.4	Mehrrechnerkoppelbus MRKB.....	15
2.3.5	Buskoppeleinheit BKE 15.....	15
2.3.6	Buskoppeleinheit BKEM 15.....	16
2.3.7	ESER-Kanal-Steuerung EKST.....	17
2.4	Software.....	19
2.4.1	Allgemeines.....	19
2.4.2	ESAP-Steuerprogramm.....	20
2.4.3	Emulationsprogramm EMLMUX.....	21
2.4.4	Test- und Hilfsprogramme	22
2.5	Integriertes Testsystem.....	25
2.5.1	Allgemeines.....	25
2.5.2	Anzeige- und Protokollfunktionen.....	25
2.5.3	Service mit MUX20 im Offlinebetrieb.....	26
2.5.4	Service mittels CCW-Simulation.....	27
3	Geschichte der Entwicklung.....	28
4	Abkürzungsverzeichnis.....	33
5	Literatur- und Bildverzeichnis.....	34

1 Einleitung

Das Kombinat Robotron hatte seit 1975 als Bestandteil seiner EDVA-Systeme einen Datenfern-Multiplexer mit der ESER-Chiffre EC8404 vertrieben. Dieses Gerät war auf der technischen Basis des Prozessrechners R4201 entwickelt worden und diese Herangehensweise hatte im Grundsatz den im ESER geltenden Regeln widersprochen [1].

Zum Ende der 1970er Jahre war der DFV-Prozessor IBM 3705 das Vorbild für Entwicklungen in anderen Ländern des ESER. Dieses System kann in zwei grundsätzlich verschiedenen Betriebsweisen arbeiten. Im Emulationsmodus werden die Vorgängersysteme IBM 270x nachgebildet und damit besteht die Möglichkeit, vorhandene hierarchische Anwendungen weiterhin zu betreiben. Der zum Entwicklungszeitpunkt aktuelle SNA-Modus orientierte sich an einer Netzarchitektur.

Bei Robotron wurde für die zu entwickelnden DFV-Prozessoren ein Systemansatz unter Verwendung der Mikrorechnersysteme K1520 und K1600 vorgegeben. Damit war von Anfang an klar, dass die Software des Vorbildsystems nicht lauffähig sein konnte und dass Passfähigkeiten zu höheren Softwareschichten, wenn überhaupt machbar, nur mit enormem Aufwand erreichbar sein würden. Aus diesen Gründen wurde eine anfangs vorgesehene SNA-Variante schließlich nicht weiter verfolgt. Die Entwicklung der Hardware erfolgte im Werk Robotron-Elektronik Radeberg und wurde so ausgeführt, dass zunächst in einem ersten Schritt Komponenten für die Datenfernverarbeitung mit den Prozessrechnern der Reihe K1600 entwickelt wurden und dass in einem zweiten Schritt das als **Multiplexer EC8404.M1** bezeichnete ESER-Gerät, ein relativ komplexes Multi-Mikrorechner-System, entstanden ist. MUX30A und TSR/A sind zeitweise verwendete gleichwertige Arbeitsbezeichnungen, im Folgendem wird **MUX30A** benutzt.

An der Entwicklung der Software waren Robotron-ZFT Dresden, Robotron-Elektronik Radeberg und Leitzentrum für Anwendungsforschung der VVB Maschinelles Rechnen beteiligt. Die Leitfunktion hinsichtlich der Entwicklung der gesamten Software lag beim ZFT.

Der Verfasser war als Leiter einer kleinen Entwicklungsgruppe für die Funktionsfähigkeit der zentralen Baugruppe des MUX30A zuständig und hat in diesem Zusammenhang die Feinkonzeption der Hardware und das Steuerprogramm erarbeitet. Die folgende Ausarbeitung basiert zum großen Teil auf Erinnerungen.

Zu technischen und funktionellen Zusammenhängen sind noch umfangreiche Unterlagen – so z. B. eine komplette technische Beschreibung und Programmlisten der zentralen Software - vorhanden. Unsicherheit besteht teilweise hinsichtlich der genauen Datierung einzelner Ereignisse.

Erste Muster des EC8404.M1 wurden 1984 eingesetzt, die Fertigung erfolgte bis 1989 in Radeberg. Es wurden jährlich ca. 60 Systeme in unterschiedlichen Ausbauvarianten ausgeliefert.

2 Die technische Lösung des MUX30A

2.1 Hardwarestruktur

Der MUX30A ist ein modular projektierbares Multi-Mikrorechnersystem. Bei maximalem Ausbau können damit 128 Datenübertragungskanäle an einen ESER-Rechner – die Mainframe-Generation der RGW-Länder der 1970iger und 1980iger Jahre – angeschlossen werden. Die Kopplung erfolgt am Multiplexkanal des ESER-Rechners, dabei ist ein alternativer Betrieb an zwei Systemen möglich. Innerhalb der Funktionsgruppen ESAP und MUX20 wird jeweils ein Mikrorechnersystem K1520 eingesetzt.

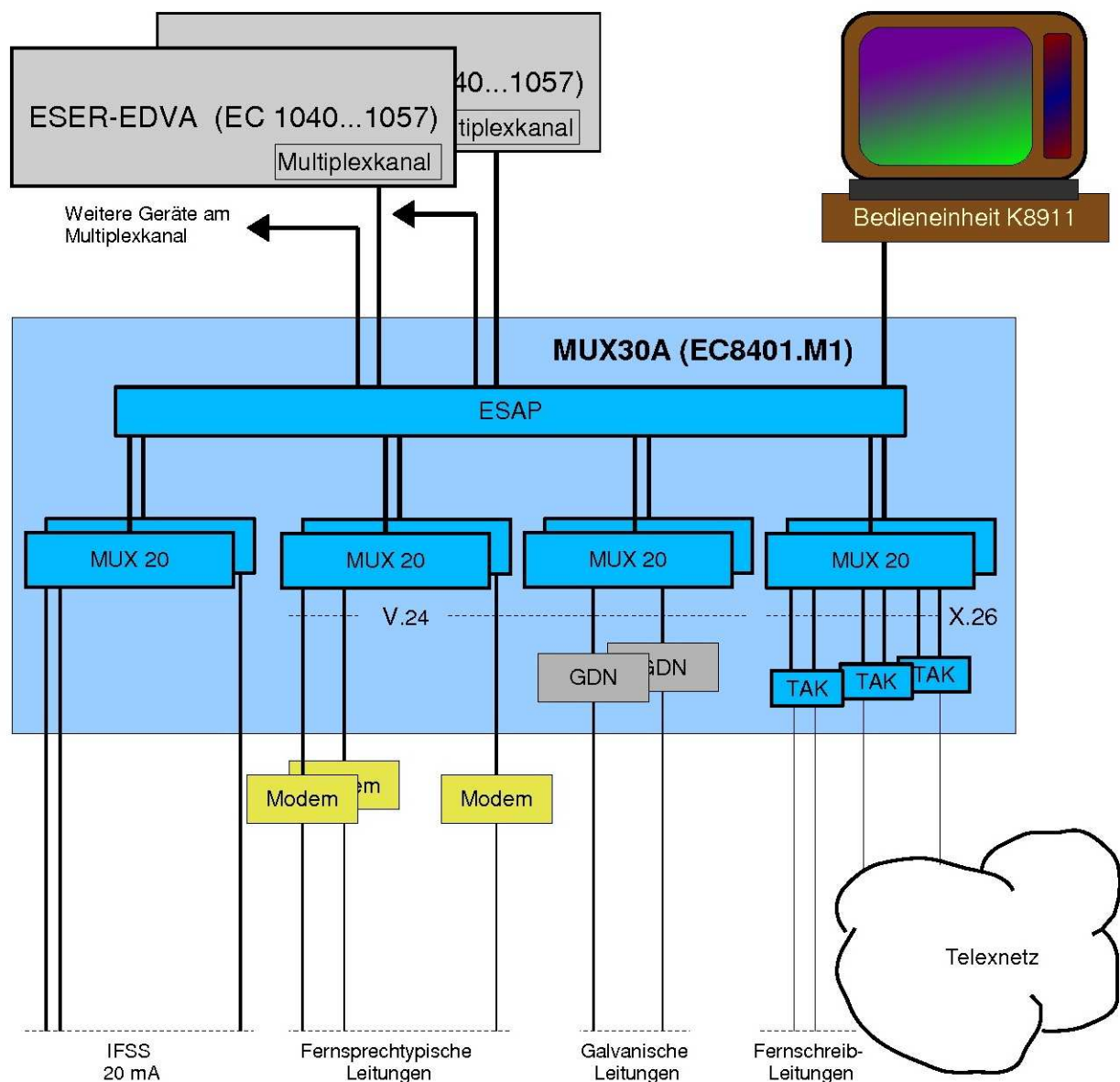


Abbildung 1: Funktionelle Struktur des MUX30A

Abbildung 1 zeigt die funktionelle Hardwarestruktur.

- Für die Ausbildung der seriellen Datenübertragungskанäle sind eine bis maximal acht Funktionsgruppen MUX20 zuständig mit jeweils maximal 16 Anschlüssen. Jeweils zwei MUX20 füllen einen Steuereinschub SE32 nach Abschnitt 2.2.2. Verfügbar sind hier die drei Interfacetypen:

CCITT V.24, CCITT X.26 und die Stromschleife 20 mA (IFFS)

- Im Fall der Ankopplung von Fernschreibleitungen sind Interfacewandler (TAK) zwischen dem Leitungsanschluss und dem X.26- Anschluss des MUX20 erforderlich. Zur Aufnahme dieser Module existieren Datenübertragungseinschübe nach Abschnitt 2.2.3 als Systembaugruppe. Bei Bedarf werden diese Datenübertragungseinschübe auch genutzt, um Einbau-GDN-Module einzusetzen. Einbau-Modem standen zur Einsatzzeit nicht zur Verfügung.
- Als Zentralsteuerung und zum Anschluss an den bzw. die zwei Zentralrechner dient der ESER-Anschluss-Prozessor ESAP, ein separater Einschub nach Abschnitt 2.2.4.
- Mit jedem MUX30A wurde ein Serviceterminal K8911 ausgeliefert.

2.2 Schranksystem und Blockeinschübe

2.2.1 Überblick

Das für den MUX30A verwendete Schranksystem wird auch bei der Rechnerlinie K1600 verwendet und ist nicht typisch für Geräte der Mainframe-Peripherie. Diese Bauform wurde festgelegt, weil die Funktionsgruppe MUX20 bereits vor der Verwendung im MUX30A in die Systeme K1620 und K1630 einbezogen worden ist. Bei den Systemen K1600 sind verschiedene Schrankhöhen möglich. Die Schrankhöhe beim MUX30A beträgt 1600 mm, damit kann jeder Schrank bis zu vier Blockeinschübe 6HE aufnehmen. In der Regel wurde jedes System MUX30A individuell für den Einsatzfall projektiert. Die Anzahl und Art der benötigten Datenleitungen bestimmten dabei die Anzahl der erforderlichen Blockeinschübe und damit die Anzahl der Schränke. Als Anpassung an die Gestaltung der ESER-Systeme wurden die Schränke des MUX30A blau/grau lackiert, während K1600-Systeme in braun/beiger Farbgebung ausgeliefert wurden.

Jeder Blockeinschub ist so gestaltet, dass er nach vorn aus dem Schrank herausgezogen werden kann. Module und Steckeinheiten werden dann von oben eingesteckt. Die Verkabelung erfolgt über Steckverbinder an der Oberseite der Steckeinheiten, dabei war besonders bei vollständig bestückten Systemen auf eine sorgfältige Verlegung der Kabel unter Nutzung spezieller Kabelführungen zwischen Schrank und Blockeinschub zu achten. Die folgenden Abbildungen zeigen die Verhältnisse.

Jeder Blockeinschub verfügt über eine eigene Stromversorgung einschließlich Überwachung, jeder Blockeinschub hat einen eigenen Netzschalter, dabei ist jedoch die zentrale Ein- und Ausschaltung gewährleistet. Eine Lüfterbaugruppe in jedem Blockeinschub zwischen den Netzteilen (gut erkennbar in Abbildung 5) und dem Steckeinheiteneinsatz sorgt für eine horizontale Durchströmung des Blockeinschubes. Zusätzliche Lüfter sind im Dach des Schrankes vorhanden und verhindern einen Stau der Kühlluft. Der Lufteintritt erfolgt über einen Filter unten in der Front des Schrankes.

Die Verkabelung von Stromversorgung, Hostinterface und Datenkanälen erfolgte in der Regel von unten unter Nutzung doppelter Fußböden.

In den Funktionsgruppen MUX20 und ESAP sind Mikrorechnerkerne K1520 enthalten, die Steckeinheiten dieses Systems mit 215 * 170 mm (B*H) und zwei zwei-reihigen Steckverbindern mit je 58 Kontakten waren Vorgabe für die konstruktive Gestaltung aller Ergänzungsentwicklungen.

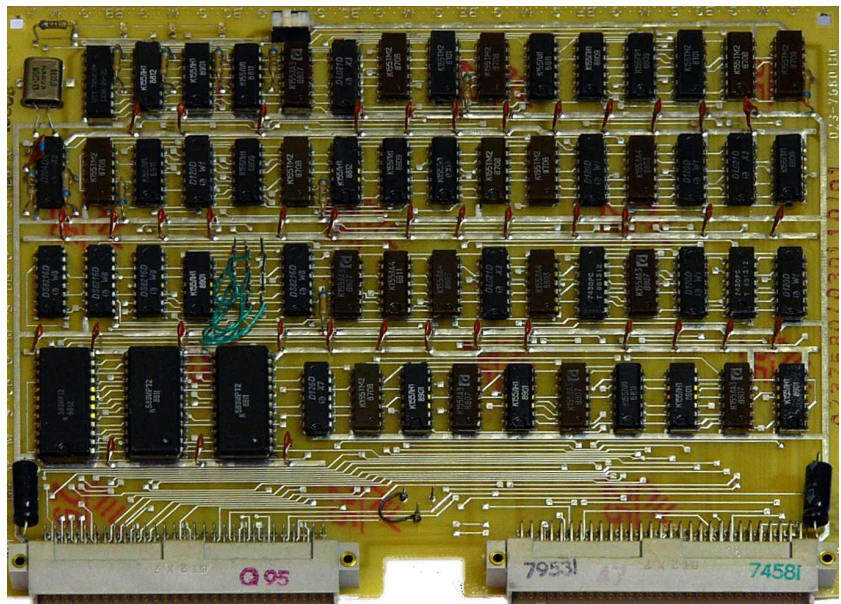


Abbildung 2: Typische Steckeinheit

2.2.2 Steuereinschub SE32 mit MUX20

Ein MUX20 ist eine Funktionsgruppe zur Realisierung von bis zu 16 seriellen Datenleitungen. Zwei MUX20 mit gemeinsamer Stromversorgung sind in einem Steuereinschub SE32 zusammengefasst. Dazu wird ein Steckeinheiteneinsatz mit 21 Plätzen verwendet, dabei ist die Rückverdrahtungsleiterplatte so gestaltet, dass zwei unabhängige Bussysteme (Platz 1...11 und Platz 12...21) vorhanden sind. Abbildung 2 zeigt die Verhältnisse. Der Einschub ist teilweise herausgezogen und damit ist der vordere MUX20 sichtbar. Für Anwendungen im Zusammenhang mit K1600-Systemen wurde ein Einschub mit einem MUX20 und der Bezeichnung SE16 genutzt.

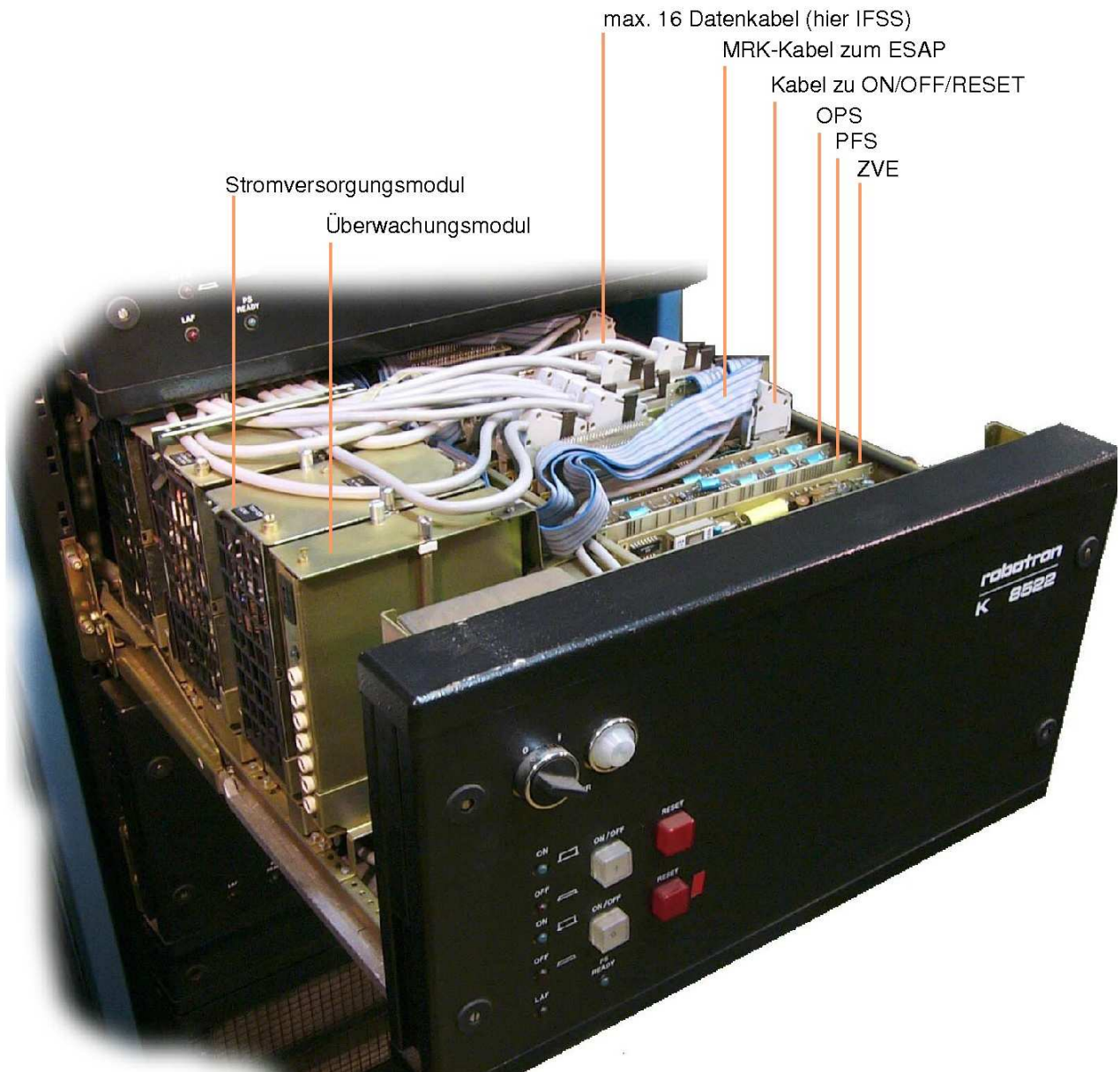


Abbildung 3: Steuereinschub SE32 mit zwei MUX20

Der **Mikrorechnerkern** jedes MUX20 besteht aus den Steckeinheiten (Plätze 2...5 und 12...15):

- ZRE K2521.00 3K Byte EPROM (3* 2716, gesteckt)
- PFS K3822.00 16K Byte EPROM (16* 2716, eingelötet)
- OPS K3525.00 16K Byte dynamisch RAM (zweimal vorhanden)

Die seriellen Kanäle werden durch Serielle Datenadapter-Steckeinheiten (SDA) bereitgestellt. Eine **SDA** hat bei Vollbestückung vier und als abgerüstete Varianten zwei gleichartige Kanäle. Interfacetypen sind V.24/V.28, X.20/X.26-FS und IFSS. Jeweils vier der folgenden Steckeinheiten können in einem MUX20 (Plätze 8...11 und 18...21) zum Einsatz kommen:

- ASV K8021.22 SDA V.24/V28 asynchron/synchron 4 Kanäle
- ASV K8021.32 SDA V.24/V28 asynchron/synchron 2 Kanäle
- ASX K8023.22 SDA X.20/X.26 asynchron 4 Kanäle
- ASX K8023.32 SDA X.20/X.26 asynchron 2 Kanäle
- ASS K8025.22 SDA IFSS 4 Kanäle
- ASS K8025.23 SDA IFSS 2 Kanäle

Zur Kopplung mit der Zentralbaugruppe (im MUX30A dem ESAP) ist die BUS-Koppel-Einheit **BKE15** vorhanden. Diese besteht aus zwei Steckeinheiten und einer griffseitig aufgesteckten Verbindungsleiterplatte.

- die Steckeinheit KE15R (Platz 6 oder 16) ist gekennzeichnet durch einen 10-polige Steckverbinder X4, an diesen erfolgt der Anschluss eines Kabels zur ON / OFF / RESET-Baugruppe an der Fronteinheit des Blockeinschubes.
- die Steckeinheit KE15K (Platz 7 oder 17) hat einen 58-polige Steckverbinder X4, an diesen erfolgt der Anschluss eines Litzenbandkabels zum MRK-BUS des ESAP.
- beide Steckeinheiten haben einen griffseitigen 58-polige Steckverbinder X3, die durch eine Verbindungsleiterplatte Pin auf Pin verbunden werden.

Es sei vermerkt, dass die angegebenen Platznummern der Steckeinheiten im Steckeinheiten-einsatz für die Funktion keine Bedeutung hatten.

Hinsichtlich der Verbindung der Datenübertragungskanäle mit dem weiteren System gilt:

- im Falle von IFSS-Kanäle ist in der Rückseite des Schrankes hinter dem betreffenden Steuereinschub ein Anschlussfeld (mit Trennelementen) montiert. Für jeden IFSS-Kanal führt von diesem ein Anschlusskabel zum Anschlussstecker der ASS. Am Anschlussfeld erfolgt dann der Anschluss der Vierdraht-Datenleitung durch Löten.
- X.20-Kanäle führen immer zu einer TAK oder TAK-R in einem Datenübertragungseinschub. Hierfür werden spezielle Verbindungskabel benutzt. Diese existieren in verschiedenen Längen in Abhängigkeit davon, ob der Dateneinschub benachbart oder weiter entfernt ist.
- V.24-Verbindungen zu Einbau-GDN im Datenübertragungseinschub benutzen ebenfalls systemeigene Verbindungskabel mit definierten Längen.
- V.24-Verbindungen zu externen Datenübertragungseinrichtungen (Modem und GDN) erfordern spezielle Kabel, weil auf der ASV-Steckeinheit nicht der allgemein verbindliche 25-polige Steckverbinder verwendet wird.

2.2.3 Datenübertragungseinschübe DUE16 und DUE 32

Die Datenübertragungseinschübe DUE16 und DUE32 sind projektabhängig eingesetzt worden:

- grundsätzlich bei Projekten mit Nutzung von Telegrafiekkanälen, hier wurde der Einsatz der Leiterkarten TAK und ggf. TAK-R (siehe unten) notwendig und dieser konnte nur innerhalb eines DUE16 oder DUE32 erfolgen. Ein DUE16 nimmt 8 TAK und ein DUE32 bis zu 16 TAK auf, Mischbestückungen mit GDN (s.u.) sind möglich. Eine TAK realisiert zwei Kanäle.
- optional bei Projekten mit Nutzung von GDN-Übertragungen, hier konnte auf Einbau-GDN-Module zurückgegriffen werden, was eine kompakte Systemgestaltung ermöglichte. In jedem der genannten Datenübertragungseinschübe konnten bis zu 10 GDN eingesetzt werden.

Grundsätzlich entspricht der Aufbau eines Datenübertragungseinschubes dem im vorigen Abschnitt dargestellten Steuereinschub. Die TAK-Leiterkarten und/oder GDN-Module werden von einem Steckeinheiteneinsatz mit 21 Plätzen aufgenommen und über dessen Rückverdrahtungsleiterplatte mit den jeweiligen Betriebsspannungen versorgt. Die folgende Abbildung zeigt einen mit mindestens 9 GDN-Modulen bestückten Einschub mit Blick auf den Steckeinheiteneinsatz.

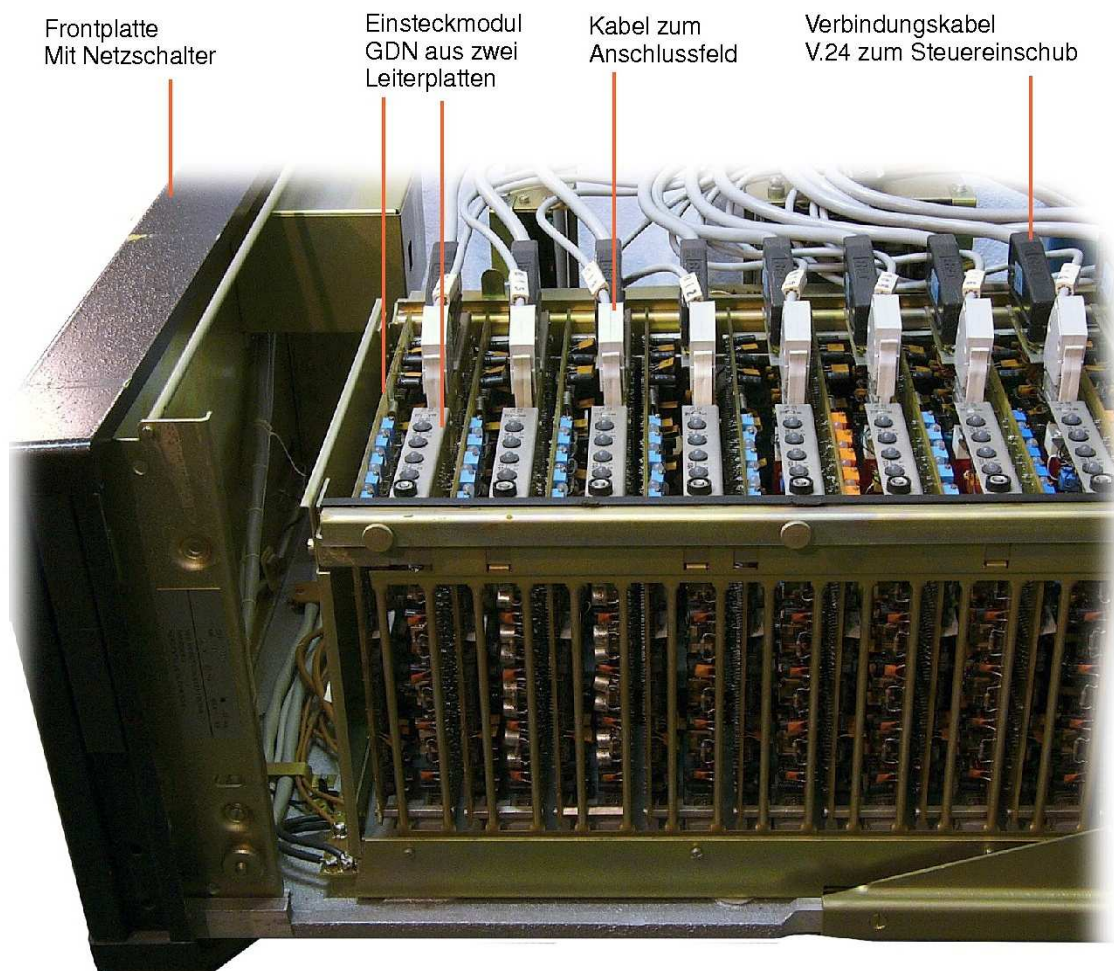


Abbildung 4: Seitenansicht Datenübertragungseinschub mit GDN-Modulen

Der Datenübertragungseinschub beinhaltet eine entsprechende Stromversorgung aus mehreren Modulen und einer Überwachungsbaugruppe. Hauptfunktion des Datenübertragungseinschubes neben der konstruktiven Aufnahme der TAK- und GDN-Baugruppen ist deren Versorgung mit den entsprechenden Niederspannungen. Art und Anzahl der Stromversorgungsmodule sind wiederum projektabhängig, denn TAK und GDN erfordern unterschiedliche Einspeisungen.

- bei Bestückung des Datenübertragungseinschubes mit bis zu drei GDN-Modulen genügt ein STV-Modul 5V/10A (K0361).
- bei Bestückung mit 4 bis 7 GDN-Modulen wird ein STV-Modul 5V/20A (K0362) benötigt.
- bei Bestückung mit 8 bis 10 GDN-Modulen ist ein STV-Modul 5V/30A (K0363) erforderlich.
- Im Fall einer Bestückung mit TAK müssen zusätzlich zum STV-Modul 5V (K038x) ein Stromversorgungs-Zusatzmodul STZ (K0367) und zwei STV-Module 36V/1,4A (K0361.16) eingesetzt werden.

Die nach außen gerichteten Anschlusskabel von den TAK oder GDN werden über die Kabelführung des Einschubes zu einem Trennelemente- und Anschlussfeld geführt, das in der Rückseite des Schrankes hinter dem betreffenden Einschub montiert ist. Hier erfolgt der Anschluss der Vierdraht-Datenleitung durch Löten.

Die Steckereinheit **TAK-R** wurde bei Projekten mit Nutzung des Telexnetzes eingesetzt. Mithilfe der TAK-R kann die Umschaltung eines Fernschreibkanals auf einen Fernschreiber erfolgen.

2.2.4 ESER-Anschluss-Prozessor

Zentralsteuerung des MUX30A und Anschluss zum jeweiligen Hostrechner ist der sogenannte **ESAP** (ESER-Anschluss-Prozessor). Diese Funktionseinheit füllt einen eigenen Einschub, den Bus-Koppel-Prozessor BKP K4501 aus. Zusätzlich ist ein Steckeinheiteneinsatz im Sockelbereich des Schrankes vorhanden, dieser enthält spezielle Kabelstufen-Steckeinheiten und die Interfacestecker für zwei Hostrechner-Interfaces. Die folgende Abbildung zeigt den Aufbau dieses Einschubes.

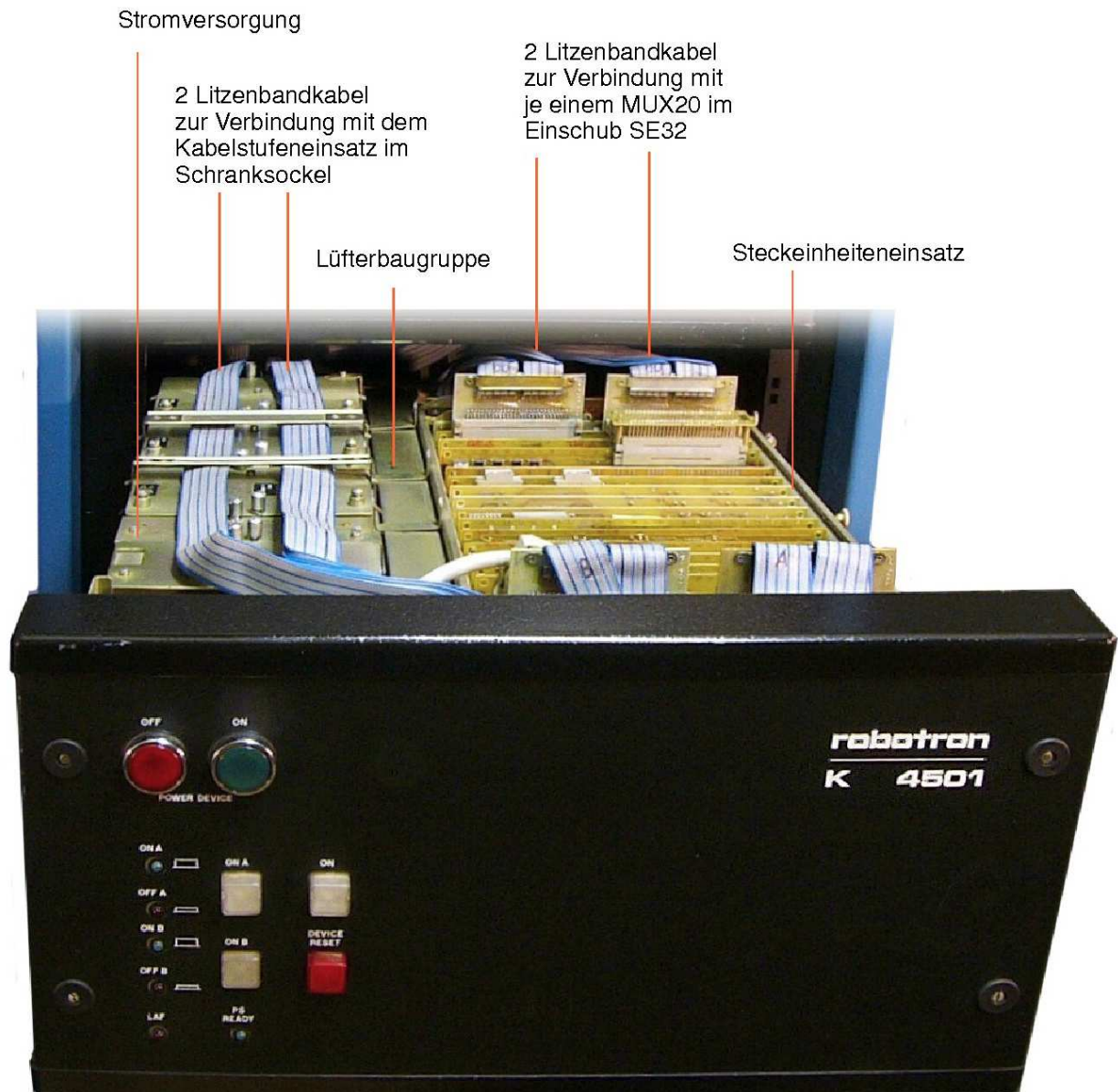


Abbildung 5: Bus-Koppel-Prozessor Einschub ESAP

Am Frontpaneel befinden sich Tasten zum Ein- und Ausschalten des gesamten Gerätes sowie Tasten zur On/Off-Schaltung der Hostinterfaces. Die angewählten Zustände werden im Ergebnis von Aktionen des internen Steuerprogramms erreicht und dann mittels LED angezeigt.

In seinem grundsätzlichen Aufbau entspricht dieser Einschub den bereits beschriebenen Einschüben. Den Boden des Steckeinheiteneinsatzes bildet eine spezielle Rückverdrahtungsleiterplatte mit drei funktionellen Bereichen:

- die Plätze 1... 9 realisieren mit jeweils zwei Steckverbindern X1 und X2 die interne Verdrahtung der Funktionsgruppe **EKST**. Zum Teil sind hier dreireihige 87-polige Steckverbinder eingesetzt und es existiert eine zusätzliche Wickelverdrahtung.
- die Plätze 10...16 realisieren mit jeweils zwei Steckverbindern X1 und X2 einen modifizierten **Systembus K1520** zur Aufnahme von Systemsteckeinheiten. Die Busstruktur wird auch an die Plätze 7...9 herangeführt.
- die Plätze 17...21 realisieren mit einem Steckverbinder X1 den **MRK-Bus** (siehe Abschn. 2.3.4). Dieser Bereich hat keine logische Verbindung zu den anderen Bereichen.

Die Funktionsgruppe EKST umfasst die neun unterschiedlichen Steckeinheiten EKST1, EKST2, EKST3.1, EKST3.2, EKST4 bis EKST7, die in dieser Reihenfolge auf die Plätze 1 bis 9 gesteckt sind. Die EKST1 hat griffseitig zwei 58-polige Steckverbinder, diese dienen zur Verbindung mit dem Kabelstufeneinsatz, jeweils für einen Hostkanal. Die EKST2 hat griffseitig einen 10-poligen Steckverbinder zur Verbindung mit den Tasten und Anzeige-LED des Frontpaneels.

Der **Mikrorechnerkern** des ESAP besteht aus den Steckeinheiten:

- ZRE K252x.00 3K Byte EPROM (3* 2716, gesteckt), x = 1 oder 2
- PFS K3822.02 16K Byte EPROM (16* 2716, eingelötet)
- OPS K3520.00 4K Byte statisch RAM (in der Regel zweimal vorhanden)

Eine Steckeinheit **SDA ASS K8025.23** (siehe Abschnitt 2.2.2) ermöglicht den Anschluss des Serviceterminals über eine IFSS-Verbindung.

Die Ankopplung der einzelnen MUX20, d.h. deren Buskoppereinheiten BKE15, erfolgt an den MRK-Bus über Koppelbusverteilerkarten KVK K0501.32. Eine KVK hat griffseitig zwei 58-polige Steckverbinder und ermöglicht so den Anschluss von zwei MUX20 über Flachbandkabel. Bis zu vier KVK können auf den Plätzen 18 bis 21 gesteckt sein.

Die Ankopplung des ESAP an den MRK-Bus erfolgt mithilfe der Funktionsgruppe **BKEM15**, dabei steht das M für die Masterfunktionalität. Sie besteht aus zwei Steckeinheiten und einer griffseitig aufgesteckten Verbindungsleiterplatte.

- die Steckeinheit BKEM15R ist dem K1520-Bus (Platz 16) zugeordnet.
- die Steckeinheit BKEM15K ist dem MRK-Bus (Platz 17) zugeordnet.

Beide Steckeinheiten haben je einen griffseitigen 58-poligen Steckverbinder X3, diese werden durch eine Verbindungsleiterplatte Pin auf Pin verbunden. Damit sind die o.g. festen Plätze festgelegt.

2.3 Funktionsgruppen

2.3.1 Allgemeines

In funktioneller Hinsicht besteht der MUX30A aus einer Vielzahl von Funktionsgruppen. In Abbildung 6 ist deren Zusammenschaltung und Einordnung dargestellt.

2.3.2 Mikrorechnerkerne K1520

Die Mikrorechnerkerne von MUX20 und ESAP sind unter Verwendung von K1520-OEM-Steckeinheiten der ersten Generation ausgeführt.

- als Zentraleinheit wird die Steckeinheit **ZRE K2521** eingesetzt, alternativ kann im ESAP auch K2522 (ohne Taktgenerator) eingesetzt werden. Basis dieses Systems sind die zum System Z80 kompatiblen Schaltkreise UA880D (ZE), UA855D (PIO) und UA857 (CTC) mit einer Taktfrequenz von ca. 2,5 MHz. Auf dieser Steckeinheit sind auch drei 1KByte-EPROM und ein 1KByte RAM-Bereich vorhanden.
- 1KByte-EPROM-Schaltkreise bilden auch die technische Basis der Festwertspeicher-Steckeinheit **PFS K3822**. Bei dieser Steckeinheit sind die 1 KByte-EPROMs eingelötet, die Programmierung erfolgt auf der Steckeinheit unter Nutzung von zusätzlichen Steckverbinderkontakten auf der Griffseite. Programminhalte waren:
 - in den MUX20 Laderoutine LAD und Testprogramm MKOF.
 - im ESAP das ESAP-Steuerprogramm.
- bei der Operativspeicher-Steckeinheit **OPS K3520** werden als Speicherschaltkreise statische 1K*1bit-Schaltkreise U202 verwendet. Diese Steckeinheit wird ein- oder zweimal im ESAP eingesetzt:
 - mit Adressumfang 1000H...1FFFFH für Arbeitsbereiche des Steuerprogramms und der EKST
 - optional mit dem Adressumfang 2000H...2FFFFH als Protokollbereich für mögliche Systemauswertungen.
- bei der Operativspeicher-Steckeinheit **OPS K3525** werden als Speicherschaltkreise dynamische 4 K*1bit-Schaltkreise benutzt. Diese Steckeinheit wird zweimal je MUX20 eingesetzt, dabei wird ein zusammenhängender Adressbereich 1000H...9FFFFH eingestellt. In diesen RAM-Bereich arbeitet das MUX20-Steuerprogramm **EMLMUX**.

Zum Zeitpunkt der Produktionseinführung des MUX30A standen modernisierte Produkte zur Verfügung, aus verschiedenen Gründen unterblieben jedoch entsprechende Ablösungen.

2.3.3 Serieller Datenadapter SDA

Die Seriellen Datenadapter SDA sind K1520-kompatible Steckeinheiten. Die Entwicklung erfolgte im Entwicklungsbereich Radeberg. Eine **SDA** hat bei Vollbestückung vier und als abgerüstete Varianten zwei gleichartige Kanäle. Interfacetypen sind V.24/V.28, X.20/X.26-FS und IFSS (20 mA-Stromschleife). Kern der Steckeinheiten sind die Mikrorechnerschaltkreise **UA856D** (kompatibel zum Z80-SIO) in Verbindung mit dem UA857D (CTC). Ein Schaltkreis SIO stellt zwei serielle Kanäle bereit, für die Takterzeugung ist jedem SIO- ein CTC-Schaltkreis zugeordnet. Für Zeitgeberfunktionen ist ein weiterer CTC- Schaltkreis vorhanden. Zur Busseite hin werden die I/O-Adresse (in Bereichen zu 20H) und die Interrupteigenschaften mittels Schaltern bzw. Wickelbrücken festgelegt.

Zur Leitungsseite hin unterscheiden sich die drei Steckeinheitentypen grundlegend:

- die **ASV SDA V.24/V28** stellt bei Vollbestückung vier Interfaces an 26-poligen EFS-Steckverbindern bereit. Die maximale Datenrate ist 9600 bps. Die Betriebsweise kann asynchron oder synchron festgelegt werden, dies wird für jeden Kanal einzeln über DIL-Schalter eingestellt.
- die **ASX SDA X.20/X.26** stellt bei Vollbestückung vier Interfaces an 10-poligen EFS-Steckverbindern bereit. Die Datenrate ist mit maximal 300 bps durch die verwendeten Kopplungen festgelegt. Die Betriebsweise ist asynchron, für jeden Kanal ist die Leitungsart (z.B. Stand/Wählleitung) per DIL-Schalter einzustellen.
- die **ASS SDA IFSS** stellt bei Vollbestückung vier Interfaces an 10-poligen EFS-Steckverbindern bereit. In den Kabelstufen sind Optokoppler eingesetzt. Die Datenrate beträgt maximal 9600 bps. Die Betriebsweise ist asynchron, für jeden Kanal kann die Art der Einspeisung des Schleifenstromes (aktiv/passiv) per DIL-Schalter eingestellt werden.

Eine Besonderheit aller drei Varianten stellt der sogenannte **Wartungsmodus** dar. Dazu wird für Testzwecke pogrammtechnisch eine Rückführung das Sendesignal eines Kanals auf dessen Empfangseingang geschaltet und es können die Komplexe Busanpassung und SIO/CTC überprüft werden.

2.3.4 Mehrrechnerkoppelbus MRKB

Der Mehrrechnerkoppelbus MRKB stellt einen Systemknoten zur Zusammenschaltung verschiedener (Mikro-) Rechnersysteme innerhalb eines komplexen Gerätes dar. Die Definition dieses Bussystems und die Entwicklung aller dafür erforderlichen Komponenten erfolgte im Entwicklungsbereich Radeberg. Am MRKB können laut Definition maximal 16 Teilnehmersysteme angeschlossen werden, wobei jedes System dazu eine Buskoppereinheit enthalten muss. Das System des MRKB wird sowohl im MUX30A als auch in Systemen K1600 verwendet. Funktionell ist dieses System so ausgelegt, dass einzelne Buskoppereinheiten in einen Offlinezustand versetzt werden können und dass in diesem Zustand umfassende Servicehandlungen möglich sind. Im MUX30A wird mit einer Datenwortbreite von 8 Bit gearbeitet, die Definition des MRKB lässt auch eine Datenwortbreite von 16 Bit zu.

Realisiert wird der MRKB durch einen zum K1520-Bus kompatiblen separaten Abschnitt eines Steckeinheiteneinsatzes, der mit Koppelbus-Verteiler-Karten **KVK** bestückt ist. Eine KVK hat griffseitig zwei 58-polige Steckverbinder und ermöglicht so den Anschluss von zwei Teilnehmer-Systemen über Flachbandkabel. Dieses Interface wird als ZK-Bus bezeichnet.

2.3.5 Buskoppereinheit BKE 15

Die Bus-Koppel-Einheit **BKE15** ist ein Teil der Funktionsgruppe MUX20 und dient dem Anschluss des MUX20 (über eine KVK-Steckeinheit) an den **Mehrrechnerkoppelbus** (MRKB). Die BKE15 ist Slaveeinheit im Sinne der MRK-Bus-Definition. Ein Datenaustausch kann immer nur mit einer Mastereinheit erfolgen, es findet also kein Transfer zwischen verschiedenen MUX20 innerhalb des MUX30A statt.

Ein internes Datenregister mit einer Kapazität von 256 Byte ermöglicht eine weitgehende Entkopplung der Zugriffe der gekoppelten Mikrorechner. Ein zusätzlicher Performancegewinn wird dadurch erreicht, dass der Datenaustausch zwischen MUX20-Speicher und diesem Datenregister per DMA-Steuerung erfolgen kann.

Bezüglich des MRK-Bus wird durch die BKE15 eine ON/OFF-Steuerung ermöglicht. Ein OFF-Gesuch (durch Setzen der Taste) führt per Software zum Zustand OFF und nur im Zustand OFF ist ein Rücksetzen des MUX20 möglich. Grundsätzlich ist es zulässig, einen MUX20 bei weiterhin arbeitenden MUX30A stromnetzseitig auszuschalten (dies sollte im Zustand OFF erfolgen) und die Verbindung zur **KVK-Steckeinheit** abzuziehen und wieder anzustecken. Damit können Wartungs- oder Reparaturarbeiten am laufenden System ausgeführt werden.

Eine BKE15 besteht aus zwei Steckeinheiten und einer griffseitig aufgesteckten Verbindungsleiterplatte:

- die Steckeinheit BKE15R ist gekennzeichnet durch einen 10-polige Steckverbinder X4, an diesen erfolgt der Anschluss eines Kabels zur ON / OFF / RESET-Baugruppe an der Fronteinheit des Blockeinschubes.
- die Steckeinheit BKE15K hat einen 58-polige Steckverbinder X4, an diesen erfolgt der Anschluss eines Litzenbandkabels zur KVK-Steckeinheit des MRK-Buses.
- beide Steckeinheiten haben einen griffseitigen 58-polige Steckverbinder X3, die durch eine Verbindungsleiterplatte Pin auf Pin verbunden werden.

2.3.6 Buskoppereinheit BKEM 15

Die Funktionsgruppe **BKEM15** ist Bestandteil des ESER-Anschluss-Prozessors ESAP und ist die Mastereinheit im Sinne der MRK-Bus-Definition. Ihre Funktion besteht in der Überwachung des gesamten MRK-Buses und dem Datenaustausch mit der jeweils ausgewählten BKE15. Bei einem Datenaustausch wird ein Datensatz von bis zu 256 Byte entsprechend der Größe des internen Datenregisters der BKE15 übertragen. Es können bis zu acht BKE15 am MRK-Bus verwaltet werden, diese werden dann im Multiplexbetrieb bedient.

Bezogen auf den ESAP hat die BKEM15 zwei unterschiedliche Datenwege:

- den Systembus des internen K1520-Mikroprozessors, hier werden hauptsächlich Daten zur Systemsteuerung und Testung übertragen.
- einen Direkten Datenkanal zur Funktionsgruppe EKST, auf diesem erfolgt der eigentliche Datenaustausch mit dem Hostrechner im Anwendungsbetrieb.

Die BKEM15 verfügt über keinen Pufferspeicher und es erfolgt auch keine Zwischenspeicherung von Hostrechnerdaten im RAM-Speicher des ESAP-Mikroprozessors. Abbildung 6 macht die Zusammenschaltung der verschiedenen Bussysteme und den Datenfluss im MUX30A deutlich.

Eine BKEM15 besteht aus zwei Steckeinheiten.

- die Steckeinheit BKEM15R ist dem K1520-Bus (Platz 16) zugeordnet, Busseitig hat diese Steckeinheit zwei 58-polige Steckverbinder X1 und X2.
- die Steckeinheit BKEM15K ist dem MRK-Bus (Platz 17) zugeordnet und erfüllt damit auch die Funktionalität der KVK-Steckeinheit und hat wie diese Busseitig nur einen 58-poligen Steckverbinder X1.
- beide Steckeinheiten haben einen griffseitigen 58-poligen Steckverbinder X3, die durch eine Verbindungsleiterplatte Pin auf Pin verbunden werden.

2.3.7 ESER-Kanal-Steuerung EKST

Die ESER-Kanal-Steuerung EKST ist ein relativ umfangreicher Logikkomplex, der auf neun EKST-Steckeinheiten aufgeteilt ist. Hostseitig ist jeweils eines der zwei möglichen Interfaces vom Typ **Bytemultiplexkanal** mit bis zu 128 Subkanälen zu bedienen. Die Anforderungen hinsichtlich hoher Durchsatzserwartungen einerseits und die technischen Daten der verfügbaren Mikrorechner andererseits erforderten spezielle Lösungen und damit den o.g. hohen Aufwand:

- zunächst gilt das Prinzip, dass die EKST weitgehend autonom (d.h. ohne Mitwirkung des K1520-Mikrorechners) auf die Anforderungen des Hostrechners reagiert, indem die einzelnen Steuerfolgen eigenständig von der EKST ausgeführt werden. Dazu ist jedem der 128 Subkanäle ein Satz von 7 Arbeitszellen innerhalb des RAM-Speichers zugeordnet. Auf 5 dieser 7 Arbeitszellen kann die EKST per **DMA-Modus** innerhalb einer Steuerfolge zugreifen und diese Informationen verwenden bzw. modifizieren.
- startet der Hostrechner ein Kanalprogramm, so wird durch die EKST das Kommando per DMA übernommen und es wird ebenfalls per DMA eine Ereignisanzeige in Form einer Fifo-Warteschlange in einem RAM-Bereich gesetzt, die dann zeitversetzt vom Steuerprogramm ausgewertet wird.
- um die gestarteten Kanalprogramme auszuführen und/oder zu beenden, wird die EKST durch Befehle des Mikrorechners zur Anforderung von Zuschaltfolgen auf dem Hostinterface veranlasst, die entsprechenden Daten- oder Endebehandlungsfolgen werden dann autonom durch die EKST abgearbeitet.
- der Datenaustausch auf dem Hostinterface erfolgt in der **Betriebsweise Multibyte**, d.h. bei jeder Datenfolge wird eine Gruppe von Bytes übertragen. Diese Übertragung erfolgt ohne Beteiligung des Mikrorechners direkt über die BKEM15 hinweg vom bzw. zum Datenbuffer der BKE15 im MUX20. Die Anzahl der Byte, innerhalb einer Gruppe für Schreibkommandos, wird mittels DIL-Schalter auf der Steckeinheit EKST5 festgelegt. Bei Lesekommandos bestimmt die Software im MUX20 diese Byteanzahl.
- bei der Abarbeitung eines Schreibkommandos im Multibyte-Modus tritt das Problem auf, dass beim Vorkommen bestimmter Steuerzeichen im Ausgabedatenstrom weitere Anforderungen unterbleiben müssen und danach eine Endebehandlung für das Kommando einzuleiten ist. Innerhalb der EKST gibt es deshalb eine **Steuerzeichenerkennung**, es können zwei wahlfreie Zeichen vom Mikrorechnerprogramm vorgegeben werden.

Aus Abbildung 6 wird die strukturelle Einordnung der EKST in den Komplex des MUX30A deutlich. Die internen Abläufe der EKST werden durch mehrere Zähler gesteuert, dazu ist eine eigene Takterzeugung vorhanden, von welcher auch die Taktversorgung der ZRE erfolgt. Im Bereich von EKST, BKEM15 und K1520-Kern gibt es eine aufwändige Rückverdrahtungsleiterplatte, diese realisiert neben dem K1520-Sytembus auch die interne Verschaltung der EKST-Steckeinheiten untereinander (hier sind teilweise dreireihige 86-polige Steckverbinder eingesetzt) und den Direkten Datenkanal zwischen EKST und BKEM15.

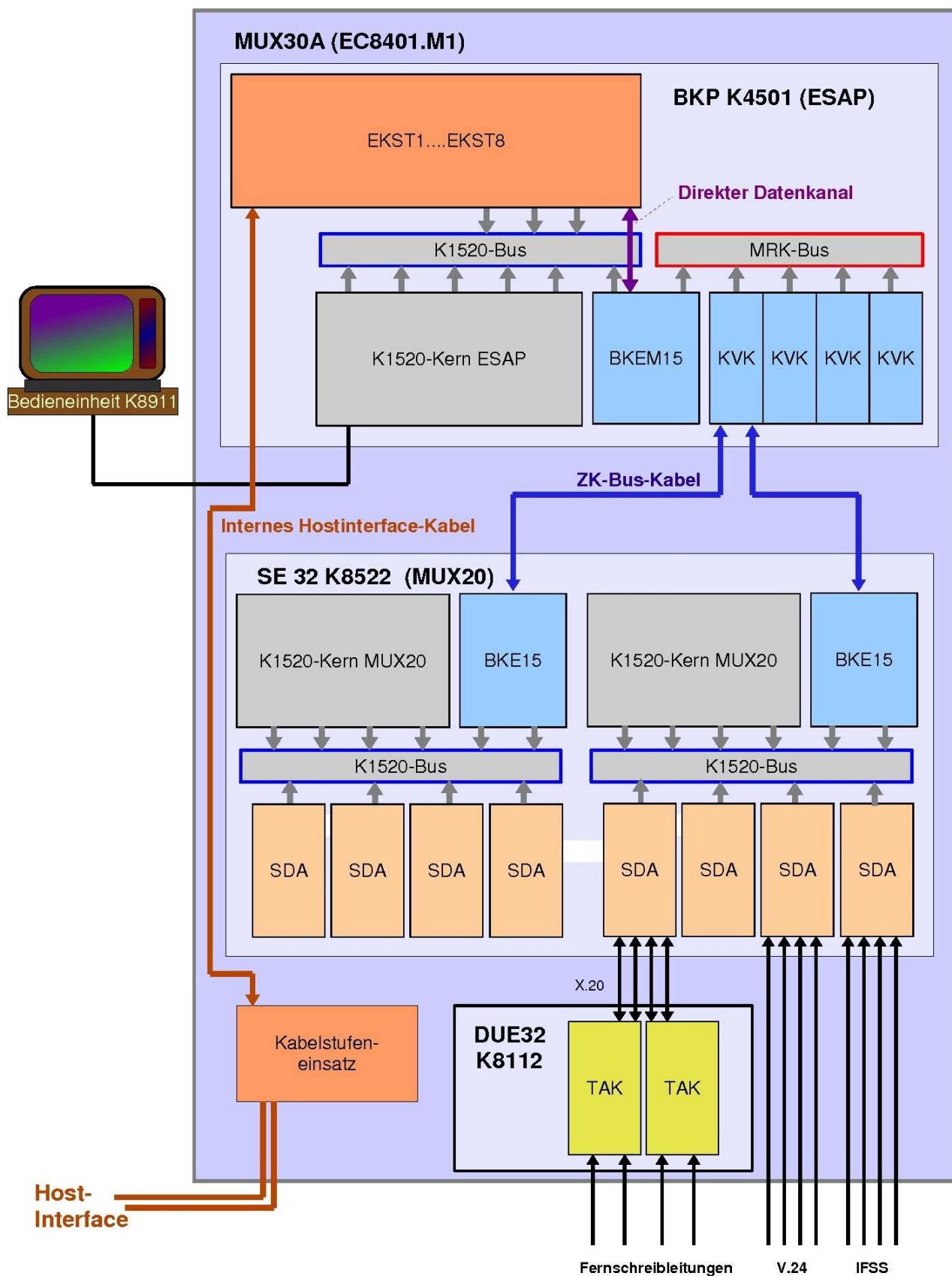


Abbildung 6: Funktionsgruppen-Übersicht MUX30A

2.4 Software

2.4.1 Allgemeines

Die Funktionen des MUX30A wird durch eine modular gestaltete Software in den Mikrorechnerkernen von ESAP und MUX20 gewährleistet. Diese Softwaremodule sind Firmware. Der Anwender kann lediglich in Form einer Parametrierung diese Firmware beeinflussen, es ist aber zu keiner Zeit vorgesehen gewesen, dass der Anwender hier eigene Software einsetzt.

Aus Gründen der Kompatibilität zu den Betriebssystemen der Hostrechner muss sich ein MUX30A funktionell wie ein Hardware-Multiplexer EC8401 – vergleichbar mit IBM2702 – verhalten. Eine derartige Emulation war Jahre zuvor mit dem MPD4 (EC8404) auf der technischen Basis des Rechners R4201 bei Robotron erfolgreich im Einsatz [1].

Der MPD4 hat nur eine funktionelle Ebene und bedient damit maximal 12 Datenleitungen, beim MUX30A sind bis zu 128 Datenleitungen zu bedienen und es existieren mit ESAP und MUX20 zwei funktionelle Ebenen. Die Leistungen der jeweiligen Rechner sind etwa vergleichbar.

Zielstellungen bei der Systemgestaltung waren:

- die funktionell exakte Emulation des Zielgerätes MPD4
- es sollte ein Systemdurchsatz mit mindestens 2400 bps je Leitung auch bei Vollausbau mit 128 Datenleitungen erreicht werden.
- es musste möglich sein, an Abschnitten der Datenfernverarbeitungsconfiguration offline zu arbeiten, während der Rest des Systems normal online weiter betrieben wird.

Hinsichtlich der technischen Realisierung hat es kein Vorbild gegeben, es war lediglich die o.g. Zielstellung umzusetzen, dabei galt es, eine sinnvolle und möglichst optimale Definition der internen Schnittstellen zu finden. Die eigentliche Emulation findet dabei in den MUX20 statt und das dafür erforderliche Emulationsprogramm wird vom Hostrechner in jeden MUX20 geladen. Dem ESAP obliegt die Verwaltung der Subsysteme MUX20 und die Ausführung und Überwachung der Interface-Aktionen auf dem Hostinterface.

Um den Systemanlauf abzusichern und um qualitativ hochwertige Testoptionen für die umfassende Überprüfung des MUX30A und der daran angeschlossenen Datenübertragungsstrecken und Terminals bereitzustellen, sind in jedem MUX20 Programmmodule auf EPROM vorhanden.

Die Entwicklungen der Softwaremodule erfolgten in verschiedenen Teams. Dabei ist es gelungen, ein in sich stimmiges System zu schaffen, das mit der oben beschriebenen Hardware eine Nachbildung der **IBM 3705 im Emulationsmodus** realisiert. Es sei vermerkt, dass die Original-Betriebssysteme OS, DOS und SVM von IBM mit ihren DFV-Zugriffsmethoden BTAM und TCAM mit diesem Gerät lauffähig waren. Auf der Basis der äquivalenten Betriebssysteme des ESER und der Mainframes EC1055 und EC1057 wurden dann DFV-Lösungen entwickelt. Genannt seien hier Systeme für die Geldwirtschaft und ein Platzbuchungssystem für die Deutsche Reichsbahn der DDR.

2.4.2 ESAP-Steuerprogramm

Das Steuerprogramm des ESAP führt in allen Betriebssituationen des MUX30A die erforderlichen Funktionen aus. Es ist vollständig im EPROM der Festwertspeicher-Steckeinheit gespeichert und wird nach Netzeinschalten oder nach Betätigen der Taste DEVICE RESET (bei gleichzeitig gesetzter Taste ON) gestartet. Die Hauptfunktionen des ESAP-Steuerprogramms sind:

- eine exakte Bedienung des Hostinterfaces für alle vorgesehenen Subkanäle unter allen Betriebsbedingungen einschließlich der Online- / Offlinesteuerung der Interfaces.
- die Initialisierung des gesamten Systems und Behandlung der Online-/ Offlinesteuerung der einzelnen MUX20.
- Ausgaben von System- und Fehlernachrichten an das Serviceterminal.
- Entgegennahme von Servicekommandos am Bedienterminal und Ausführung der entsprechenden Funktionen.

Für den normalen Anwenderbetrieb benötigt das ESAP-Steuerprogramm keine Bedienung am Serviceterminal. Beim Start des Programms erfolgt zunächst eine **Anlaufdiagnose**, diese überprüft den RAM-Speicher und den Zustand des Mehrrechnerkoppelbuses und ermittelt damit, welche MUX20 sich im Zustand ONLINE befinden. Bei positivem Testergebnis der Anlaufdiagnose wird ein Zustand BEREIT eingenommen.

Mit dem Erreichen des **Zustandes BEREIT** ist die später beschriebene **Servicefunktion** verfügbar. Im Zustand BEREIT wird das Initialisieren des ESAP erwartet, dieses erfolgt durch den Hostrechner, indem dieser ein spezielles Schreibkommando (81H) auf einer beliebigen Adresse des eingestellten Adressraumes ausführt. Mit diesem Kommando wird dem Steuerprogramm des ESAP eine Tabelle übergeben, welche die Zuordnung der einzelnen Subkanäle des Adressraumes zu den konkreten MUX20 beschreibt. Jedem Kanal eines MUX20 ist damit ein Subkanal (d.h. eine Adresse) des Hostrechners zugeordnet und damit wird der Zustand EMULATION erreicht. Der **Zustand EMULATION** stellt den Standardbetriebszustand dar. In diesem Zustand werden die Kanalkommandos vom Hostrechner zur weiteren Ausführung an den MUX20 weitergeleitet. Der MUX20 führt diese Kommandos aus, indem er (bei Schreibkommandos) Daten anfordert, (bei Lesekommandos) Daten übergibt und schließlich das Kommando durch die Übergabe eines Status- und Abfühlbytes an den ESAP beendet. Auf dem Hostinterface stellt sich dieser Ablauf als ein statistisch multiplexes Durcheinander von Steuerfolgen dar, auf dem MRK-Bus werden diese Steuerfolgen als Datensätze mit vereinbarten Strukturen abgebildet.

Um auch die MUX20 zu initialisieren, erfolgt per Schreibkommando vom Hostrechner über jeweils eine der vorher zugewiesenen Adressen das Laden und Starten eines Emulationsprogramms EMLMUX in jedem MUX20. Um diesen Ladevorgang auszuführen, ist nach dem Netzeinschalten oder RESET in jedem MUX20 eine auf EPROM abgelegte Laderoutine aktiv. Nach dem erfolgreichen Start seines EMLMUX meldet der MUX20 dies dem ESAP und erst danach werden dort die betreffenden Subkanäle als frei gekennzeichnet und damit kann die normale Funktionsteilung zwischen ESAP und MUX20 stattfinden. Solange diese Freischaltung nicht erfolgt ist, führt der ESAP eine Art Notbetrieb auf dem Hostinterface aus, indem er Kanalkommandos mit Fehlerstatus beendet. Ausnahmen sind bestimmte Kommandos wie ABFÜHLEN, TEST und ein spezielles DIAGNOSE-LESEN.

2.4.3 Emulationsprogramm EMLMUX

Jeder MUX20 innerhalb eines MUX30A erhält **sein** Emulationsprogramm, es wird im Hostrechner aus einer Grundversion für die konkrete Konfiguration erzeugt und wie oben beschrieben in den jeweiligen MUX20 geladen und aktiviert. Diesem Programm obliegt die eigentliche Emulation des Zielgerätes MPD4.

Den bis zu acht MUX20 ist hostseitig der ESAP vorgeschaltet, aus der Sicht des EMLMUX wirkt dieser wie folgt:

- als Zuordner zwischen maximal 16 zugeordneten Subkanal-Adressen am Hostkanal und den maximal 16 logischen Kanälen des MUX20.
- als Interfacewandler zwischen dem Hostinterface und dem internen MRK-Bus, dabei werden die Funktion „Einkellern eines Gerätestatus“, sowie die Komandos TEST und ABFÜHLEN allein vom ESAP ausgeführt.

Jeder MUX20 führt **auf dem MRK-Bus** zum ESAP hin einen eigenständigen statistischen **Multiplexbetrieb** zwischen seinen logischen Kanälen aus. Dazu werden Datensätze ausgetauscht, die faktisch als Abbild der Steuerfolgen auf dem Hostrechnerkanal zu interpretieren sind. Nur beim Start eines einzelnen Kanalkommandos bzw. einer Kette von Kanalkommandos geht dabei die Initiative vom ESAP aus. Die Abarbeitung und Beendigung eines Kanalkommandos erfolgt durch den MUX20, indem dieser Zuschaltfolgen beantragt. Mit einer Zuschaltfolge werden (bei Lesekommandos) Daten übergeben oder (bei Schreibkommandos) Daten angefordert und zur Beendigung eines Kommandos werden Endstatus- und Abfühlbyte an den ESAP übergeben. Wesentlich für die erreichten Leistungsdaten ist hierbei, dass bei jedem Datentransfer eine Multibytegruppe transportiert wird.

Zur Leitungsseite hin findet ein **Simultanbetrieb** statt, gesteuert wird dieser von den Interrupts der SIO und CTC. In Anlehnung an das emulierte Hardwaregerät bzw. den MPD4 [1] gibt es sogenannte Leitungsadapter. Ein Leitungsadapter ist hierbei ein Satz von Programmroutinen. Er ist für eine Gruppe von Terminals mit gleichen oder weitestgehend ähnlichen Datenübertragungsverfahren zuständig. Verfügbar sind folgende Leitungsadapter:

- der **AD1** wird für 5-Bit-Fernschreiber über Standleitungen benutzt.
- der **AD2** ist ein Testadapter und bedient keine externen Geräte. Er ermöglicht das Einlesen von Speicherinhalten des MUX20 wie Traceprotokollen usw.
- der **AD3** wird zum Fernanschluss von Terminals mit der Prozedur nach EC8564 über Fernsprechstandleitungen mittels Modem bzw. über GDN-Übertragungsstrecken benutzt.
- der **AD6** ermöglicht über Fernsprechstandleitungen mittels Modem bzw. über GDN-Übertragungsstrecken die Kopplung nach den Datenprozeduren BSC1 oder BSC3.
- der **AD7** diente dem Anschluss an das Telexnetz der DDR und stellt wie schon beim MPD4 eine besondere Lösung dar. Durch eine Simulation wird einen anderer Gerätetyp nachgebildet, die Telexleitung wurde vom Zentralrechner her als Schreibmaschinen-Terminal AP70 an einer Fernsprechwählleitung angesprochen, hier wurde also ein AD4 codiert.

Hinsichtlich der Erzeugung eines konkreten Emulationsprogramms kann in einem ersten Schritt eine Variante als Kombination verschiedener Adapter ausgewählt werden. In einem zweiten Schritt erfolgt eine Konfigurierung. Hierbei werden jedem logischen Kanal die physischen Adressen der SIO und CTC und der Leitungsadapter zugeordnet und es werden weitere Parameter eingestellt.

2.4.4 Test- und Hilfsprogramme

Im Festwertspeicher des MUX20 sind eine Reihe von Test- und Hilfsprogrammen verfügbar, die den Systemanlauf und die Ausführung von verschiedenen Testaufgaben ermöglichen und dabei miteinander kommunizieren.

Das **Monitorprogramm MONI** ist das zentrale Programm zur Bedienung aller Testprogramme des MUX20 und wirkt als Bindeglied zwischen Hardware und Software beim Systemanlauf. Es befindet sich im Festwertspeicherbereich der ZRE-Steckeinheit (Adressbereich 0...0BFFH) und ist unmittelbar beim Systemanlauf wirksam, wobei zunächst lediglich eine Warteschleife ausgeführt wird, die in folgenden Fällen verlassen wird:

- vom ESAP wird als Teil seiner Anlaufdiagnose ein Test der Koppereinheit BKE15 ausgeführt, dann wird der zur entsprechenden Routine des unten genannten Programms MKOF verzweigt. Nach dem Test wird erneut in die o.g. Warteschleife eingenommen.
- vom ESAP wird angezeigt, dass MONI durch das Serviceterminal des ESAP bedient werden soll. Daraufhin wird die Kommunikation von MONI in Zusammenwirken mit dem ESAP aktiviert.
- es wird erkannt, dass ein lokal am MUX20 angeschlossenes Terminal die Bedienung übernimmt. (Dazu muss dieses Terminal am SIO mit der Grundadresse 0E0H angeschlossen sein und hinsichtlich seiner Prozedur passen, siehe unten). Daraufhin wird die Kommunikation von MONI in Zusammenwirken mit diesem Terminal aktiviert.
- bei allen anderen Aktionen des ESAP wird davon ausgegangen, dass es sich um das Laden des Emulatorprogramms EMLMUX handelt, dann wird zur Laderoutine LAD verzweigt.

Im MUX30A stellt die Bedienung des Programms vom Serviceterminal des ESAP her den Normalfall dar. Es kann dabei allerdings jeweils nur mit einem MUX20 kommuniziert werden. Dabei hat der MUX20 am MRK-Bus den Zustand ONLINE, das ESAP-Steuerprogramm markiert zum Hostrechner hin alle diesem MUX20 zugeordneten Subkanaladressen als nicht betriebsbereit.

Das Monitorprogramm MONI übernimmt den Start anderer Programme und die Kommunikation mit diesen, es ermöglicht eine Reihe von für die Programmentwicklung nützlicher Funktionen wie ZRE-Registeranzeigen, Befehlsschrittbetrieb und das Setzen von Testpunkten (im RAM !). Weitere Funktionen sind Anzeigen von Speicher- und Registerinhalten, Eingaben in den RAM und in Register, Befehlsschrittbetrieb, Programmstart, Verschieben von Speicherinhalten und die Arbeit mit einer Lochbandperipherie, sofern die entsprechende Hardware (ATA) im System ergänzend verfügbar ist. Um die Funktion Lochbandausgabe nutzen zu können, muss ein vierter EPROM auf der PFS-Steckeinheit (Adresse 0C000H) vorhanden sein. Die Kommandoeingabe wird grundsätzlich mit dem Zeichen Punkt beendet,

MONI wird auch in der K1600-Variante des MUX20 und im Konzentrator KON20 verwendet. Im ESAP ist das Programm ebenfalls auf der ZRE-Steckeinheit vorhanden, hier es wird aber für die Funktionen des ESAP nicht wirklich benutzt, es sorgt lediglich für den Ansprung des ESAP-Steuerprogramms nach Netzeinschalten oder RESET.

Die **Laderoutine LAD** ist ein eigenständiges Programm, das als Partnerprogramm des ESAP-Steuerprogramms das Laden des Emulatorprogramms EMLMUX in den RAM-Speicher des MUX20 ausführt und dieses Programm nach dem Laden startet. Darüber hinaus kann die Routine einen beliebigen Speicherbereich des MUX20 in den Hostrechner zurücklesen. Diese Funktion war bei der Entwicklung der Software hilfreich.

Das **MUX/KON-Offlinetestprogramm MKOF** dient zur Funktionsprüfung und Fehlersuche innerhalb eines MUX20 sowie hinsichtlich der Kopplungen auf der Leitungsseite. Es stehen 22 unterschiedliche Tests zur Verfügung, welche sich in folgende Gruppen einteilen lassen:

- interne Funktionen für MUX/KON20 wie Prüfung des Interruptsystems, Test des RAM-Bereichs, Ermittlung von CRC-Summen im EPROM-Speicher,
- Überprüfungen der SDA-Kanäle eines MUX/KON20 durch Schleifung der Signale. Hier ist zwischen Wartungsmodus, Kurzschlussbetrieb, direkter externer Schleifung und Schleifung unter Einbeziehung von Datenübertragungsstrecken zu unterscheiden,
- Betrieb und damit Testung von Terminals am MUX/KON20: Es kann jeweils eine Leitung als Hostsimulator für folgende Terminals fungieren, dabei können Datenübertragungsstrecken Teil der Konfiguration sein:
 - Fernschreiber an Standleitungen
 - Fernschreiber am Telexnetzes
 - Terminal im Start/Stop-Prozedur nach EC8664
 - Terminal mit BSC-Prozedur nach EC7925,
- Emulation eines Terminals durch einen MUX/KON20. Hier können die oben genannten Geräte auch funktionell nachgebildet werden, womit die Kommunikation zwischen zwei MUX/KON20 für Testzwecke möglich ist.
- ein Servicetest für Einstellarbeiten an Leitungen eines MUX/KON20
- der Test der Buskuppeleinheit **BKET** (siehe unten).

Alle Tests werden vom Monitorprogramm MONI aufgerufen bzw. gestartet, dabei übernimmt MONI die Ein- und Ausgaben. Beim Aufruf wird der konkrete Test in den RAM-Bereich verschoben und hier gestartet, damit sind Eingriffsmöglichkeiten gegeben. In der Regel sind vor der eigentlichen Testausführung eine Reihe von Parametern einzugeben.

Die erfolgreiche Durchführung der Kurzschluss- und Schleifenprüfungen setzt entsprechende Hilfsmittel (Kurzschlussstecker bzw. Schleifenkabel) voraus. Bei der Auslieferung des MUX30A waren die IFSS- und die TAK- Kontakte an den Anschlussfeldern mit einer Schleifenverschaltung versehen, die erst im Rahmen der Systeminstallation beim Anwender aufgetrennt wurde.

Für den Einsatz des MUX20 im MUX30A hat der Test der Buskuppeleinheit **BKET** eine besondere Bedeutung:

- zum einen wird ein Teil dieses Tests beim Systemanlauf durch die Anlaufdiagnose des ESAP-Steuerprogramms automatisch durchlaufen,
- zum anderen kann im ESAP eine sogenannte Testinitialisierung veranlasst werden in deren Folge BKET aufgerufen und gestartet wird. BKET ist danach in der Lage als Partner des ESAP-Steuerprogramms einen unbedienten Emulationsbetrieb auf einer logischen Adresse auszuführen, sodass eine Zusammenarbeit mit dem Hostrechner gegeben ist. Damit ist ein aussagekräftiger Test der Hardwarekette Hostinterface – ESAP – MUX20 möglich, ohne dass im MUX20 ein Emulatorsteuerprogramm EMLMUX aktiviert werden muss. Dieser Test kann multiplex mit mehreren bzw. allen MUX20 ausgeführt werden.

2.5 Integriertes Testsystem

2.5.1 Allgemeines

Aus den Erfahrungen mit dem vorherigen Gerät MPD4 [1] war deutlich geworden, dass die Inbetriebnahme von Datenfernverarbeitungs-konfigurationen und eine Fehlersuche in derartigen Systemen in der Regel mit hohem Zeitaufwand verbunden ist. Erschwerend kam hinzu, dass im Einsatzzeitraum keine diesbezügliche Mess- und Prüftechnik für den Service beim Anwender zur Verfügung gestanden hat. Deshalb wurden die bereits beim MPD4 implementierten Technologien bei der Entwicklung des MUX30A weiter ausgestaltet. Im Ergebnis ist im MUX30A ein modular gestaltetes Testsystem integriert. Dabei wird eine Teststrategie verfolgt, die davon ausgeht, dass nur Teile des Systems – wie ein Subsystem MUX20 oder auch nur eine einzelne Übertragungsleitung – aus dem Anwendungsbetrieb herausgenommen werden, um Testaufgaben auszuführen. Basis dieser Strategie ist die **ServiceRoutine** des ESAP-Steuerprogramms in Verbindung mit den bereits genannten Test- und Hilfsprogrammen im MUX20.

2.5.2 Anzeige- und Protokollfunktionen

Vom Grundsatz her benötigt der MUX30A keine Bedienung. Nach Netzeinschalten oder DEVICE RESET werden alle ONLINE gesetzten Subsysteme MUX20 in den Anwenderbetrieb einbezogen und können vom Hostrechner durch Laden eines Emulationsprogramms aktiviert werden. Das Serviceterminal muss nicht vorhanden sein. Wenn es vorhanden ist, werden in Verbindung mit der Anlaufdiagnose Diagnosecodes angezeigt. Darüber hinaus kann es zur Ausgabe von **Fehlernachrichten** kommen, die entweder vom ESAP-Steuerprogramm oder vom Emulationsprogramm im MUX20 generiert werden.

Über eine Kommandoeingabe am Serviceterminal können zunächst Informationen zum Verhalten des Systems gewonnen werden, dabei wird der Anwendungsbetrieb nicht beeinträchtigt. Nachfolgend werden einige typische Betriebsfälle ohne Anspruch auf Vollständigkeit dargestellt:

- per Servicekommando kann die **Anzeige** eines Speicherbereichs oder von Systemzuständen des ESAP ausgelöst werden ohne dass dadurch ein Eingriff in die internen Funktionen erfolgt. So wird z.B. nach Eingabe des Kommandos **[OSxx.]** auf einer Zeile das aktuelle Subkanalsteuerwort des Subkanals xx dargestellt, man kann durch Eingabe von **[+]** zum nächsten Subkanalsteuerwort wechseln oder durch Eingabe von **[C]** die zyklisch wiederholte Ausgabe (Überschreiben auf der gleichen Zeile) des aktuellen Subkanalsteuerwortes veranlassen. So kann die Abarbeitung der Kanalkommandokette für den betreffenden Subkanal visuell verfolgt werden.
- unmittelbar nach einem Systemanlauf, d.h. bevor der Hostrechner mit dem Initialisieren des MUX30A begonnen hat, kann durch Eingabe des Kommando **[PA.]** der ESAP in den **Protokollmodus** versetzt werden. Das ist mit einer Performanceeinbuße verbunden, die allerdings nur bei Performanceanforderungen im Grenzbereich spürbar werden. Im Protokollmodus werden alle Aktionen in einem Umlaufspeicher protokolliert. Dazu muss die zweite Festwertspeicher-Steckeinheit vorhanden sein. Die Protokollierung wird gestoppt beim Erkennen einer Fehlersituation im Steuerprogramm (dann wird eine Fehlernachricht erzeugt und an das Serviceterminal gesendet) oder durch Eingabe des Befehls **[PH.]** am Serviceterminal, wobei das System weiterhin normal arbeitet. Das Protokoll kann am Serviceterminal angezeigt und/oder vom Hostrechner ausgelesen werden.

2.5.3 Service mit MUX20 im Offlinebetrieb

Am übersichtlichsten können die Testmöglichkeiten eines Subsystems MUX20 genutzt werden, wenn dieses System mit einem eigenen Serviceterminal ausgerüstet (ggf. temporär für eine Servicehandlung) und dann einfach in den Offlinezustand geschaltet wird. Problemlos ist dies jedoch nur möglich, wenn der SDA-Kanal mit Adresse 0E0H für ein Serviceterminal mit passender Betriebsweise (IFSS oder V.24 bei jeweils 9600 bps und ohne DFV-Prozedur wie K8911 und VT340) frei gehalten oder freigeschaltet werden kann. Bei der weitgehend ähnlichen Applikation KON20 und MUX20 im System K1600 war vorgesehen, auch ein vom System unterstütztes Anwendungsterminal als Serviceterminal zu nutzen. Die dazu nötige Vorgehensweise ist grundsätzlich auch hier anwendbar und setzt den Einsatz der speziellen Steckeinheit SBM K0423 voraus. Auf dieser Steckeinheit befindet sich ein DIL-Schalter, dessen Codierung beim Systemanlauf vom Monitorprogramm MONI abgefragt wird, was zur Auswahl eines anderen Terminaltypes (z.B. eines Fernschreibers) genutzt werden kann.

Beim MUX30A war die virtuelle Durchschaltung des zentralen Serviceterminals über den ESAP hinweg auf den MUX20 zweifellos der vom Service verwendete Standard. Das erfolgt, indem ein Subsystem MUX20 in einen logischen Offlinezustand geschaltet wird und danach vom Serviceterminal des ESAP bedient werden kann, während die anderen MUX20 des Systems im Anwenderbetrieb verbleiben.

Dazu muss am betreffenden MUX20 zunächst die Taste ON/OFF gesetzt werden. Dies führt zunächst zum definierten Ende der aktiven Kanalprogramme auf den betroffenen Leitungen und danach zum hostseitigen Sperren der entsprechenden Subkanaladressen im ESAP. Erst dann wird der Zustand OFFLINE erreicht. Danach ist ein RESET des MUX20 auszuführen und die Taste ON/OFF zu lösen. Auf das Kommando **[K x0.]** (x=0...7 : Nummer des MUX20) am Serviceterminal führt das System eine Kommunikation mit dem Programm MONI des jeweiligen MUX20 aus. Soll nach einem solchen Test der Anwendungsbetrieb wieder fortgesetzt werden, so muss der MUX20 erneut rückgesetzt werden (Setzen ON/ OFF, RESET, Lösen ON/ ON) und vom Hostrechner muss ein erneutes Laden des EMLMUX ausgeführt werden.

In erster Linie ist diese Betriebsweise sinnvoll, um die Ressourcen des Programms MKOF (siehe Pkt. 2.4.3.2) hinsichtlich der Testung von Leitungsanschlüssen, Übertragungseinrichtungen und Terminalkopplungen zu nutzen. Grundsätzlich ist diese Technologie auch dazu geeignet, den Prozessorkern eines solchen MUX20 für andere Aufgaben zu nutzen, sofern diese auf der Basis des Monitorprogramms MONI gestaltet ist. So wurde mit Erfolg eine **Datenflussanalyse** unter Nutzung einer Funktionsgruppe MUX20 realisiert und bei mehreren Anwendern betrieben.

2.5.4 Service mittels CCW-Simulation

Der Test nach Abschnitt 2.5.3 entzieht eine ganze Leitungsgruppe dem Anwenderbetrieb und dies ist zweifellos dann nachteilig, wenn lediglich Servicearbeiten an einer einzelnen Leitung auszuführen sind. Für derartige Servicefälle bietet sich die Nutzung der CCW-Simulation an. Bei der CCW-Simulation arbeitet das EMLMUX im MUX20 ganz normal weiter, es bemerkt praktisch nicht, dass die zum Test ausgewählte Leitung nicht mehr mit dem Hostrechner, sondern direkt mit der CCW-Simulationsroutine des ESAP zusammenarbeitet.

Wer dieses Verfahren anwenden will, muss über das Grundverständnis hinsichtlich des Funktionsprinzips am Peripheriekanal eines IBM360-kompatiblen Systems verfügen. Danach benutzt das System sogenannte Kanalprogramme, jeweils bestehend aus (in der Regel) mehreren Kanalkommandoworten (engl: CCW), um mit seinen Ein/Ausgabeadressen zu kommunizieren.

In Vorbereitung einer CCW-Simulation muss zunächst ein **Kanalprogrammmodell** kreiert werden. Ein solches Kanalprogrammmodell besteht aus Kanalkommandoworten zu jeweils 8 Byte. Kommandocode (1 Byte), Datenadresse (2 Byte) und Bytezähler (1 Byte) sind die ersten und wesentlichsten Elemente eines derartigen internen Kanalkommandowortes. Zur Eingabe dieses Kanalprogrammmodells in den dafür vorgesehenen RAM-Bereich (ab 1D80H) steht das Servicekommando **[IC.]** zur Verfügung. Die einzelnen Kanalkommandoworte müssen fortlaufend eingegeben werden, dabei dient **[>]** zur Weiterschaltung des Speicherbereichs modulo 8 (auf den Anfang des nächsten Kanalkommandowortes). Der RAM-Bereich 1E00H...1FFFFH ist für Sende- und Empfangsdaten reserviert und wird in der Form von 16 Buffern zu je 64 Byte verwaltet. Für die Eintragung von Daten in diese Buffer ist das Kommando **[IB0x.]** vorgesehen. Darüber hinaus stehen Datenmuster im PROM als Nachrichten an Terminals zur Verfügung (deren Adressen sind in der Beschreibung angegeben).

Sobald Kanalprogrammmodell und Buffer-Daten eingetragen sind, kann eine beliebige Subkanaladresse (es sollte natürlich ein zum Kanalprogrammmodell passendes Terminal zugeordnet sein) für den Test aktiviert werden. Das Servicekommando **[A xx.]** bewirkt, dass nach Beendigung eines eventuell noch laufenden Kanalprogramms des Hostrechners der Subkanal mit Adresse xx hostseitig gesperrt wird. Der eigentliche Test mit der CCW-Simulation wird durch das Kommando **[S 0yxx.]** gestartet (y=0...F gibt die Nummer des zu startenden Kanalkommandowortes im Kanalprogrammmodell an, xx dient als Durchlaufzähler für zyklische Wiederholungen).

Auf dem Serviceterminal wird nach dem Start eine zyklisch wiederholte Ergebniszeile angezeigt, die eine direkte Beobachtung des Testablaufes und des Testergebnisses ermöglicht.

Im Kanalprogrammmodell lassen sich bedingte und unbedingte Sprünge einordnen, über Pseudokommandos kann erreicht werden, dass Daten an das Serviceterminal ausgegeben bzw. von diesem übernommen werden können. Damit sind Kommunikationen zwischen Anwendungsterminal und Serviceterminal möglich.

Anmerkung:

Diese Testmethode hat wahrscheinlich im praktischen Serviceablauf keine größere Rolle gespielt, die Testdurchführung war relativ kompliziert und erforderte detaillierte Systemkenntnisse, sobald man über die beschriebenen Testbeispiele hinausgehen wollte. Der Verfasser verfolgte deshalb die Zielstellung, im Anschluss der Entwicklung des ESAP-Steuerprogramms das Programm des Serviceterminals dahingehend zu modifizieren, dass eine Anwendungsebene zur Generierung der CCW-Daten im Serviceterminal verfügbar sein sollte. Diese Absicht konnte nicht verwirklicht werden.

3 Geschichte der Entwicklung

Wie in [1] und [2] dargestellt, sind im Zeitraum 1970 bis 1978 die wesentlichsten Arbeiten zur Datenfernverarbeitung bei Robotron im Entwicklungsbereich Radeberg gelaufen. Dabei war dieser Entwicklungsbereich bis Ende 1974 dem GFZ/ZFT Dresden unterstellt, hatte aber gleichzeitig bedingt durch Aufgaben im Bereich der Mainframe-Architektur eine starke fachliche Anbindung an Strukturen des GFZ/ZFT-E2 Chemnitz (Karl-Marx-Stadt). Ergebnisse dieser Jahre waren die Datenfernverarbeitungs-Lösungen mit MPD4 und LK4221.

1975 wurde dann der Bereich Datentechnik als Hauptabteilung ED in Radeberg wieder in ein eigenständiges Direktorat Entwicklung eingegliedert und war nun neben einem Bereich Rundfunk-/Fernsehempfänger und einem Bereich Richtfunktechnik formal eigenständig, war aber weiterhin fachlich in die Arbeiten der genannten ZFT-Strukturen eingebunden und konnte auch nur so die folgenden 10 Jahre überleben. Die Leitfunktion für Datenfernverarbeitung übernahm ZFT-E41 (Dresden). Hier entstanden später auch wesentliche Teile der neuen Software, der Anteil der Beiträge des LFA verringerte sich.

Zunächst wurde Radeberg durch ZFT-E2 in Arbeiten zur Mikrorechnerlinie K1520 einbezogen und so konnten die vorhandenen Erfahrungen hinsichtlich der seriellen Übertragungsverfahren bei der Funktionsanalyse des Z80-SIO-Schaltkreises eingebracht werden. Diese Arbeiten waren Grundlage für die Entwicklung der SDA-Steckeinheiten in Radeberg.

Im Zusammenhang mit den konzeptionellen Arbeiten zur Rechnerlinie K1600 entstanden vor 1980 im ZFT Dresden die ersten Vorgaben für die Entwicklung einer Reihe neuer DFV-Steuereinheiten unter Nutzung von Komponenten der Mikrorechnersysteme K1520, K1620 und K1630. Die weiteren Untersuchungen zeigten aber, dass mit diesem Ansatz eine Systemlösung analog der SNA von IBM nicht machbar sein würde. In dieser Arbeitsphase wurde deshalb auch eine Entwicklung auf der funktionellen Grundlage der IBM3705 unter Nutzung der Technologie des Rechnersystems R55 konzeptionell durch ZFT-E2 betrachtet, entsprechende Arbeiten konnten aber nicht eingeordnet werden.

Es wurde entschieden:

- zunächst das Thema **MUX/KON20** für den Einsatz in der Systemfamilie K1600 zu bearbeiten,
- zeitversetzt unter Nutzung des MUX20 den Multiplexer **MUX30A** für die ESER-Mainframes zu entwickeln, der ausschließlich den Emulationsmodus beherrschen musste. Für SNA-Lösungen sollte auf den DFV-Prozessor EC8371 aus Polen zurückgegriffen werden.

In den Gremien des ESER konnte dieses neue Gerät als Modernisierung des Vorgängers EC8404 dargestellt werden und so wurde die Chiffre EC8404.M1 vergeben und es konnte auf einen aufwändigen ESER-Test verzichtet werden.

Das Thema MUX/KON20 wurde bis etwa 1983 in Radeberg unter der fachlichen Federführung von Hans-Peter Sauer bearbeitet, so entstanden die weiter oben beschriebenen Funktionsgruppen SDA und TAK sowie die Grundlagen des Mehrrechnerkoppelbuses und die Funktionsgruppen BKE15, BKE16 und KVK. Unter Leitung von Dieter Hofmann wurde ein Cross-Assembler zur Erstellung von K1520-Programmen auf der vorhandenen R21 und die Programme MONI und MKOF entwickelt.

Damit waren die Anwendungen im Zusammenhang mit der Systemfamilie K1600 realisierbar, die Anwendungssoftware entstand im ZFT Dresden.

In den oben genannten Vorgaben war vorgesehen, in der Zentralsteuerung der Emulationsvariante MUX30A einen Mikrorechner K1620 einzusetzen. An dieser Stelle möchte der Verfasser in einen persönlichen Stil der Berichterstattung wechseln.

Ich war als Gruppenleiter - oder nannte sich das in dieser Zeit Fachgebietsverantwortlicher - für die fachliche Realisierung der Zentralsteuerung des MUX30A zuständig und hatte ein Konzept für die funktionelle Durchbildung der Aufgabenstellung erarbeitet. Nach meinen Untersuchungen brachte der Einsatz des K1620 keine Performancevorteile, aber neben höheren Kosten vor allem das Risiko, eine weitere Softwaretechnologie einbinden zu müssen. Entstanden war ein Feinkonzept zur grundsätzlichen Funktionsverteilung zwischen Zentralsteuerung und MUX20, zur Hardwarestruktur und zur Software der Zentralsteuerung. Dieses Gesamtkonzept auf Basis eines K1520 wurde zur Realisierung freigegeben. Aus verschiedenen Gründen habe ich mich dann entschlossen, die gesamte Software der Zentralsteuerung (das ESAP-Steuerprogramm, es erreichte einen Umfang von ca. 11.000 Zeilen Quellcode) selbst zu programmieren und auszutesten. Meine Zielstellung war, ein Maximum an Performance zu erreichen und gleichzeitig eine praktikable Servicetauglichkeit sicherzustellen. Programmiert habe ich das zu großen Teilen zu Hause nach Feierabend ohne zusätzliche Vergütung. Die umfangreiche Hardware (der EKST) wurde durch Johannes Böhm und Siegfried Jährg entwickelt. In dieser Arbeitsphase wurden einige Veränderungen am ursprünglichen Konzept notwendig, an die perfekte Zusammenarbeit mit diesen Kollegen in dieser schwierigen Arbeitsphase denke ich heute noch gern zurück.

1982 wurde zunächst ein Labormuster des MUX30A aufgebaut. Um Zeit zu sparen – damals war die Leiterplattenkonstruktion ein zeitaufwändiger Prozess - wurden die neuen Hardwarekomponenten in Form einer Handverdrahtung auf Rasterleiterplatten aufgebaut. Abbildung 7 zeigt eine der acht Leiterplatten der EKST-Hardware. Die Erprobungen erfolgten an einem Rechner R21 im Testlabor Haus Technik in Radeberg. An diesem Labormuster wurden das endgültige Design der Hardware und das ESAP-Steuerprogramm erarbeitet. Erst mit Abschluss dieser Arbeitsphase begann die Trassierung der Leiterplatten, dabei war es notwendig geworden, die Leiterplattenanzahl auf neun zu erhöhen.

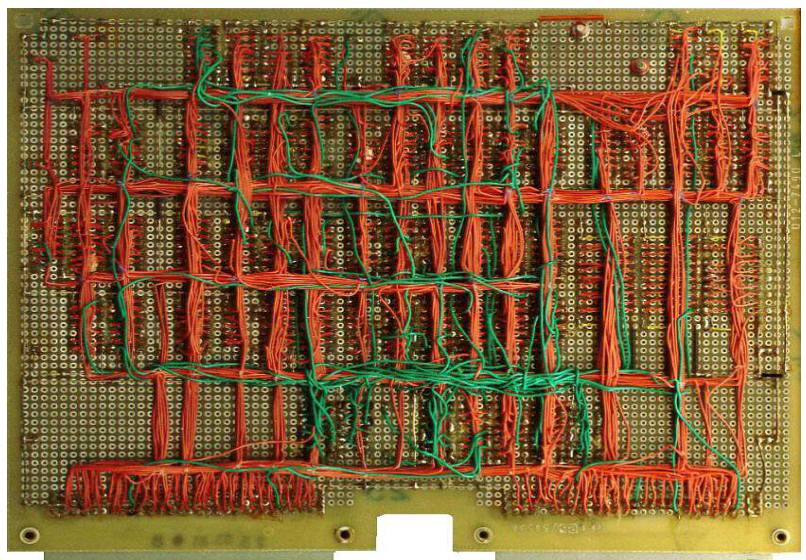


Abbildung 7: Labormuster einer EKST-Steckeinheit

Ein zweites ähnlich aufgebautes Labormuster wurde im zweiten Halbjahr 1982 im ZFT Dresden eingesetzt, um die Software EMLMUX in den MUX20 im Zusammenspiel mit dem ESAP an aktuellen Mainframes unter Systembedingungen auszutesten. Das EMLMUX und ergänzende Softwaremodule für den Hostrechner waren durch ein Team des ZFT-E41 um Dr. Günter Pollender entwickelt worden. Im Jahr 1983 wurden dann fünf Funktionsmuster aufgebaut und an Kooperationspartner und erste Anwender ausgeliefert.

Ein technisches Problem dieses Zeitabschnitts bestand darin, dass Kontaktunsicherheiten der EPROM-Bausteine in den Fassungen auftraten. Die Ursache dafür war, dass zunächst auf dem Weltmarkt beschaffte Speicherschaltkreise und Fassungen diverser Hersteller und Ausführungen verwendet wurden, Goldkontakte waren die Ausnahme. Das Problem wurde dadurch abgestellt, dass die an anderer Stelle entwickelte Steckereinheit PFS K3822 mit eingelöteten und auf der Steckereinheit programmierbaren EPROM verwendet wurde. Als deren Fertigung anlief, standen Speicherschaltkreise mit Goldkontakten aus der inzwischen angelaufenen eigenen Produktion zur Verfügung und so wurden nun diese eben eingelötet.

Im Zusammenhang mit der Entwicklung des MUX30A entstanden einige Innovationen:

- Weil entsprechende Messtechnik in der DDR nicht verfügbar war, wurden Möglichkeiten zur **Datenflussanalyse** auf V.24-Kanälen geschaffen. Erste Improvisationen waren mit dem MPD4 erfolgreich durchgeführt worden [1]. 1981 wurde auf dieser Grundlage ein Neuerer-vorschlag eingereicht und letztendlich realisiert, der im Kern eine einfache Weichen-schaltung für ein V.24-Interface darstellte, welche den Sendekanal und den Empfangskanal jeweils einem SDA-Kanal zuführt. Die Analyse dieser zwei Kanäle erfolgte durch einen MUX20 mit einer speziellen Analyse-Software. Diese Lösung wurde mehrfach (>10) nachgenutzt.
- Zunächst bestand bei der Fertigungsprüfung des Hostanschlusses die Notwendigkeit, einen entsprechenden Großrechner (eine R55) einzusetzen. Um die Belegungszeiten dieses Rechners zu reduzieren, erfolgte eine **Simulation des Hostinterfaces** durch einen K1520-Mikrorechner mit einer speziellen Steckereinheit mit Z80-PIO-Schaltkreisen.
- Im ESAP-Steuerprogramm ist zusätzlich die Funktion **Bedienkonsole für Hostrechner** implementiert. Es ist damit möglich, das Serviceterminal des MUX30A direkt zur Bedienung des Hostrechners zu verwenden. Diese Funktion ist nicht dokumentiert und wurde wahrscheinlich nur im eigenen Testlabor genutzt.

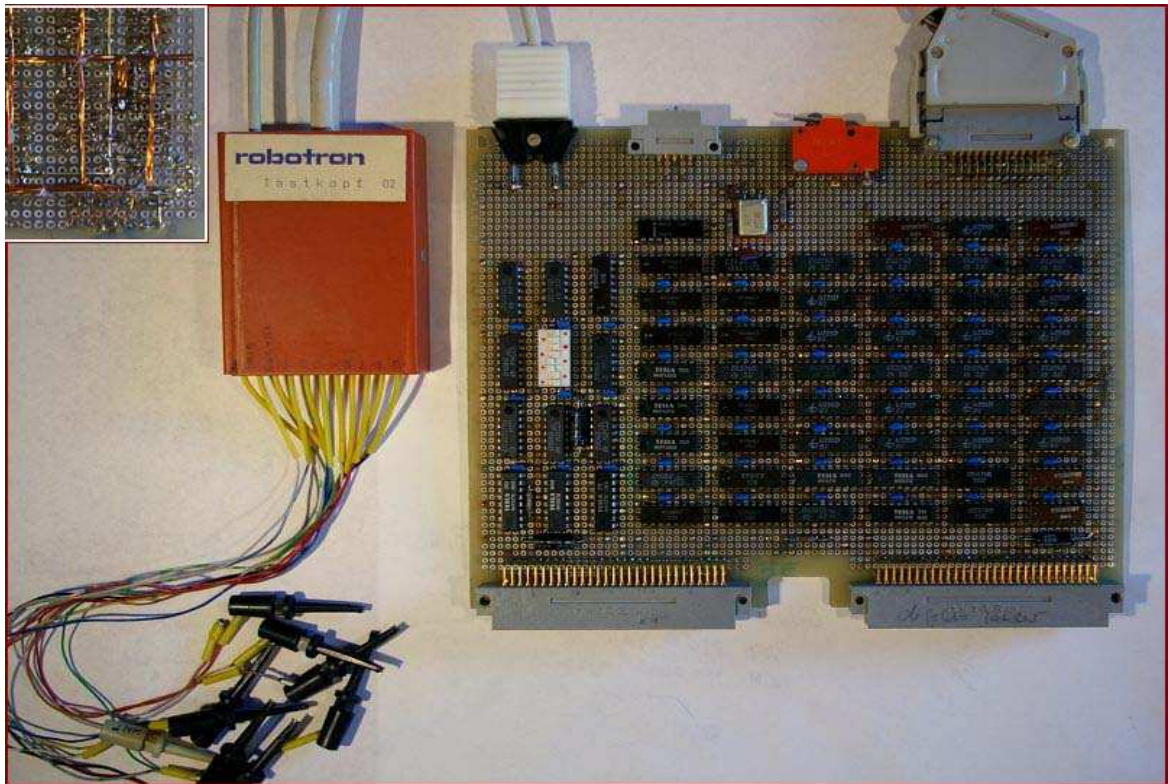


Abbildung 8: Logik-Analyse als K1520-Modul in Handverdrahtung

- Dietmar Kürth, damals Absolvent, entwickelte eine **8-Kanal-Logikanalyse** (20 MHz) in Form einer K1520-Steckeinheit und baute erste Muster in Handverdrahtung auf (Abbildung 8). Diese Steckeinheit verwandelte im Zusammenwirken mit einem speziellen Programm ein einfaches Serviceterminal K8911 (oder einen MUX20-Systemkern) in einen Logikanalysator und verbesserte damit die messtechnische Ausrüstung von Entwicklungs- und Fertigungsabteilungen ganz entscheidend.

Zum Ende des Jahres 1984 wurde nach einem entsprechenden Test durch die Gütekontrolle die Leistungsstufe K8-0 und damit der Entwicklungsabschluss bestätigt.

Die angestrebte Systemdurchsatzleistung ist nicht erreicht worden. Bei Vollausbau mit 128 Leitungen waren 1200 bps je Leitung möglich. Als Projektierungsrichtwert galten 25000 bps je MUX20.

Als eine Folge der Schaffung eines Bereichs Spezialtechnik (mit militärisch orientierten Aufgaben) in Radeberg wurde zum Anfang 1985 die Hauptabteilung ED aufgelöst, damit war die Grundlage einer Weiterentwicklung des MUX30A zerstört. An angedachte Modernisierungen (z.B. hinsichtlich der Mikrorechnerkerne) bestand seitens der staatlichen Leitung kein Interesse, die Betreuung durch die ehemaligen Entwickler, die danach auf verschiedene Struktureinheiten verteilt waren, wurde aber gewährleistet.

Ab 1985 wurde der MUX30A unter der Leitung von Heinz-Joachim Schroeter im Fertigungsbereich Datentechnik Radeberg in Serie gebaut. Die Einführung der Systeme bei den Anwendern verlief ohne wirkliche Probleme, es war nur eine Revision von Hardware und Steuerprogramm der Zentralsteuerung notwendig. In den folgenden Jahren bis einschließlich 1989 sind mehr als 300 Systeme ausgeliefert worden, dabei sind diese Lieferzahlen unter den Absatzmöglichkeiten des Vertriebes geblieben. Ein wesentlicher Anwender waren die Sparkassen, hier gab es im Raum

Berlin und im Bezirk Dresden umfassende Lösungen im Echtzeitbetrieb. Erwähnt sei auch die Deutsche Reichsbahn mit einem flächendeckenden Platzreservierungssystem, wobei dieser Anwender über eigene Übertragungsleitungen verfügte und damit bessere Voraussetzungen als Behörden und Industrie hatte, die auf Leitungsbereitstellungen durch die Deutsche Post der DDR angewiesen waren.

Zum Anfang 1989 wurden der Bereich Spezialtechnik wieder zivilen Aufgaben zugeführt und später aufgelöst. Es wurde begonnen, den MUX30A in die an anderen Stellen laufenden Entwicklungsarbeiten zu Datennetzen einzuordnen und es fanden (erstmal!) Tests zum Zusammenwirken mit Richtfunk-Übertragungssystemen des eigenen Hauses statt. Im ersten Halbjahr 1990 gab es nochmals eine Vertriebsoffensive seitens des Robotron-Anlagenbau mit Unterstützung durch die Entwicklung. Diese Bemühungen blieben ergebnislos, Bestellungen wurden storniert, die gesamte Fertigung Datentechnik wurde eingestellt.

Im Rückblick möchte ich festhalten, dass die Bearbeitung dieses Projektes eine außerordentlich interessante Aufgabe gewesen ist. Nicht zuletzt deshalb, weil es für das Projekt nur eine Zielstellung aber keine Vorbildlösung gegeben hat, konnte eine eigenständige Architektur geschaffen werden. Es ist gelungen, ein stimmiges Gesamtkonzept zu finden und das komplexe Zusammenwirken der verschiedenen Hardwarekomponenten und der diversen Softwaremodule zu gewährleisten. Für die ausnahmslos gute Zusammenarbeit mit allen Beteiligten möchte ich mich herzlich bedanken.

4 Abkürzungsverzeichnis

CCW	Kanalkommandowort
DFV	Datenfernverarbeitung
DUE	Datenübertragungseinrichtung
ESAP	ESER-Anschluss-Prozessor
ESER	Einheitliches System der elektronischen Rechentechnik
GDN	Gleichstrom-Datenübertragung mit Niedrigpegel
IFSS	Interface sternförmig seriell (20mA-Stromschleife)
LFA	Leitzentrum für Anwendungsforschung der VVB Maschinelles Rechnen
MPD	Multiplexer für Datenübertragung, die Abkürzung resultiert aus der russischen Bezeichnung; z. B. MPD4 (EC8404)
MUX	Multiplexer
OPS	Operativspeicher (beim Mikrorechnersystem K1520)
PFS	Festwertspeicher (beim Mikrorechnersystem K1520)
RGW	Rat für gegenseitige Wirtschaftshilfe
SDA	Serieller Datenadapter
SNA	Systems Network Architecture
TAK	Telegrafie-Anschaltkassette
VVB	Vereinigung volkseigener Betriebe
ZFT / GFZ	Zentrum für Forschung und Technik / Großforschungszentrum im VEB Kombinat Robotron
ZRE	Zentrale Recheneinheit (beim Mikrorechnersystem K1520)

5 Literatur- und Bildverzeichnis

- [1] Gutbier, H.: **Die Entwicklung des Multiplexors MPD 4 EC8404**

<http://heinzgutbier.de>

- [2] Gutbier, H., Greve, B.: **Der Beitrag des Werkes Radeberg an Entwicklung und Produktion von Datentechnik im VEB Kombinat Robotron**

<http://heinzgutbier.de>

Abbildungen

Abbildungen Verfasser