

<b>Zeitschrift:</b>	Technische Mitteilungen / Schweizerische Post-, Telefon- und Telegrafenbetriebe = Bulletin technique / Entreprise des postes, téléphones et télégraphes suisses = Bollettino tecnico / Azienda delle poste, dei telefoni e dei telegrafi svizzeri
<b>Herausgeber:</b>	Schweizerische Post-, Telefon- und Telegrafenbetriebe
<b>Band:</b>	53 (1975)
<b>Heft:</b>	5
<b>Artikel:</b>	Eine experimentelle Anlage für Rechner-Ein- und -Ausgabe von digitalisierten Tonsignalen
<b>Autor:</b>	Kündig, Albert / Müller, Heinrich / Vejlstrup, Jan Grove
<b>DOI:</b>	<a href="https://doi.org/10.5169/seals-875597">https://doi.org/10.5169/seals-875597</a>

### **Nutzungsbedingungen**

Die ETH-Bibliothek ist die Anbieterin der digitalisierten Zeitschriften auf E-Periodica. Sie besitzt keine Urheberrechte an den Zeitschriften und ist nicht verantwortlich für deren Inhalte. Die Rechte liegen in der Regel bei den Herausgebern beziehungsweise den externen Rechteinhabern. Das Veröffentlichen von Bildern in Print- und Online-Publikationen sowie auf Social Media-Kanälen oder Webseiten ist nur mit vorheriger Genehmigung der Rechteinhaber erlaubt. [Mehr erfahren](#)

### **Conditions d'utilisation**

L'ETH Library est le fournisseur des revues numérisées. Elle ne détient aucun droit d'auteur sur les revues et n'est pas responsable de leur contenu. En règle générale, les droits sont détenus par les éditeurs ou les détenteurs de droits externes. La reproduction d'images dans des publications imprimées ou en ligne ainsi que sur des canaux de médias sociaux ou des sites web n'est autorisée qu'avec l'accord préalable des détenteurs des droits. [En savoir plus](#)

### **Terms of use**

The ETH Library is the provider of the digitised journals. It does not own any copyrights to the journals and is not responsible for their content. The rights usually lie with the publishers or the external rights holders. Publishing images in print and online publications, as well as on social media channels or websites, is only permitted with the prior consent of the rights holders. [Find out more](#)

**Download PDF:** 14.01.2026

**ETH-Bibliothek Zürich, E-Periodica, <https://www.e-periodica.ch>**

# Eine experimentelle Anlage für Rechner-Ein- und -Ausgabe von digitalisierten Tonsignalen

Albert KÜNDIG, Bern  
Heinrich MÜLLER, Georgia/USA  
Jan Grove VEJLSTRUP, Zollikofen

621.391.3.037.37—501.222:681.322.06:621.376.56

*Zusammenfassung. Eine in der Abteilung Forschung und Entwicklung PTT gebaute Anlage gestattet es, digitalisierte Tonsignale in den Grossrechner IBM/370-168 des Elektronischen Rechenzentrums PTT einzugeben und dort für Simulationen und statistische Auswertungen zu speichern. Die Entwicklung der Anlage sowie einige Anwendungen werden beschrieben.*

**Installation expérimentale pour l'introduction et l'extraction de signaux acoustiques numériques dans un ordinateur**

*Résumé. Une installation construite à la division des recherches et du développement des PTT permet d'introduire des signaux acoustiques numériques dans l'ordinateur de grande capacité IBM 370-168 du Centre de calcul électronique des PTT et de les mémoriser pour des simulations et des évaluations statistiques. Les auteurs décrivent le développement de l'installation et quelques-unes de ses applications.*

**Un impianto sperimentale per immettere e prelevare da calcolatori segnali acustici in forma digitale**

*Riassunto. Un impianto costruito presso la divisione ricerche e sviluppo delle PTT permette di immettere nel grande calcolatore IBM/370-168 segnali acustici trasformati in valenze digitali dove possono essere memorizzati per scopi di simulazione e per l'elaborazione statistica. Si descrivono lo sviluppo dell'impianto e alcune possibilità d'impiego.*

## 1 Einleitung und Zielsetzung

### 1.1 Die Bedeutung von Simulationen mit Tonsignalen

Die Entwicklung der elektronischen Schaltungstechnik hat dazu geführt, dass heute für die Übertragung und Speicherung von Tonsignalen Verfahren in Betracht gezogen werden können, bei denen der Aufwand noch vor wenigen Jahren eine wirtschaftliche Verwirklichung verboten hätte. Im Telefoniebereich wird dabei als erstes an die *PulscodeModulation* (PCM) und verwandte Übertragungsmethoden, wie die *Deltamodulation* und *differentielle PCM* (DPCM) gedacht. Bereits sind auch sogenannte *Vocoder* (aus dem Englischen: «voice coder») erhältlich, die bei erheblichem schaltungstechnischem Aufwand eine extreme Bandkompression und damit die Übertragung oder Speicherung und Wiedergabe von Sprachsignalen mit beispielsweise 1200 bit/s gestalten. Viele dieser Methoden werden in öffentlichen Nachrichtennetzen mehr und mehr Bedeutung erlangen, sei es zum Beispiel bei der Mehrfachausnutzung wie im Falle der PCM oder bei der digitalen Speicherung von Tonsignalen für die automatisierte Ausgabe von Sprechtexten.

Dass die technische Entwicklung solche Verfahren in beachtlicher Vielfalt attraktiv gemacht hat, stellt nun aber den Entwerfer keineswegs vor leichte Probleme. Vielmehr gilt es meistens, aus einer Vielzahl *möglicher* Methoden die *qualitativ besten* auszusuchen beziehungsweise bei einer gegebenen Methode die Parameter so zu wählen, dass sich eine *billige* Konstruktion bei *genügender* Qualität ergibt. Häufig wird dabei die Kompliziertheit der Modulations- und Codierungsverfahren zum voraus ein Vorgehen verbieten, das aus einem praktischen, schaltungsmässigen Aufbau und dem Austesten aller Lösungsvarianten besteht. Zudem fehlen Methoden, um die Übertragungsqualität bei oft neuartigen Verzerrungen theoretisch vorauszusagen [1]. Der Gedanke liegt nahe, dass hier möglicherweise die Simulation einen Ausweg darstellen könnte, indem die verschiedenen Verfahren auf einem Mehrzweckrechner in der Form von Programmen nachgebildet werden, die auf reelle Tonsignale «einwirken». *Figur 1* zeigt das Prinzip dieses Vorgehens, wobei Echtzeitverfahren und Simulation in Stapelverarbeitung unterschieden werden müssen. Häufig lässt

nämlich die Kompliziertheit der Simulationen bei der beschränkten Arbeitsgeschwindigkeit von Mehrzweckrechnern nur die Stapelverarbeitung zu, so dass dann lediglich die Ein- und Ausgabe der digitalisierten Tonsignale in Echtzeit erfolgt, während vor der eigentlichen Simulation die Signale auf besonders geeigneten Datenträgern, wie Magnetbändern oder -platten, zwischengespeichert werden. Selbstverständlich wird es sich meistens nicht um eine Stapelverarbeitung im herkömmlichen Sinne handeln, indem es anzustreben ist, die Simulationen vom gleichen Bedienungsgerät aus wie die Ein- und Ausgabe interaktiv zu steuern. Die Zwischenspeicherung schafft zudem die Möglichkeit, beliebige statistische Analysen mit den Signalen vorzunehmen.

Die beschriebene Technik hat seit etwa 10 Jahren erfolgreich Einzug in vielen Laboratorien gehalten, die sich mit der Übertragung und Speicherung von akustischen Signalen befassen [2, 3, 4, 5]. In der gleichen Zeitspanne sind die für Simulationen notwendigen theoretischen Hilfsmittel stark ausgebaut und verbessert worden. Mit der Entwicklung von Methoden, wie der *schnellen Fouriertransformation* [6] wurden überhaupt erst die Voraussetzungen geschaffen, um entsprechende Simulationen mit verantwortbaren Re-

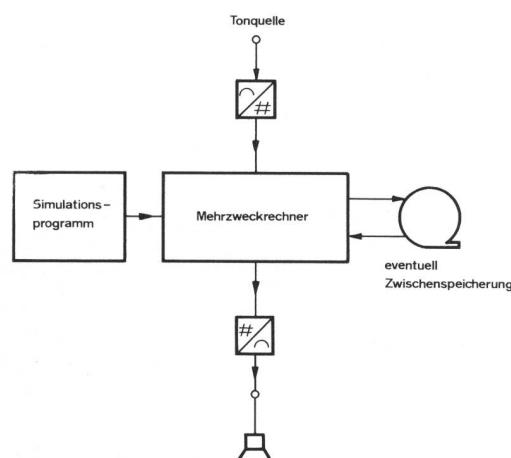


Fig. 1  
Simulation mit Tonsignalen

chenzeiten durchzuführen. Auch für die Abteilung Forschung und Entwicklung der PTT-Betriebe stellte sich die Frage, ob – bei einem der Schwergewichte der Entwicklungsarbeiten auf dem Gebiet der digitalen Nachrichtentechnik [7] – eine entsprechende Anlage nicht ein äusserst zweckmässiges Hilfsmittel für verschiedene Entwicklungsgruppen darstellen könnte. Mit dem Anschluss der Abteilung an das Elektronische Rechenzentrum PTT über einen Satellitenrechner [8] bot sich die Gelegenheit, dieses Vorhaben mit verhältnismässig bescheidenen Investitionen zu verwirklichen.

## 12 Anforderungen, Lösungsvarianten und Zielsetzung

Bei den Anforderungen müssen solche rein technischer und betrieblicher oder benutzerorientierter Natur unterschieden werden.

Im Interesse einer möglichst vielseitigen Verwendung sollte gefordert werden, dass die Anlage qualitativ sowohl für die Verarbeitung von Sprach- wie auch Musiksignalen geeignet ist. Daraus ergeben sich folgende technischen Richtwerte:

$$\text{Abtastfrequenz} \quad f_s > 32 \text{ kHz} \quad (1)$$

$$\text{Auflösung} \quad \geq 14 \text{ bit/Abtastwert,} \\ \text{gleichförmige Quantisierung} \quad (2)$$

$$\text{Dauer einer ununterbrochenen Ein- oder Ausgabe} \quad > 30 \text{ s} \quad (3)$$

Aus der Sicht eines potentiellen Benutzers in einem Entwicklungslabor konnte ferner etwa folgende Wunschliste zusammengestellt werden:

- Ein/Ausgabe und Simulationen möglichst auf der gleichen Anlage (ohne umständliche Verschiebung von Datenträgern) (4)
- Vorhandensein geeigneter Programmiersprachen und entsprechender Compiler (Übersetzungsprogramm) sowie leistungsfähiger Dienstprogramme für die Datenverwaltung (5)
- Einfache und rasche Bedienung; kleine Einarbeitungszeit, auch für gelegentliche Benutzer. Möglichkeit, die Anlage jederzeit benutzen zu können (6)
- Baukastenmässiger Aufbau sowohl auf der Hardware- wie Softwareseite, damit das System später leicht neuen Anforderungen angepasst werden kann (7)
- Möglichkeit, an den gespeicherten Tonsignalen nach der Eingabe oder vor der Ausgabe on-line gewisse Untersuchungen anzustellen, zum Beispiel eine Übersteuerungskontrolle oder die Darstellung von Tonsignalabschnitten im Zeit/Amplituden- oder Frequenzbereich (8)
- Die aus der Sicht des Benutzers wichtigen Systemfunktionen werden in 23 im Detail beschrieben. (9)

Von der Beschaffungsmöglichkeit einer betriebsbereiten Anlage musste zum vornherein abgesehen werden. Es existieren keine entsprechenden Produkte, und eine Fremd Entwicklung wäre aus Kostengründen nicht in Frage gekommen. Es galt also, unter weitgehender Verwendung käuflicher Bauteile und besonders aus vorhandenen Rechenanlagen ein System selbst zu entwickeln. Dabei zeigte es sich rasch, dass die Verwirklichung einer ausschliesslich für die Arbeiten auf dem Sprach- und Tongebiet zur Verfügung stehenden Anlage (beispielsweise im Rahmen des abteilungseigenen Prozessrechnerzentrums mit einer PDP-11-Maschine [9] wohl wünschbar wäre [Forderung (6)], aber des Aufwandes wegen nicht in Frage kam. Ein günstiger Ausweg ergab sich mit der Möglichkeit, wesentliche

Teile eines im *IBM-Forschungslaboratorium Rüschlikon/Zürich* entwickelten Programmpaketes zu übernehmen. Dieses Laboratorium verfügt bereits über eine vergleichbare Einrichtung [5] und ist wie das Elektronische Rechenzentrum PTT mit Anlagen des Systems IBM 360/370 ausgerüstet.

Damit stellten sich für die Entwicklung folgende Probleme:

- **Hardware:** Verbindung des in den Labors der Abteilung Forschung und Entwicklung aufgestellten Analog/Digital-Wandlers mit der Rechneranlage IBM/370-168 des ERZ durch eine Datenleitung mit variablen Bitraten bis mindestens 448 kbit/s (32 kHz × 14 bit) netto.
- **Software und System:** Anpassung der von IBM übernommenen Ein/Ausgabe-Programme an die lokalen Gegebenheiten. Anpassung und Erweiterung der ebenfalls übernommenen Anwenderprogramme. Verbindung der Rechenanlage mit einem geeigneten, im Labor aufgestellten Steuerterminal.

Diese Aufgaben werden im folgenden Abschnitt noch genauer umschrieben.

## 2 Aufgabenstellung für die Entwicklung

### 21 Räumliche Randbedingungen

Die unmittelbare Nachbarschaft des Gebäudes der Abteilung Forschung und Entwicklung (V) zum ERZ erleichterte zwar eine Benutzung der dortigen Rechenanlagen, stellte aber verbindungsmaessig doch einige Probleme. Es soll deshalb nicht versäumt werden, die räumliche Situation anhand von *Figur 2* darzustellen. Man erkennt, dass sowohl für das Steuerterminal wie für die Datenleitung eine etwa 300 m lange Verbindung zu erstellen war.

### 22 Rechnerkonfiguration

Die Rechnerkonfiguration ERZ/V, soweit sie für das hier behandelte Problem relevant ist, wird in *Figur 3* gezeigt. Es ergibt sich daraus, dass im besonderen folgende Punkte zu beachten waren:

- **Steuerterminal:** Wiederum aus Aufwandgründen konnte vorläufig nicht – wie im IBM-Forschungslabor – ein Bildschirmgerät als Steuerterminal verwendet werden. An dessen Stelle musste ein auch für den APL-Rechendienst [10] eingesetztes Schreibmaschinenterminal IBM 2741 benutzt werden. Dies schränkt die Möglichkeiten einer raschen Wiedergabe von Signalformen stark ein und läuft damit der Forderung (8) entgegen.

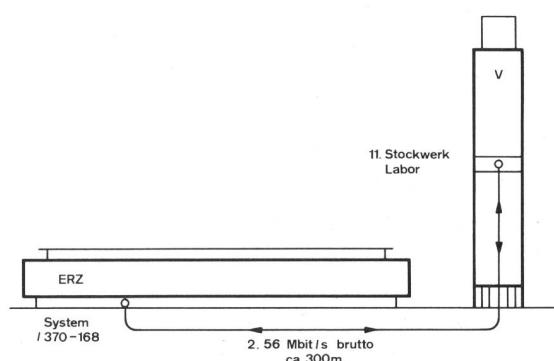


Fig. 2  
Situationsskizze

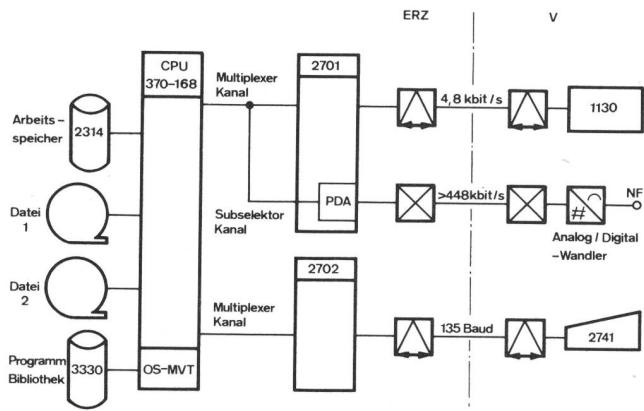


Fig. 3  
Konfigurationsschema ERZ/V (Auszug)

- Bestehendes Modem
- Zu entwickelnde Anpass- und Übertragungsausrüstung

- **Arbeitsspeicher:** Als Arbeitsspeicher stehen Plattspeicher Typ 2314 und 3330 zur Verfügung. Es war nicht im voraus zu klären, welcher der beiden Speicher betrieblich vorteilhafter eingesetzt werden kann. Versuche ergaben, dass mit dem Speicher 2314 weniger Unterbrüche zu verzeichnen sind, da dessen Verfügbarkeit vom System 168 aus durch entsprechende Prioritätszuteilung gross erscheint.
- **Digitale Tonsignaldateien:** Zweckmässigerweise werden dafür Magnetbänder eingesetzt, in Ausnahmefällen auch Plattspeicher. Ein Normalmagnetband genügt dabei für das Ablegen von digitalisierten Tonsignaldaten von maximal 12 min Dauer (20 kHz Abtastfrequenz).
- **Datenadapter:** Rechnerseitig geschieht die Übergabe der Daten bei einem Anschlussgerät vom Typ 2701, das für diesen Zweck mit einem sogenannten *parallel data adapter* (PDA) ausgerüstet ist. Im wesentlichen stehen daran je 16 Datenleitungen (entsprechend 16 bit je Abtastwert) für die Ein- und Ausgabe zur Verfügung. Auf speziellen Leitungen wird die Bereitschaft des Rechners signalisiert, einen Abtastwert aufzunehmen beziehungsweise abzugeben. Die eigens konstruierten Schnittstellenschaltungen (Abschnitt 4) müssen daher eine Anpassung des synchronen Datenflusses der laborseitigen Einrichtungen mit diesem asynchronen Rechnerbetrieb gewährleisten.
- **Rechnerkonsolen:** Eine Arbeitssitzung kann vom Satellitenrechner 1130 im Rahmen des Betriebssystems RJE/HASP/MVT gestartet werden. Besondere Fehlermeldungen werden aber nachher direkt auf der Konsole der ERZ-Anlage ausgedruckt, während das Schlussprotokoll wiederum über den Satellitenrechner ausgegeben wird.

## 23 Funktionelle Erfordernisse und Datenformate

### 231 Funktionelle Erfordernisse

Im wesentlichen wurden die bereits auf der IBM-Anlage in Rüschlikon vorhandenen Funktionen verwirklicht. Es sind dies (Fig. 4):

- **Eingabe** von digitalisierten Tonsignalen und deren Speicherung in einem Arbeitsspeicher. Der effektiv benutzte Bereich innerhalb dieses Speichers ist mit einer zeitlichen Auflösung von einem Abtastwert durch sogenannte

Marken (MARK 1 = Anfang, MARK 2 = Ende) spezifizierbar.

- **Ausgabe** der Signale aus dem Arbeitsspeicherbereich (MARK 1, MARK 2).
- **Ablegen** eines Arbeitsspeicherbereichs als sogenannten *subset* in einer Tonsignaldatei. Diese enthält im Prinzip beliebig viele aufeinanderfolgende subsets.
- **Laden** eines subsets aus der Tonsignaldatei in den Arbeitsspeicherbereich (MARK 1, MARK 2).
- **Spezifizierung** der *Ausdehnung* des benutzten Arbeitsspeicherbereichs durch Eingabe der Marken (MARK 1, MARK 2).

Da wie erwähnt eine grafische Ausgabe von Signalformen auf dem Schreibmaschinenterminal ausgeschlossen werden muss, wurde stattdessen zur Überprüfung der in einem Arbeitsspeicherbereich befindlichen Signale eine vom Terminal aus aufrufbare *Amplitudenstatistik* eingebaut.

Namentlich im Blick auf die Entwicklungsphase und beim Fehlersuchen werden folgende zusätzliche Funktionen programmiert:

- **Abfrage** von einzelnen *Abtastwerten* und bitweise Darstellung auf dem Steuerterminal
- Bitweises **Setzen** einzelner Abtastwerte vom Steuerterminal aus
- **Auffüllen** ganzer Arbeitsspeicherbereiche mit Null-Abtastwerten.

Ferner erwies es sich als zweckmässig, dem Benutzer gleichzeitig zwei getrennte Tonsignaldateien zur Verfügung zu stellen. Damit können die Manipulationen beim Aufbau neuer Dateien wesentlich vereinfacht werden, und die Gefahr des Überschreibens wertvoller Originaldaten wird verringert.

Die schliesslich verwirklichten Funktionen sind im vereinfachten Zustandsdiagramm in Figur 5 dargestellt. Daraus ist ausserdem ersichtlich, dass eine Ein- oder Ausgabe durch einen Doppelbefehl eingeleitet wird, und zwar durch den Vorbereitungsbefehl «Eingabe» (I) beziehungsweise «Ausgabe» (O) und die Auslösebefehle «Start» (A) oder «Wiederstart» (C). Durch den Stoppbefehl Z kann ein Datentransfer auch vor Erreichen von MARK 2 vom Steuerterminal aus abgebrochen werden, wobei die letzte verwendete Speicheradresse als MARK C festgehalten wird. Der Auslösebefehl C ermöglicht dann eine Wiederaufnahme der Ein/Ausgabe-Operation an dieser Stelle. Zu den eigentlichen Funktionssteuerbefehlen kommen verschiedene System- und Hilfsbefehle hinzu, und zwar:

- **FA** Löst den Ausdruck aller im Augenblick zulässigen Befehle (Funktionen) aus

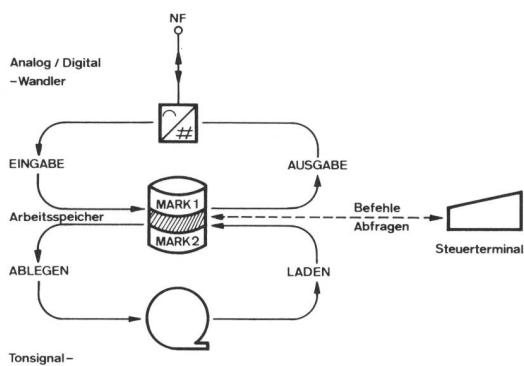


Fig. 4  
Funktionsmöglichkeiten

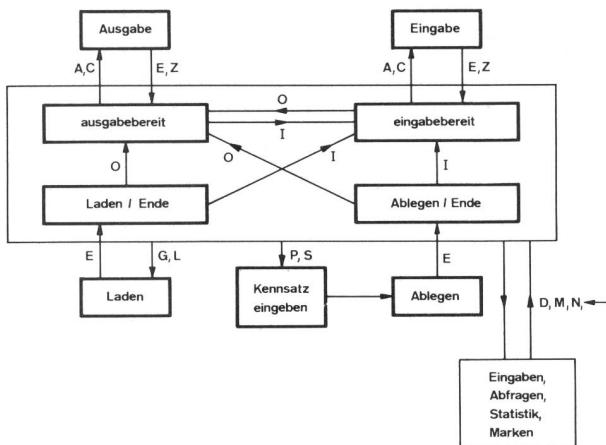


Fig. 5  
Vereinfachtes Zustandsdiagramm und Verzeichnis der Befehle

- A Start
- C Wiederstart
- D Amplitudenstatistik
- (E MARK 2 erreicht)
- G Laden aus Datei 2
- I Eingabe verlangen
- L Laden aus Datei 1
- M Marken setzen
- N Nullwerte eingeben
- O Ausgabe verlangen
- P Ablegen auf Datei 1
- R Nächsten Kennsatz auf Datei 2 suchen
- S Ablegen auf Datei 2
- T Nächsten Kennsatz auf Datei 1 suchen
- Z Stop
- ← Abtastwert eingeben

- )SI Bewirkt den Ausdruck des augenblicklichen Systemzustandes
- )MI Abfrage der derzeitigen Marken
- )OFF Schliesst eine Arbeitssitzung ab; das System antwortet mit der Bekanntgabe der Verbindungs- und Rechenzeit.

### 232 Datenformate

Je Abtastwert wird eine Gruppe von 16 bit (2 bytes) in Zweierkomplementcode [11] verwendet. Dieser Code entspricht der rechnerintern angewendeten Verschlüsselung und verhindert aufwendige Umcodierungen bei Simulationen.

Die vom Benutzer aus gesehen nächstgrößere Dateneinheit in den Dateien ist ein *logischer Satz* von 40 Abtastwerten (80 bytes). Diese Formatierung erlaubt eine einfache Speicherung auf allen lochkartenkompatiblen Datenträgern.

Die je Lade- oder Ablegevorgang transferierte Datenmenge umfasst im allgemeinen eine Vielzahl logischer Sätze und wird als *subset* bezeichnet. Jeder subset weist neben den Signaldaten am Anfang einen zusätzlichen *Kennsatz* von 80 bytes auf, in dem gemäß Figur 6 die wichtigsten Angaben über die abgespeicherten Signale enthalten sind. Die ersten 3 bytes, nämlich die Zeichen '=2C', dienen der eindeutigen Identifizierung des Subset-An-

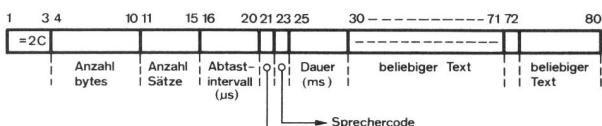


Fig. 6  
Aufbau der Kennsätze

fanges und wurden bewusst so gewählt, dass eine Imitation durch digitalisierte Sprachdaten sehr unwahrscheinlich ist. Mehrere subsets bilden schliesslich einen auch im Rechnerbetriebssystem bekannten Datensatz. In Figur 7 wird versucht, die Datenstruktur anschaulich darzustellen.

### 233 Fehlerbehandlung

Der Fehlerbehandlung in einem interaktiven Echtzeitsystem des beschriebenen Umfangs muss grösste Beachtung geschenkt werden, da sonst die Benutzerfreundlichkeit stark leiden könnte. Ziel sollte sein, dem Benutzer im Falle eines Fehlers die Informationen zu geben, die für eine rasche und zweckmässige Fortsetzung der Arbeit erforderlich sind. Dies gilt sowohl für Fehlersituationen, die der Benutzer selbst herbeigeführt hat – zum Beispiel durch Fehlmanipulationen – als auch für Systemfehler, zum Beispiel bei Überlastung der Anlage aus Echtzeitgründen. Der erste Fall muss in den Anwendungsprogrammen berücksichtigt werden, der zweite in den Ein/Ausgabe-Steuerprogrammen. Es sprengte den Rahmen dieses Artikels, wenn alle Fehlermöglichkeiten dargestellt würden. Stattdessen wird auf Figur 8 verwiesen, die eine typische Fehlerregistrierung auf dem Journal des Steuerterminals zeigt.

### 3 Niederfrequenzteil und Analog/Digital-Wandler

Der Niederfrequenzteil ist für asymmetrische Signale von  $\pm 10$  V ausgelegt. Der Frequenzbereich erstreckt sich von Gleichspannung bis über 32 kHz.

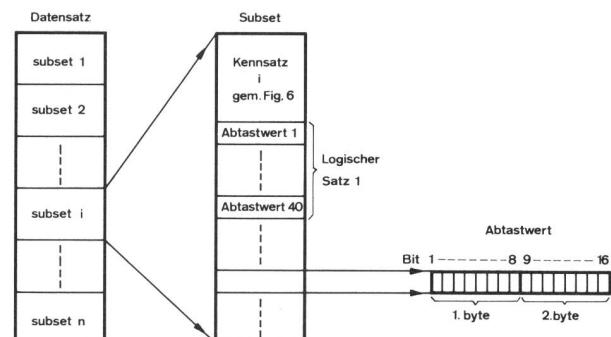


Fig. 7  
Datenstruktur

### 31 Vorfilter

Bei jedem System, in dem ein kontinuierliches Signal abgetastet und später verarbeitet und/oder rekonstruiert wird, muss das Rohsignal von Frequenzkomponenten, die über der halben Abtastfrequenz liegen, befreit oder zumindest auf ein vernachlässigbar kleines Mass reduziert werden. In der Praxis bedeutet dies, dass man das Eingangssignal ein geeignetes Tiefpassfilter durchlaufen lässt, bevor es zum Abtaster gelangt. Unterlässt man diese Filterung, so entstehen sogenannte Alias-Verzerrungen. Desgleichen benötigt man ein Filter, um zu hohe Frequenzen im treppenförmigen Signal am Ausgang des Digital/Analog-Wandlers zu eliminieren [11].

Um dem Benutzer diese Massnahmen zu vereinfachen, ist ein fertig gekauftes Gerät in die beschriebene Anlage eingebaut worden; damit können zwei voneinander unabhängige aktive Tiefpassfilter mit *Butterworth*-Charakteristik 8. Ordnung verwirklicht werden. Die Grenzfrequenz kann an beiden Filtern zwischen 0,01 Hz und 111 kHz auf drei Dezimalen genau eingestellt werden. Erfordert ein gegebenes

```

L,3,1000,10000
STATUS: LOAD END
Q
STATUS: OUTPUT WAIT
A
STATUS: OUTPUT RUN
Z
PDAC 7I BUFFER OVERRUN
MARKC= 14560 MARKE= 728000
STATUS: OUTPUT WAIT
)SI
STATUS: OUTPUT WAIT
)OFF
PCM I/O ENDED
DATE : 740913 TIME : 132422
CONNECTED : 1 16 1

```

Fig. 8  
Auszug aus dem Journal des Steuermeldes

- 1 Ladebefehl: subset 3, Abtastwerte 1000...10 000
- 2 Ende der Ladung
- 3 Ausgabe verlangen
- 4 Ausgabebereit
- 5 Start
- 6 Ausgabe
- 7 Stop
- 8 Fehlermeldung: Puffer überfüllt nach 14 560 Abtastwerten (Echtzeit-Probleme)
- 9 Status-Abfrage
- 10 Beendigung der Arbeitssitzung
- 11 Meldung der Anschlusszeit

Problem eine besondere Filtercharakteristik oder einen erhöhten Geräuschabstand, so ist es dem Anwender freigegeben, anstelle des vorhandenen Gerätes andere, unter Umständen passive Filter, in den Signalfluss zu schalten.

### 32 Abtastung und Analog/Digital-Wandlung

Die hohen Anforderungen, die in bezug auf Frequenzumfang, Auflösung und Geräuschabstand an ein Experimentiersystem der vorliegenden Art gestellt werden, fallen bei Abtaster und Analog/Digital-Wandler besonders schwer ins Gewicht. Es lassen sich auf dem Markt keine fertigen Geräte finden, die die gehegten Wünsche in allen Punkten befriedigen. Selbst wenn man Abtaster und Wandler getrennt aussucht und zusammenbaut, muss man kleine Kompromisse zulassen.

Gewählt wurden schliesslich ein Sample/Hold-Verstärker von Teledyne Philbrick (4853) und ein Analog/Digital-Wandler von Analogic (2715). Der Wandler ist für 15 bit ausgerüstet und benötigt für eine Umwandlung 16,5  $\mu$ s. Sein interner Takt kann jedoch schneller eingestellt und die Umwandlung nach dem 14. Bit abgebrochen werden, wenn man eine Einbusse an Genauigkeit und Auflösung in Kauf nimmt. Man kann so für Abtaster und Wandler zusammen bei 14 bit Auflösung zu einer Umwandlungsrate von 64 kHz gelangen. Diese Leistungen sind bei der Digital/Analog-Wandlung leichter zu erreichen. Es muss einzig für die Beseitigung der Spannungsspitzen, die beim Wechsel zwischen zwei aufeinanderfolgenden Abtastwerten auftreten können (*glitches*) besondere Sorgfalt aufgewendet werden.

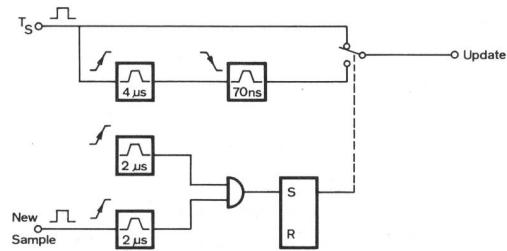


Fig. 9  
Antikoinzidenzschaltung

Die Firma Analogic bietet für diesen Zweck unter der Bezeichnung MP259 Transient Suppressing Amplifier den geeigneten Baustein an [12].

Beide Wandler besitzen eine gemeinsame Taktversorgung, die als Zeitbasis einen 8-MHz-Quarzoszillator enthält. Das Abtastintervall kann an dezimalen Ziffernschaltern in Schritten von 1/8  $\mu$ s eingestellt oder durch einen externen Takt vorgegeben werden.

Um auch den Digital/Analog-Wandler mit einem jitterfreien Takt versehen zu können, war eine besondere Schaltung nötig. Jedes Eintreffen eines neuen Abtastwertes am Eingang des Digital/Analog-Wandlers wird mit einem «New Sample»-Puls gemeldet. Aus verschiedenen Gründen, wie variable Leitungslänge, Charakteristik der Digitalfilter (siehe 44) und anderen, ist die zeitliche Lage gegenüber dem Abtasttakt  $T_s$  aber ungewiss. Jedoch bleiben diese Einflüsse während einer Ausgabeoperation konstant, vorausgesetzt, dass dabei die Abtastfrequenz nicht verändert wird. Außerdem ist der New Sample, der während genau eines Übertragungsbittaktes erscheint, gegenüber  $T_s$  mit zeitlichen Schwankungen von bis zu zwei solchen Takten behaftet, da Abtastung und Übertragung nicht synchron laufen (siehe 43).

Die in Figur 9 vereinfacht dargestellte Schaltung hat nun folgende Aufgabe: Beim Eintreffen des ersten Abtastwertes einer Output-Operation wird untersucht, ob  $T_s$  und New Sample mit mindestens 2  $\mu$ s zeitlichem Aufwand auftreten. Ist dies der Fall, so wird während der ganzen eben angefahrenen Output-Operation  $T_s$  direkt zum Laden des neuen Wertes in den Digital/Analog-Wandler (*Update*) verwendet, andernfalls mit einer Verzögerung von 4  $\mu$ s.

### 4 Schnittstellenschaltungen und Übertragung

Figur 10 zeigt das Blockschaltbild der Anlage. In den folgenden Abschnitten wird auf einige interessante Aspekte eingegangen.

#### 41 Laborseitige Einrichtungen

Immer war der Wunsch vorhanden, die Anlage später auch erweitern zu können. Deshalb ist darauf geachtet worden, den Aufbau so zu gestalten, dass einzelne Blöcke der laborseitigen Einrichtungen ersetzt oder neue Blöcke in den Signalweg eingefügt werden können. Dies wurde mit der aus Figur 11 ersichtlichen einfachen Blockstruktur erreicht. Der Abtastwert bei DATA IN wird vom betref-

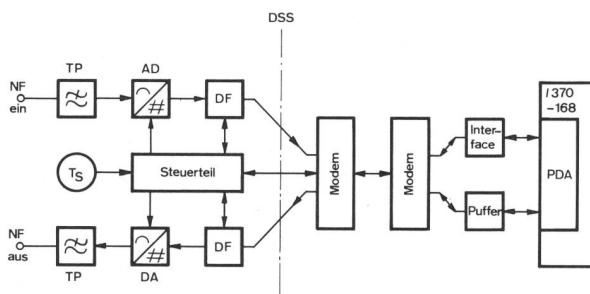


Fig. 10  
Blockschaltbild

- AD Analog/Digital-Wandler  
 DA Digital/Analog-Wandler  
 DF Digitalfilter (rekursiv, 2...8. Ordnung)  
 DSS Digitale Schnittstelle  
 PDA Parallel data adapter  
 TP Tiefpassfilter  
 $T_s$  Abtastintervall

fenden Block gemäss dessen Aufgabe verarbeitet, sobald an TRIG ein Puls erscheint. Ist der verarbeitete Wert bei DATA OUT verfügbar, gibt der Block dies durch einen Puls an NEW zu erkennen. Die beiden Wandler sind Sonderfälle solcher Blöcke, indem nur das digitale Ende die beschriebene Struktur aufweist. Einen weiteren Sonderfall stellt die *digitale Schnittstelle* (DSS, Fig. 10) dar: Sowohl logisch als auch physikalisch konzentriert (auf einen einzigen Stecker) finden sich dort die beiden Seiten eines «Blockes», der die Übertragung und die ganzen rechnerseitigen Einrichtungen umfasst. Zusätzlich erscheinen auf diesem Stecker zwei Signale *Input* und *Output*; sie signalisieren den (vom Steuerterminal aus kontrollierten) Zustand des Rechners. Die gesamten laborseitigen Einrichtungen können also an der digitalen Schnittstelle von Übertragung und Rechner getrennt werden. Der Anwender kann so einerseits die Anlage lokal (ohne Rechner) betreiben, anderseits ist es möglich, für besondere Probleme eigens konstruierte Geräte an dieser Schnittstelle einzufügen.

## 42 Übertragungsstrecke

Für die Übertragung zum 300 m entfernten Rechner stand ein PCM-Anschlusskabel mit 74 symmetrischen Aderpaaren zur Verfügung. Trotz des grossen Leitungsbuches entschied man sich, die Daten bitweise seriell zu übertragen. Auf diese Weise konnte dem Wunsch, galvanisch vollkommen vom Rechner getrennt zu sein, mit

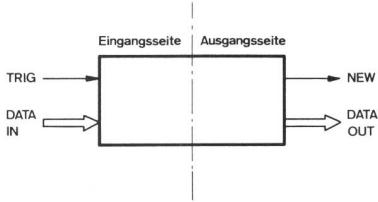


Fig. 11  
Blockstruktur

weniger Aufwand entsprochen werden als mit paralleler Übertragung der 16-bit-Abtastwerte. Dies um so mehr, als geeignete PCM-Zwischenverstärker bereits vorhanden waren (Taktfrequenz 2,56 MHz). Diese wurden nach kleinen Änderungen als Leistungstreiber und zur Pulsregeneration am Leitungsende eingesetzt.

Mit der variablen Abtastfrequenz ergibt sich für die Übertragung eine ebenfalls veränderliche Nutzbitrate. Im folgenden wird beschrieben, wie diese Tatsache mit dem starren Brutto-Bittakt der Übertragung in Einklang gebracht wurde.

Figur 12 zeigt ein Blockschaltbild des Parallel/Serie-Wandlers, der seinen Platz zwischen digitaler Schnittstelle und Leistungstreiber hat. Auf einen Puls an TRIG hin werden die beiden Schieberegister synchron mit dem nächsten Übertragungstakt mit dem neuen Abtastwert beziehungsweise dem Synchronisationswort geladen. Dann wird zuerst das Synchronisationswort auf die Leitung gegeben; ihm folgen die Datenbits, die, um das Leitungssignal gleichstromfrei zu halten, mit Hilfe der 1:2-Untersetzerstufe und dem gesteuerten Inverter (Exklusiv-Oder-Tor) in Dipulse codiert werden. Das 16-bit-Schieberegister ist inzwischen mit Einsen aufgefüllt worden; diese werden bis zum Eintreffen des nächsten TRIG-Pulses weitergesendet, um die Taktgewinnung auf der Empfangsseite dauernd zu gewährleisten.

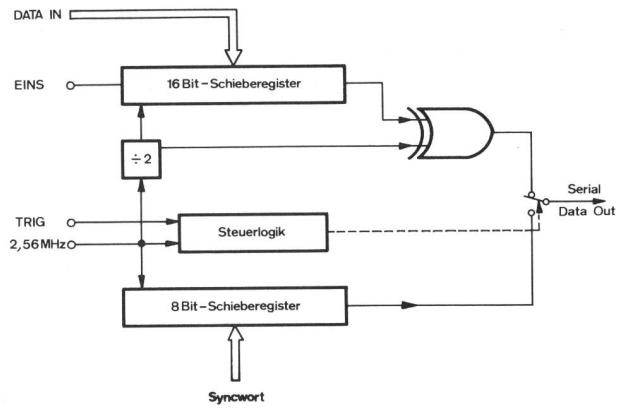


Fig. 12  
Parallel/Serie-Wandler (Sender)

In Figur 13 ist ein typisches Leitungssignal dargestellt. Für jeden Abtastwert wird ein Paket von 40 bit, bestehend aus einem 8-bit-Synchronisationswort und  $16 \times 2$  Datenbits, gesendet. Das Sync-Wort 10001110 enthält zwei Dipulsverletzungen; deshalb kann es von den Datenbits nicht imitiert werden. Der Empfänger überwacht den Bitstrom auf der Leitung und lädt bei der Detektion eines Sync-Wortes die zugehörigen 16 Datenbits in ein Register (Serie/Parallel-Wandler).

Bei der Ausgabe werden die einzelnen Abtastwerte beim Rechner «abgeholt»: Jedes auf der Input-Leitung detektierte Sync-Wort veranlasst die rechnerseitigen Einrichtungen zum Senden eines Ausgabe-Abtastwertes.

Auf einigen weiteren Leitungen werden Hilfssignale übertragen, die den Zustand des Rechners (Eingabe, Ausgabe) signalisieren, Fehler in der Übertragung melden und direkt kontrollierten Start, Unterbruch oder Abbruch einer Operation erlauben. Diese quasistatischen Signale werden auf der Empfangsseite mit Opto-Kopplern abgenommen, um auch hier die galvanische Trennung sicherzustellen.

## 43 Rechnerseitige Einrichtungen

Physikalische und logische Bedingungen für die Anschaltung an den Parallel Data Adapter (PDA) des Rechners finden sich in den einschlägigen Kapiteln von [13] und [14]. Die Schnittstelle besteht aus 34 Datenleitungen [ $2 \times (16 + \text{Parity})$ ] und 11 Steuerleitungen.

Neben Pegelanpassung und je einem Pufferregister für Eingabe und Ausgabe ist das *Interface* (die Steuerlogik) der Hauptbestandteil der rechnerseitigen Einrichtungen. Dieses tauscht mit dem PDA die nötigen Steuersignale aus, um auf diese Weise Rechner und Übertragung zu synchronisieren.

## 5 Programmierung

### 51 Programmarchitektur

Ein wesentlicher Baustein der hier beschriebenen Anlage ist der *Parallel Data Adapter* (PDA) (Fig. 3 und 10). Diese Einheit nimmt in der Hardware eine Stelle ein, die logisch

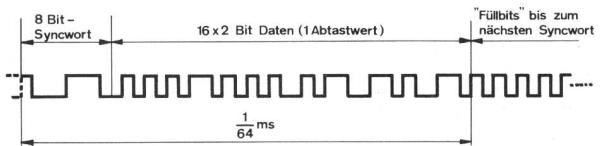


Fig. 13  
Pulsfolge auf der 2,56-Mbit/s-Verbindungsleitung für den Abtastwert 000011100101101

und programmtechnisch mit jener einer Magnetplatte oder eines Magnetbandes vergleichbar ist. Ein Gerät, das sich auf dieser logischen Ebene befindet, wird normalerweise als peripherie Einheit bezeichnet. Für solche Einheiten liefert die Rechnerfirma in den meisten Fällen Programme und Routinen, die es dem Benutzer erlauben, seine Anwendungsprogramme ohne Rücksicht auf die spezifisch hardwaremässige Funktionsweise der peripheren Einheiten zu erstellen. Diese Systemprogramme werden als *Zugriffsmethode* bezeichnet. Für den PDA besteht keine solche Zugriffsmethode: Im Normalfall muss sie der Benutzer selber erstellen. Wie erwähnt, konnte ein entsprechend entwickeltes Programm Paket vom IBM-Forschungslabor Rüschlikon übernommen werden, das diese Zugriffsfunktionen besorgt. Dieses Paket besteht aus einer Reihe von Programmen, die zum Teil in der Maschinensprache (*assembler*) und zum Teil in der Programmiersprache *PL/1* geschrieben sind; das ganze Paket wird im folgenden kurz mit *PUP* (*PDA-Unterstützungsprogramme*) bezeichnet. Um es einzusetzen, muss es durch ein *PL/1-Anwendungsprogramm* ergänzt werden. Das Anwendungsprogramm (AP) wird in die PUP integriert, und dabei wird eine Einheit gebildet (in der Terminologie des Betriebssystems ein *load module*), die vom Betriebssystem des Rechners als ein normales Benutzerprogramm, zum Beispiel eine Lohnabrechnung, angesehen wird. Um Platz im Hauptspeicher zu sparen, ist das load module in drei Abschnitte gegliedert: Initialisierung, eigentliche Verarbeitung und Abschlussarbeiten. Diese laufen nacheinander ab und können deshalb im Hauptspeicher überlagert werden. Sobald die Initialisierung beendet ist, wird zum Anwendungsprogramm verzweigt, das bei der eigentlichen Verarbeitung als Hauptprogramm fungiert; dabei nimmt es die Dienste der PUP in Anspruch (Fig. 14). Ist die Verarbeitung der Tondaten fertig (das heisst, wenn der Benutzer einen *OFF*-Befehl gegeben hat), wird die Kontrolle dem Unterstützungsprogramm für den Abschluss zurückgegeben.

## 52 Daten-Ein- und -Ausgabe

In Abschnitt 12 sind die Forderungen aufgezeichnet, die an die Anlage gestellt wurden. Dabei kommen den Punkten (1..3), wonach der Rechner die Daten im Echtzeitmodus aufzunehmen beziehungsweise auszugeben hat, besondere Bedeutung zu. Die Anlage ist imstande, Tondaten mit einer Geschwindigkeit bis zu 800 kbit/s (50 kHz  $\times$  16 bit) zu verarbeiten, und dies während mehreren Sekunden. Wenn man diese Forderung beurteilt, muss man bedenken, dass die Rechnerserie IBM/370 nicht speziell für derartige Verarbeitungen geeignet ist.

Im Grossrechner 370-168 des ERZ befinden sich normalerweise mehrere Programme gleichzeitig in Ausführung. Diese konkurrieren miteinander sowohl um die vorhandenen peripheren Einheiten als auch um die Zentraleinheit und die

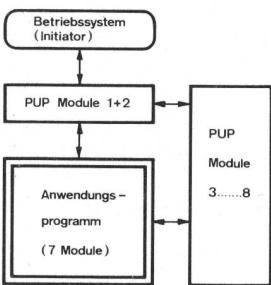


Fig. 14  
Prinzip des Programmaufbaus

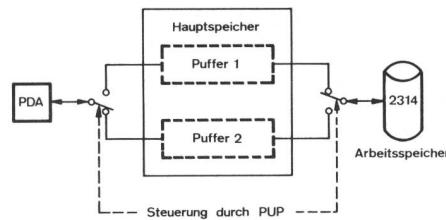


Fig. 15  
Prinzip der doppelten Pufferung

verschiedenen Rechnerkanäle. Dabei kann es vorkommen, dass ein Programm auf eine Einheit warten muss, weil diese bereits von einem anderen Programm verwendet wird. Das Betriebssystem sorgt für eine möglichst wirkungsvolle Zuteilung aller Ressourcen. Um die geforderte hohe Übertragungsrate zu gewährleisten, muss im besonderen sicher gestellt werden, dass die für den Datentransfer benötigten Einheiten und Kanäle nur ausnahmsweise anderen Programmen zur Verfügung gestellt werden, und dann nur für eine möglichst kurze Zeitspanne. Sonst wäre es kaum möglich, die verlangte Leistung zu erbringen.

Dadurch, dass sehr fortgeschrittene und komplizierte Werkzeuge des Betriebssystems (zum Beispiel *multitasking*, *program controlled interrupt*, *BDAM-macros*) in Anspruch genommen wurden, ist es gelungen, einen Programmkomplex zu erstellen, der die beschriebenen Echtzeitbedingungen erfüllt. Es sollen einige der Methoden kurz beschrieben werden, um einen Eindruck von den Lösungswegen zu vermitteln.

Ein Datentransfer zu oder von einer peripheren Einheit wird von einem *Kanalprogramm* gesteuert. Diese Kanalprogramme befinden sich auf der untersten Stufe der Software-Rechnerhierarchie. Sie gehören normalerweise den Zugriffsmethoden an und werden vom Hersteller geliefert; ihre Erstellung ist ausserordentlich aufwendig. Für den Parallel Data Adapter (PDA) aber besteht keine Zugriffsmethode, und die Kanalprogramme, die nötig sind, um den PDA zu betreiben, mussten als Bestandteil des Unterstützungsprogramms (PUP) extra für diese Anlage geschrieben werden. So war es aber auch möglich, sie den strengen Echtzeitbedingungen besonders anzupassen. Sie leisten, kombiniert mit der Inanspruchnahme ganz spezieller Dienste des Betriebssystems, einen wichtigen Beitrag dazu, dass die hohe Übertragungsrate überhaupt erreicht werden konnte.

Um die kurzen Unterbrechungen zu überbrücken, die entstehen, wenn das Betriebssystem ab und zu anderen Programmen Einheiten für kurze Zeit zur Verfügung stellt, benützen die PUP das Prinzip der doppelten Pufferung (Fig. 15). Im Hauptspeicher sind zwei Bereiche (Puffer) reserviert, die in ihrer Grösse je einer Spur des Platten-Speichers 2314 entsprechen, also eine Datenmenge von 7280 bytes oder 3640 Abtastwerte aufnehmen können. Während bei der Eingabe einer der Puffer vom PDA her aufgefüllt wird, besteht im allgemeinen genügend Zeit, um die Daten aus dem anderen zum Platten-Speicher zu transferieren. In gleicher Weise werden auch bei einer Datenausgabe die Puffer abwechselungsweise durchlaufen.

## 53 Steuerterminal

Wie bereits im Abschnitt 22 erwähnt, musste das Anwendungsprogramm für die Zusammenarbeit mit einem Schreibmaschinenterminal Typ 2741 abgeändert werden.

Es existiert wohl eine Zugriffsmethode für dieses Gerät, die darin enthaltenen Routinen können aber nicht von einem PL/1-Programm aufgerufen werden. Da das Anwendungsprogramm anderseits in dieser Programmiersprache erstellt werden muss, bedarf es also eines Schnittstellenprogramms zwischen der Zugriffsmethode und dem PL/1-Programm. Diese Routine musste in der Assembler-Sprache geschrieben werden. Der Auslegung der Routine wurden folgende allgemeine Ziele zugrundegelegt:

- Die Routine muss von einem PL/1-Programm aufgerufen werden können.
- Die Routine muss erlauben, vom oder zum Terminal eine Zeichenkette zu transferieren. Die maximale Länge der Kette wurde auf eine 2741-Zeile, entsprechend 130 Zeichen, festgelegt.
- Die Routine sollte so allgemein geplant sein, dass sie auch für andere Anwendungen zum Einsatz kommen könnte.

Bei der Programmierung der Routine wurde bald klar, dass neben der Beachtung der Minimalforderungen auch der Benutzerfreundlichkeit vermehrt Beachtung geschenkt werden musste. Unter anderem wurde die Routine so gestaltet, dass sie dem Benutzer mitteilt, wenn er eine Eingabe vorzunehmen hat. Bevor er Zeichen eintippen sollte, erscheint auf dem Terminal ein Unterstreichungszeichen (...); nach der Ausgabe dieses Zeichens wird der Schreibkopf um ein Zeichen zurückgesetzt, so dass das erste vom Benutzer eingegebene Zeichen gerade über die Unterstreichungsmarke gedruckt wird. Damit wurden zwei Vorteile erreicht, nämlich bessere Kommunikation mit dem Rechner und bessere Lesbarkeit der Terminalprotokolle (Fig. 8).

Es hat sich auch gezeigt, dass der Benutzer bei der Behandlung von Eingabefehlern unterstützt werden muss. Das Schreibmaschinenterminal 2741 besitzt eine Spezialtaste (die sogenannte *Attention-Taste*), die bei der Verwendung des Gerätes für den APL-Rechendienst von besonderer Bedeutung ist. Wenn der Benutzer sich bei der Eingabe vertippt, so bietet ihm die Routine die Möglichkeit, durch Betätigung dieser Unterbruchtaste die Eingabe abzubrechen und zu wiederholen. Ebenso ist es möglich, eine Ausgabe des Rechners mit der Taste abzustoppen.

#### 54 Anwendungsprogramm

Eine genaue Beschreibung des Anwendungsprogramms würde den Rahmen dieses Artikels sprengen. Es soll hier versucht werden, in einer kurzen Übersicht einige interessante Punkte zu behandeln.

Es stand von Anfang der Entwicklung an fest, dass das bereits bestehende Anwendungsprogramm (AP) nicht unverändert übernommen werden konnte (siehe 53). Einige darin enthaltene Funktionen würden sich nur mit einem Bildschirmterminal sinnvoll verwirklichen lassen. Es wurde daher entschieden, das Anwendungsprogramm – ausser der nur durch den Terminalwechsel bedingten Änderungen – auch in weiteren Punkten zu modifizieren und zu erweitern. Bei diesen Änderungen waren die Erfahrungen, die bei Arbeiten mit der Bildschirmversion gewonnen wurden, sehr nützlich.

Das Anwendungsprogramm, wie es nun zum Einsatz kommt, ist nur in den funktionellen Grundzügen mit der vorherigen Version identisch. Als wichtigste programmtechnische Änderung muss die Aufteilung in 6 Modulen mit insgesamt 19 *entry points* bezeichnet werden, die alle in PL/1 geschrieben sind. Dazu kommt noch das im vorhergehenden Abschnitt beschriebene Modul, das 3 *entry points* umfasst und in Assembler-Sprache codiert ist. Durch diese Aufteilung hat das Anwendungsprogramm an Transparenz gewonnen, und vor allem wurde das Aus- testen sehr vereinfacht. Die Durchführung zukünftiger Änderungen oder Erweiterungen ist durch diese Umstrukturierung wesentlich erleichtert worden.

Das Anwendungsprogramm hat, wie in 51 erwähnt, während der eigentlichen Verarbeitung die Rolle eines Hauptprogrammes, das für sich die Dienste der Unterstützungsprogramme beansprucht. Das bedeutet, dass in diesem Zustand das Anwendungsprogramm die Verarbeitung steuert. Diese Steuerung ist verhältnismässig einfach, aber für das Verstehen des Anwendungsprogramms (AP) sehr wichtig. Wenn das AP sich im Zustand befindet, in dem es einen Befehl des Benutzers erwartet, wird dieser durch das Unterstreichungszeichen verlangt. Sobald der Benutzer eine Zeichenkette eingetippt hat, werden die Zeichen auf Plausibilität untersucht und das AP stellt fest, um welchen Befehl es sich handelt. Ist der Befehl im gegenwärtigen Zustand des AP erlaubt, wird zu einem Modul gesprungen, das diesen Befehl ausführt; in allen anderen Fällen wird eine Fehlermeldung ausgedruckt und eine erneute Eingabe verlangt. Bei der Ausführung des Befehls kann es wiederum zum Austausch von Informationen mit dem Benutzer über das Terminal kommen. Wenn die Verarbeitung des Befehls beendet ist, wird unmittelbar eine neue Eingabe von Zeichen auf dem Terminal verlangt, so dass sich der beschriebene Zyklus schliesst.

In 231 findet man eine Übersicht über die vorhandenen Funktionen. Dieser ist zu entnehmen, dass in der überarbeiteten Version des Anwendungsprogramms zwei Tonsignaldateien zur Verfügung stehen. Die Überarbeitung gab dabei die Möglichkeit, auch den Ladebefehl für den Benutzer zweckmässiger zu konzipieren. Diese Änderung soll hier stellvertretend als Beispiel beschrieben werden. Dem Abschnitt 232 ist zu entnehmen, dass die Daten in einer Tonsignaldatei aus subsets zusammengesetzt sind. Diese sind sequentiell aneinandergereiht. Wenn der Benutzer ein bestimmtes (und nicht einfach das nächste) subset laden möchte, musste er bisher durch wiederholte Verwendung des Ladebefehls die Tonsignaldatei selber absuchen. Bei grossen Dateien kann dies sehr viel Zeit in Anspruch nehmen, so dass sich eine gewisse Automatisierung aufdrängt. Dem Benutzer ist es nun nach den Änderungen möglich, den Ladebefehl zusätzlich mit einer Nummer zu versehen. Die subsets in der Datei besitzen relative Nummern, und das Anwendungsprogramm besorgt selber das Aufsuchen, indem, beim Laden entsprechend viele Kennsätze abgezählt werden. Gleichzeitig wurde es durch eine weitere Option dem Benutzer möglich gemacht, nur Teile eines subsets zu laden.

Bei den Tests des Anwendungsprogramms erwies es sich als sehr nützlich, wenn die Versuche ohne zeitliche Abstimmung auf die Hardware-Entwicklung durchgeführt werden können. Zu diesem Zweck wurden zusätzliche Testroutinen erstellt, die die Funktionen der Unterstützungs-

programme simulieren. Ebenso wurden Hilfsprogramme erstellt, mit denen das Schreibmaschinenterminal durch einen Kartenleser und einen Schnelldrucker nachgebildet werden können. Die erwähnte Strukturierung des Anwendungsprogrammes in verschiedene, klar abgegrenzte Modulen mit einfachen Schnittstellen hat wesentlich dazu beigetragen, die Testphase zu erleichtern.

## 6 Erfahrungen, Anwendungen und Ausblick

### 61 Erfahrungen

Mit der aufgebauten Anlage konnten die gesteckten Ziele (Abschnitt 12) weitgehend erreicht und zum Teil übertroffen werden. Die Wahrscheinlichkeit, dass ein Datentransfer hoher Geschwindigkeit unterbrochen wird, nimmt naturgemäß mit zunehmender Auslastung des Rechnersystems durch parallellaufende Arbeiten zu. Ein wirkungsvoller Betrieb ist bis zu Abtastfrequenzen von 32 kHz ohne weiteres möglich, und bei geringer Systemaktivität können Werte bis zu 50 kHz erreicht werden. Damit erreicht man die durch die Rechnerkanäle selbst gegebenen Grenzen.

Eine Überprüfung der 300 m langen Übertragungsstrecke hat ergeben, dass die Fehlerrate unter  $10^{-10}$  liegt.

### 62 Anwendungen

Mit Hilfe der beschriebenen Anlage konnten bereits einige interessante Untersuchungen durchgeführt werden; weitere sind geplant. Anhand von drei Beispielen soll kurz illustriert werden, welche Anwendungsmöglichkeiten sich ergeben.

– **Statistische Untersuchung:** Gewinnung von Hilfskanälen aus PCM-Telefonikanälen.

Bekanntlich weisen Telefoniesignale während eines beträchtlichen Zeitanteils nur geringe Pegel auf. Dieser Effekt wird zum Beispiel in den TASI-Systemen praktisch ausgenutzt. Bei PCM-codierten Sprachsignalen führt er dazu, dass häufig die den grossen Gewichten entsprechenden Bits «0» sind. Aufeinanderfolgende derartige Codewörter könnten also teilweise für die Übertragung

von zusätzlichen Informationen benutzt werden, wenn man Beginn und Ende eines Blocks entsprechender Abtastwerte durch einen ausgewählten Code markiert. Im Empfänger werden dann die für den Hilfskanal benutzten Bits wieder durch Nullen ersetzt; es ergibt sich also keine Beeinträchtigung der Sprachqualität. Die Charakteristik des Hilfskanals, besonders die erreichbare Bitrate, wird weitgehend durch die statistische Verteilung der Blocklängen beschrieben. Ein Beispiel einer solchen Verteilung ist in *Figur 16* wiedergegeben. Sie zeigt, dass auch bei nachträglicher nichtlinearer CEPT-Codierung ein Hilfskanal von einigen kbit/s gewonnen werden kann.

- **Simulation nichtlinearer Quantisierung:** Es hat sich gezeigt, dass die Simulation verschiedener Quantisierungsmethoden mit höheren Programmiersprachen sehr einfach ist, sofern diese Bitmanipulationen erlauben. Die Nachbildung eines CEPT-13-Segmentkomponders benötigt beispielsweise einen Programmabschnitt (ohne Steuerung der Dateien) von lediglich 12 PL/1-Befehlen.
- **Simulation von Übertragungsstörungen:** Bequem lassen sich auch Übertragungsstörungen simulieren, indem einer gegebenen Statistik entsprechend fehlerhafte Abtastwerte in ein Sprachoriginal eingefügt werden. Diese Methode wird gegenwärtig benutzt, um geeignete Schutzmassnahmen bei PCM-Musikübertragung zu entwickeln.

Simulationen sind immer dann besonders vorteilhaft, wenn im Laufe einer Versuchsreihe verschiedene Parameter – unter Umständen in einem noch nicht zum voraus bekannten Bereich – variiert werden müssen. Diese Flexibilität lässt sich oft viel leichter erreichen als bei hardwaremässigen Versuchseinrichtungen.

### 63 Ausblick

Auf der Seite der Statistik- und Simulationsprogramme ist geplant, ebenfalls eine möglichst weitgehende Modularität zu erreichen. Es sollte letzten Endes möglich sein, ganze Übertragungssysteme durch Aneinanderreihen von Unterprogrammen, die den verschiedenen Systemblöcken entsprechen, nachzubilden. Hardwareseitig soll die Ein/Ausgabe-Einrichtung durch programmierbare Digitalfilter (Fig. 10) und allenfalls Komander ergänzt werden, so dass häufig verwendete Funktionen keine Rechenzeit mehr beanspruchen müssen.

Eine ständige Sorge bereitet die Entwicklung auf dem Gebiet des Betriebssystems. Normalerweise hat jede Neuausgabe dieser Programme durch die Lieferfirma des Rechners umfangreiche Neukompilierungs- und -assemblierungsarbeiten zur Folge. Besondere Probleme stellt dabei die geplante Umstellung auf die virtuelle Speichermethode, da die strengen Echtzeitbedingungen keine Umspeicherung der kritischen Programmteile während Ein- und Ausgaben auf den Plattspeicher gestatten.

Abschliessend sei an dieser Stelle die Zusammenarbeit mit dem IBM-Forschungslabor Rüschlikon verdankt, das wichtige Programmteile zur Verfügung stellte und auch bei der Implementierung hilfreiche Hinweise geben konnte.

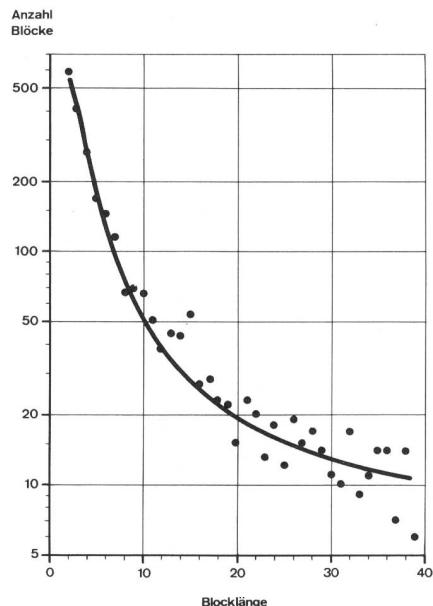


Fig. 16  
Blockhäufigkeit/8 MSB = 0, lineare Codierung (Anzahl Blöcke in 30-sec-Sprache)

## Bibliographie

- [1] *Kündig A.* Subjektive Versuche zur Ermittlung der Sprachübertragungsqualität. Bern, Techn. Mitt. PTT, 1973, Nr. 6, S. 230...241.
- [2] *Flanagan J. L.* Use of an interactive laboratory computer. IEEE Trans. AU-17 (1969), 1.
- [3] *Denes P.B.* On-line Computers for Speech Research. IEEE Trans. AU-18 (1970), 4.
- [4] *Cartier M., Génin J.* La réponse vocale: réalisations actuelles et perspectives d'avenir. L'onde électrique, 1974, n° 1.
- [5] *Rothauser E., Maiwald D., Ungerböck G., Käser H.* An experimental system for interactive on-line signal processing with an IBM system /360-50. 7. Internationaler Akustik-Kongress, Budapest 1971.
- [6] *Bergland G.D.* A guided tour of the fast fourier transform. IEEE Spectrum 6 (1969), no 7.
- [7] *Wuhrmann K. E.* Das integrierte PCM-FernmeldeSystem IFS-1. Bern, Techn. Mitt. PTT, 1973, Nr. 12, S. 554...578.
- [8] *Burgherr A.* Die Abteilung Forschung und Entwicklung bekommt eine eigene Datenverarbeitungsanlage. Bern, Techn. Mitt. PTT, 1973, Nr. 1, S. 11...14.
- [9] *Jaquier J.-J., Murbach G.* Der Einsatz von Prozessrechnern bei den PTT. Bern, Techn. Mitt. PTT (erscheint demnächst).
- [10] *Formanek Th.* Erste Erfahrungen mit dem APL-Rechendienst. Bern, Techn. Mitt. PTT, 1973, Nr. 4, S. 158...161.
- [11] *Kündig A.* Digitale Telephonie. Blaue Reihe, Heft 103. Bern, Hallwag 1972.
- [12] Analogic: Application Note on MP 259.
- [13] IBM Form A 22-6844: IBM 2701 Data Adapter Unit/OEM Information.
- [14] IBM Form GA 22-6864: IBM 2701 Data Adapter Unit/Component Description.

---

**Die nächste Nummer bringt unter anderem**  
**Vous pourrez lire dans le prochain numéro**

- |              |  |
|--------------|--|
| R. Vallotton | Übertragung frequenzmodulierter ZF-Signale von Richtstrahlantennen über längere, verstärkerlose Koaxialkabel |
| M. Herzig    | Mikrofilm – Partner der EDV  |
| E. Montandon | Die Verarbeitung der Störungsmeldungen gebäudetechnischer Anlagen in PTT-Objekten                            |
| P. Benoît    | Appareil de mesure du facteur de réduction   |
-