

KfK 2585  
Juli 1978

# **NOASYS, ein System zur on-line Analyse von Rauschsignalen**

H. Massier  
Institut für Neutronenphysik und Reaktortechnik  
Projekt Schneller Brüter

**Kernforschungszentrum Karlsruhe**

Als Manuskript vervielfältigt  
Für diesen Bericht behalten wir uns alle Rechte vor

KERNFORSCHUNGSZENTRUM KARLSRUHE GMBH

KERNFORSCHUNGSZENTRUM KARLSRUHE  
Institut für Neutronenphysik und Reaktortechnik  
Projekt Schneller Brüter

KfK 2585

NOASYS, ein System zur on-line Analyse von Rauschsignalen.

H. Massier

Kernforschungszentrum Karlsruhe GmbH, Karlsruhe



## Zusammenfassung

In diesem Bericht wird das Noise Analysis System NOASYS vorgestellt. Es handelt sich dabei um ein System zur digitalen on-line Signalanalyse mit einem Kleinrechner. NOASYS wird hauptsächlich bei der Rauschanalyse an Kernreaktoren eingesetzt. Ziel dieser Analysen ist die Ermittlung statistischer Kenngrößen von Rauschsignalen wie Auto- und Kreuzkorrelationsfunktion und spektrale Leistungsdichte, sowie deren Weiterverwendung bei der Entwicklung von Methoden zur Schaden-Früherkennung.

Mit NOASYS können bis zu 16 Analogsignale verarbeitet werden. Die maximale Abtastfrequenz beträgt 100 kHz (1 Kanal). Die Verarbeitung der Daten erfolgt mit Software-Tasks, die einzeln über Teletype aufgerufen oder zu aufgabenspezifischen Meßprogrammen zusammengestellt werden können. Bei diesen Tasks handelt es sich um häufig benötigte Signalverarbeitungsroutinen wie schnelle Fouriertransformation, komplexe Multiplikation, Summation usw. NOASYS kann leicht durch Integration zusätzlicher Tasks (Assembler oder Fortran Subroutinen) an verschiedene Aufgabenbereiche angepasst werden.

## NOASYS, a System for On-Line Noise Analysis

### Abstract

This report describes NOASYS, a versatile NOise Analysis SYStem for digital on-line signal processing. The system based on a minicomputer, was originally developed for the analysis of noise signals from nuclear reactors. NOASYS computes various statistical functions e.g. cross- and auto-correlation functions and power spectral densities resp., which may be used for reactor diagnosis and malfunction detection.

The system processes up to 16 analog signals with a maximum sampling frequency of 100 kcps (1 channel). The processing of the sampled data is done by a number of software tasks, which may be called from a teletype or linked together for specific measuring programs. The standard configuration contains the often used processing routines e.g. Fast Fourier Transform, complex multiplication, summation etc. In addition NOASYS may be extended by users own tasks (in Assembler or Fortran) to fit for specific applications.

## Inhaltsverzeichnis

	Seite
1. Einleitung	1
2. Digitale Signalanalyse	3
2.1. Korrelationsmethode	5
2.2. Direktransformation	7
3. Systembeschreibung	9
3.1. Hardware	11
3.1.1. Rechner und Peripheriegeräte	11
3.1.2. Meßdatenerfassung	12
3.2. NOASYS Software	14
3.2.1. Monitor und Task-Liste	15
3.2.2. Tasks	16
3.2.3. Sichtgerät und Steuerkonsole	17
3.3. NOASYS-Datenblöcke	20
3.3.1. Zugriff zu NOASYS-Datenblöcken	22
3.4. Beschreibung der Standard-Tasks	27
3.4.1. Monitor-Tasks	27
3.4.2. Steuer-Tasks zur Datenerfassung	29
3.4.3. Ein-Ausgabe Tasks	32
3.4.4. Datenblock-Operationen	36
4. Systemerweiterung	42
4.1. Systemgenerierung mit dem DOS-Loader	44
Abkürzungen	45
Literaturverzeichnis	47
Abbildungen	49

## 1. Einleitung

Zur Analyse der dynamischen Eigenschaften von Kernreaktoren werden schon seit längerer Zeit Methoden aus dem Bereich der Rauschanalyse angewandt. Daneben gewinnen diese Methoden eine zunehmende Bedeutung für die betriebliche Überwachung von Leistungsreaktoren im Sinne einer Schaden-Früherkennung /2,3/. Ein Überblick über die vielfältigen Anwendungsmöglichkeiten der Rauschanalyse an Kernreaktoren wird in /1/ gegeben.

Eine wichtige Voraussetzung für die Analyse des Reaktorrauschens ist die meßtechnische Erfassung und Verarbeitung der Schwankungen verschiedener physikalischer Zustandsgrößen. Dazu gehören z.B. Neutronenfluß, Kühlmitteldurchsatz, Brennelementaustrittstemperaturen sowie mechanische Schwingungen von Brennelementen. Während bei der konventionellen Meß- und Regelungstechnik die den Meßgrößen überlagerten Rauschanteile unerwünscht sind und durch geeignete Filter unterdrückt werden, nutzt man bei der Rauschanalyse gerade diese Signalanteile aus um weitere Informationen über das zu untersuchende System zu erhalten. Als besonders leistungsfähig haben sich dabei Kreuzkorrelationsanalysen zwischen verschiedenen Meßgrößen wie z.B. zwischen Neutronenfluß und Brennelementaustrittstemperaturen erwiesen /12/.

Die Charakterisierung und Analyse von Rauschsignalen erfolgt allgemein durch Berechnung verschiedener statistischer Kenngrößen /7/. Hierzu gehören:

- Autokorrelationsfunktion
- Spektrale Leistungsdichte
- Amplitudenverteilung

Zwei Rauschsignale werden entsprechend durch folgende statistische Kenngrößen miteinander verknüpft:

- Kreuzkorrelationsfunktion
- Spektrale Kreuzleistungsdichte
- Zweidimensionale Amplitudenverteilung



Aus diesen statistischen Kenngrößen können dann wiederum wichtige Systemeigenschaften, wie z.B. Übertragungsfunktionen, bestimmt werden.

Zur Ermittlung der statistischen Kenngrößen von Rauschsignalen sind aufwendige Auswertemethoden notwendig. Entsprechend der Zielsetzung bei der meßtechnischen Analyse des Reaktorrauschens gibt es verschiedene Konzepte:

- a.) Unmittelbare Analyse der Meßdaten im real-time-Verfahren, mit dem Ziel, relevante Informationen über den Betriebszustand des Reaktors zu gewinnen.  
(z.B. für Abschaltkriterien oder zur Regelung.)
- b.) Breitbandige Aufzeichnung der Meßdaten auf Magnetband (analog oder digital) zur späteren Auswertung.

Die Anforderungen an die für den Fall a.) benötigten real-time-Analysatoren werden im wesentlichen bestimmt durch:

- 1.) Anzahl der zu analysierenden Meßkanäle.
- 2.) höchste Analysierfrequenz und Frequenzauflösung.
- 3.) Komplexität der Analyseverfahren.

Dabei können sowohl spezielle real-time Analysatoren /4/, als auch Kleinrechnersysteme zur digitalen Signalanalyse in Betracht gezogen werden.

Wenn eine unmittelbare Auswertung der Meßdaten nicht notwendig ist, wird im allgemeinen nach Methode b.) verfahren. Hierbei ist es möglich, verschiedene Auswertemethoden anzuwenden und zu vergleichen, sowie Korrelationsmessungen mit allen sinnvollen Signalkombinationen durchzuführen. Voraussetzung hierfür ist jedoch, daß alle Signale mit ausreichender Bandbreite aufgezeichnet sind. Die nachträgliche Auswertung von Meßsignalen dient dabei folgenden Zielen:

- 1.) Messung wichtiger Systemparameter wie z.B. Übertragungsfunktionen /5/.

- 2.) Qualitative und quantitative Identifizierung von Rauschquellen /12/.
- 3.) Entwicklung und Erprobung von Überwachungsmethoden und geeigneten real-time Analysatoren für die unter a.) genannten Aufgaben.

Für die nachträgliche Analyse von Meßdaten, die auf Magnetbändern gespeichert sind, gibt es zwei Konzepte:

- 1.) Auswertung der auf Digitalbändern gespeicherten Meßdaten in einem Rechenzentrum mit einem speziellen Programmsystem wie z.B. SEDAP, /6/ (off-line Analyse).
- 2.) Auswertung der auf Analogbändern vorliegenden Meßdaten mit einem als on-line Analysator verwendeten Kleinrechnersystem.

Beide Konzepte haben Vor- und Nachteile, die hier jedoch nicht im einzelnen diskutiert werden sollen. Eine Entscheidung für das on-line Verfahren ist jedoch immer dann sinnvoll, wenn die Entwicklung und Erprobung von Überwachungsmethoden und Nachweissystemen angestrebt wird.

Für die Analyse von Rauschsignalen, sowie für die Erprobung neuer Meß- und Nachweisverfahren wurde NOASYS, ein Noise Analysis System entwickelt. Es handelt sich dabei um ein Kleinrechner-System zur digitalen on-line Signalverarbeitung, das sowohl für die unmittelbare Auswertung von Meßdaten am Reaktor, als auch für die nachträgliche Auswertung von gespeicherten Meßdaten gut geeignet ist.

Zur Zeit stehen zwei Systeme (NOASYS-1 und NOASYS-2) zur Verfügung, die sich im wesentlichen durch unterschiedliche Hardwarekomponenten unterscheiden. Der vorliegende Bericht gibt einen kurzen Überblick über die Besonderheiten der digitalen Signalanalyse, sowie eine Beschreibung der beiden Signalanalysatoren NOASYS-1 und NOASYS-2.

## 2. Digitale Signalanalyse

Die mathematischen Methoden zur statistischen Beschreibung von Rausch-

signalen, sowie eine Definition ihrer Kenngrößen ist in /7/ zu finden. Die Ermittlung dieser Kenngrößen in NOASYS erfolgt digital mit Hilfe eines Kleinrechners. Bei der digitalen Signalanalyse können im Gegensatz zur Analogtechnik nur zeit- und amplitudendiskrete Meßwerte verarbeitet werden. Die Umwandlung der Analogsignale  $x(t)$  in zeit- und amplitudendiskrete Signale  $x(n\Delta t)$  erfolgt mit einem Analog-zu-Digital Konverter (ADC) jeweils zum Zeitpunkt  $n\Delta t$ .

$$\Delta t = \text{Abtastperiode} ; f_s = 1/\Delta t$$
$$f_s = \text{Abtastfrequenz}$$

Die Konsequenzen, die sich aus der Zeit- und Amplitudendiskretisierung ergeben werden in /9/ ausführlich behandelt. Das durch die Quantisierung der Amplituden erzeugte Quantisierungsrauschen kann durch die Verwendung eines hochauflösenden ADC (12 bit) vernachlässigt werden. Der Übergang von kontinuierlichen zu zeitdiskreten Signalen, sowie die Verarbeitung endlicher Zeitabschnitte hat jedoch weitaus größere Konsequenzen, hierzu gehören folgende Effekte:

- a.) aliasing
- b.) unerwünschte Faltungsprodukte (leakage)
- c.) zyklische Korrelationen

Der unter dem Namen "aliasing" bekannte Effekt tritt auf, wenn ein Signal Spektralanteile mit Frequenzen  $f > f_s/2$  enthält. Solche Spektralanteile sind aufgrund des Abtasttheorems nicht mehr von Frequenzen  $f < f_s/2$  zu unterscheiden. Um dies zu vermeiden, müssen alle Frequenzanteile oberhalb von  $f_s/2$  durch geeignete Tiefpassfilter unterdrückt werden.

Die Begriffe "zyklische Korrelationen" und "unerwünschte Faltungsprodukte" beschreiben Effekte gleicher Ursache im Zeitbereich (c) und im Frequenzbereich (b). Ihre gemeinsame Ursache ist die Fouriertransformation endlicher, in den Transformationsintervallen nicht periodischer

Funktionen. Durch geeignete Analyseverfahren (siehe Abschnitt 2.1 und 2.2) können diese Effekte ausreichend unterdrückt werden.

Während zur Ermittlung von Varianz und Amplitudenverteilung nur ein relativ geringer Aufwand benötigt wird, gilt dies nicht mehr für die spektrale Leistungsdichte (SLD) und die Korrelationsfunktion (KF). Es handelt sich dabei um gleichwertige Kenngrößen zur Beschreibung von Rauschsignalen im Frequenzbereich oder im Zeitbereich, die über die Fouriertransformation eindeutig miteinander verknüpft sind. Zur Ermittlung von Korrelationsfunktion und spektraler Leistungsdichte sind zwei Analyseverfahren gebräuchlich:

1. Korrelationsmethode
2. Direktrtransformation

## 2.1 Korrelationsmethode

Bei dieser Methode wird die Korrelationsfunktion unmittelbar aus den zeitdiskreten Meßsignalen  $x(m\Delta t)$  und  $y(m\Delta t)$  berechnet. Für eine Kreuzkorrelationsfunktion wird dabei folgender Algorithmus angewandt:

$$R(n\Delta t) = \frac{1}{M-N+1} \sum_{m=N/2}^{M-N/2} x(m\Delta t) \cdot y((m+n)\Delta t) \quad (1)$$

mit  $-N/2 \leq n < N/2$  und  $M \gg N$

Ersetzt man in (1) das Signal  $y(m\Delta t)$  durch  $x(m\Delta t)$  so erhält man die Autokorrelationsfunktion (AKF), für die gilt:

$$R(n\Delta t) = R(-n\Delta t)$$

Für eine AKF genügt es daher die Verzögerungszeiten  $0 \leq n\Delta t < N/2 \cdot \Delta t$  zu berechnen.

Durch diskrete Fouriertransformation (DFT) von (1) erhält man die spektrale Kreuzleistungsdichte (SKLD)

$$S(k\Delta f) = \frac{1}{N\Delta f} \sum_{n=-N/2}^{N/2-1} R(n\Delta t) \exp\left(-i\frac{2\pi nk}{N}\right) \quad (2)$$

mit  $k = 0, 1, 2, \dots, N/2$

und  $\Delta f = 1/(N\Delta t)$

Da eine Korrelationsfunktion in der Praxis nur für endliche Verzögerungszeiten  $n\Delta t$ , ( $-N/2 \leq n < N/2$ ), gemessen werden kann, ist dies gleichbedeutend mit der Multiplikation der vollständigen Korrelationsfunktion  $R(n\Delta t)$ , ( $-\infty < n < \infty$ ), mit dem Rechteckfenster.

$$w_R(n\Delta t) = \begin{cases} 0 & \text{für } n < -N/2 \text{ und } n \geq N/2 \\ 1 & \text{für } -N/2 \leq n < N/2 \end{cases} \quad (3)$$

Einer Multiplikation im Zeitbereich entspricht jedoch eine Faltung im Frequenzbereich. Das Resultat dieser Faltung führt zu einer unerwünschten gegenseitigen Beeinflussung von Spektralanteilen (leakage).

Eine Abschwächung dieses Effektes kann durch die Verwendung geeigneter Fensterfunktionen erreicht werden. Die Auswirkung solcher Fensterfunktionen auf die SLD besteht in einer Verringerung der gegenseitigen Beeinflussung entfernter Spektralanteile und einer Zunahme der Analysierbandbreite. In der Praxis werden verschiedene Fensterfunktionen verwendet, wobei ein Kompromiß zwischen den oben genannten Einflüssen angestrebt wird. Häufig verwendet wird das sogenannte Hanning-Fenster, das auch in NOASYS zur Verfügung steht. Es ist wie folgt definiert:

$$w_H(n\Delta t) = \begin{cases} 0 & \text{für } n < -N/2 \text{ und } n \geq N/2 \\ (1 + \cos 2\pi n/N)/2 & \text{für } -N/2 \leq n < N/2 \end{cases}$$

Zur Ermittlung der SLD muß bei der Korrelationsmethode nur die Korrelationsfunktion schritthaltend mit der Datenerfassung berechnet werden. Dies erfordert  $N$  Multiplikationen und  $N$  Additionen innerhalb einer Abtastperiode  $\Delta t$ . Die zur Ermittlung der spektralen Leistungsdichte erforderliche DFT ist dann nicht mehr zeitkritisch, es sei denn die SLD wird in kurzen Zeitabständen für Nachweissysteme benötigt.

Aufgrund des einfachen Rechenalgorithmus werden zur Ermittlung der Korrelationsfunktion häufig Hardware-Korrelatoren verwendet. Ein System aus Korrelator und Kleinrechner wurde bereits als Vorgänger zu NOASYS für die Messung der spektralen Leistungsdichte eingesetzt. Für solche Systeme wird die maximale Abtastfrequenz nur von der Verarbeitungsgeschwindigkeit des Korrelators bestimmt. An den Kleinrechner werden dabei keine großen Anforderungen gestellt, da die Berechnung der SLD aus der Korrelationsfunktion nicht mehr schritthaltend mit der Datenerfassung erfolgen muß. Die Analyse einer größeren Anzahl von Signalen ist nach dieser Methode jedoch nicht mehr sinnvoll, da mehrere Korrelatoren benötigt würden.

## 2.2 Direkttransformation

Bei dieser Methode wird die spektrale Leistungsdichte unmittelbar aus den Zeitsignalen  $x(n\Delta t)$  und  $y(n\Delta t)$  ermittelt, während die Korrelationsfunktion anschließend aus der SLD berechnet wird.

Die Verarbeitung der Meßdaten erfolgt bei der Direkttransformation in Blöcken von  $N$  Meßwerten. Jeder Datenblock repräsentiert dabei einen Signalausschnitt der Länge  $N\Delta t$ . Durch diskrete Fouriertransformation dieser Blöcke erhält man die linearen Spektren  $X(k\Delta f)$  und  $Y(k\Delta f)$ .

$$X(k\Delta f) = \frac{1}{N} \sum_{n=0}^{N-1} x(n\Delta t) \exp(-i \frac{2\pi nk}{N}) \quad (6)$$

Durch konjugiert komplexe Multiplikation und Mittelwertbildung über L Spektren erhält man dann die spektrale Kreuzleistungsdichte (SKLD)

$$S(k\Delta f) = \frac{1}{L\Delta f} \sum_{m=0}^{L-1} X(k\Delta f) \cdot Y^*(k\Delta f) \quad (7)$$

mit  $\Delta f = 1/N\Delta t$  ;  $k = 0, 1, 2, \dots, N/2$

und einer Meßzeit  $T = L \cdot N \cdot \Delta t$

Für  $y(n) = x(n)$  erhält man die spektrale Autoleistungsdichte (SALD).

Ein Vergleich der Algorithmen von Korrelationsfunktion (1) und diskreter Fouriertransformation (6) zeigt, daß für die Zahl der Rechenoperationen Z pro Abtastperiode  $\Delta t$  in beiden Fällen gilt:  $Z \sim N$ .

Erst die Entwicklung des Fast Fourier Transform (FFT) Algorithmus /8/ brachte der Direkttransformationsmethode einen entscheidenden Vorteil. Beim FFT-Algorithmus steigt die Zahl der Rechenoperationen für eine N-Punkte DFT nur noch proportional  $N \log_2 N$  und damit bezogen auf  $\Delta t$  mit  $\log_2 N$ . Aufgrund dieser Vorteile ist es sinnvoll auch Korrelationsfunktionen über die Direkttransformations-Methode zu ermitteln, da dies in NOASYS bereits für  $N > 64$  kürzere Rechenzeiten ergibt.

Die bei der Transformation endlicher Signalstücke auftretenden unerwünschten Faltungsprodukte können auch bei der Direkttransformation durch Anwendung von Fensterfunktionen abgeschwächt werden. Dabei ergibt sich eine Erhöhung der Analysierbandbreite  $B = 1/(N\Delta t)$  um den Faktor a. Die effektive Analysierbandbreite  $B_e = Ba$  ist dabei abhängig von der Fensterfunktion, mit der die einzelnen Signalstücke gewichtet werden.

Bei dieser Wichtung geht ein Teil der Information verloren, so daß zur Bestimmung des Fehlers eines mit der effektiven Bandbreite  $B_e$  gemessenen Spektralwerts weiterhin gilt:

$$E = \frac{1}{\sqrt{B \cdot T}} \quad (8)$$

$T = \text{Meßzeit}$  ;  $B = 1/(N\Delta t)$

Erst bei einer überlappenden Signalverarbeitung, bei der dieser Informationsverlust ausgeglichen wird, kann für die Fehlerformel (8) die effektive Bandbreite  $B_e$  verwendet werden. NOASYS bietet daher die Möglichkeit, Meßsignale um jeweils eine halbe Blocklänge überlappend zu verarbeiten. Aufgrund der günstigen Eigenschaften des FFT-Algorithmus ist es für große Werte von  $N$  vorteilhaft, auch die Korrelationsfunktion über eine Frequenzbereichsanalyse aus der SLD zu berechnen. Diese unter dem Namen Fast Correlation /9/ bekannte Methode erfordert jedoch eine besondere Technik der überlappenden Signalverarbeitung, um die sogenannten zyklischen Korrelationen zu vermeiden. Zyklische Korrelationen sind dadurch bedingt, daß die bei einer DFT ermittelten Fourierkoeffizienten für eine mit  $N\Delta t$  periodische Zeitfunktion gelten. Um zyklische Korrelationen zu vermeiden, wird aus dem um eine Blockhälfte überlappenden Signal  $x_a(n\Delta t)$  ein zweites Signal  $x_b(n\Delta t)$  gewonnen, bei dem die 2. Blockhälfte mit Nullen aufgefüllt wird. Aus  $x_a$  und  $x_b$  wird dann die SKLD ermittelt. Nach einer inversen DFT der gemittelten SKLD erhält man dann die Korrelationsfunktion. Die erste Blockhälfte enthält dabei die korrekten Funktionswerte  $R(n\Delta t)$  mit  $0 \leq n \leq N/2$ , während die 2. Blockhälfte die unerwünschten zyklischen Korrelationsprodukte enthält.

Die gleiche Methode kann auch zur Realisierung nichtrekursiver digitaler Filter angewandt werden. Hierbei wird die Faltung der Zeitfunktion mit der gewünschten Impulsantwortfunktion des Filters im Frequenzbereich durchgeführt (Fast Convolution). Die Technik der überlappenden Signalverarbeitung ist dabei die gleiche wie bei der Ermittlung der Korrelationsfunktion.

### 3. Systembeschreibung

Bei NOASYS-1 und NOASYS-2 handelt es sich um Kleinrechnersysteme zur on-line Signalanalyse. Das ältere System NOASYS-1 ist mit dem Rechner HP2116B ausgerüstet und kann 4 Meßsignale verarbeiten /10/. Mit NOASYS-2 können maximal 16 Signale verarbeitet werden, als Rechner wird hier ein HP2100A eingesetzt. Rechenzeitintensive Verarbeitungsroutinen wie z.B. der FFT-Algorithmus werden in beiden Systemen nach unterschiedlichen Methoden ausgeführt. Der HP2116B wurde durch eine spezielle FFT-Arithmetik Unit erweitert,



während für den HP2100A entsprechende Mikroprogramme erstellt wurden /11/. Abb.1 zeigt das Zusammenwirken der verschiedenen System-Komponenten von NOASYS. Die Software für beide Systeme ist weitgehend indentisch, Abweichungen ergeben sich lediglich aus der Verwendung unterschiedlicher Hardware-Komponenten. Die Kommunikation mit NOASYS erfolgt über eine Teletype und eine Steuerkonsole. Der Benutzer hat dabei die Möglichkeit, spezielle Analyseprogramme aus einem Grundvorrat von Tasks zusammenzustellen. Die Tasks lassen sich entsprechend ihrer Funktion in folgende Gruppen unterteilen:

1. Steuerprogramme zur Datenerfassung
2. Ein-Ausgabeprogramme
3. Koordinierungsprogramme
4. Rechenoperationen mit Datenblöcken

Durch Integration zusätzlicher Tasks hat jeder Benutzer die Möglichkeit, NOASYS zu erweitern und an seine speziellen Bedürfnisse anzupassen (siehe Abschnitt 4.). Häufig benötigte Tasks, wie z.B. die Sichtgeräteausgabe, können von einer Steuerkonsole über Schalter und Tasten mit Parametern versorgt und gestartet werden.

Bei der Analyse im Frequenzbereich erfolgt die Verarbeitung der Meßdaten blockweise, jeder Datenblock enthält dabei jeweils einen Signalausschnitt der Länge  $N\Delta t$ . Die dabei am häufigsten benötigten Verarbeitungsroutinen sind:

- a.) diskrete Fouriertransformation (FFT-Algorithmus)
- b.) reelle und komplexe Addition und Multiplikation von Datenblöcken.

Die Häufigkeit mit der die einzelnen Verarbeitungsroutinen während einer Blockfüllzeit  $N\Delta t$  benötigt werden, ist abhängig vom Analyseverfahren und von der Anzahl der gleichzeitig zu analysierenden Signale. Wird eine lückenlose Verarbeitung der Meßsignale angestrebt, so darf die gesamte dafür benötigte Rechenzeit den Wert  $N\Delta t$  nicht überschreiten. Die maximale Abtast-

Frequenz  $f_s$  wird daher im wesentlichen von der Rechenzeit der oben genannten Verarbeitungsroutinen bestimmt. Die größte Rechenzeit wird für den FFT-Algorithmus zur diskreten Fouriertransformation benötigt. Die Leistungsfähigkeit eines on-line Analysators wird daher hauptsächlich von der für eine FFT benötigten Rechenzeit bestimmt. Bei Kleinrechnersystemen wie z.B. NOASYS gibt es dafür folgende Möglichkeiten:

- 1.) Assemblerprogrammierung des FFT-Algorithmus (1s)
- 2.) Mikroprogrammierung des FFT-Algorithmus (160 ms)
- 3.) Erweiterung des Rechners durch eine FFT-Arithmetik-Unit (230 ms)
- 4.) Anschluß eines peripheren FFT-Prozessors (15 ms)

Die in Klammern angegebenen Rechenzeiten beziehen sich jeweils auf eine DFT von 1024 reellen Funktionswerten und gelten für die Rechner HP2100A bzw. HP2116B. Die mit den Methoden 2 und 3 erzielten Rechenzeiten von etwa 200 ms für eine 1024-Punkte DFT sind für die meisten Anwendungen von NOASYS ausreichend. Der für NOASYS-1 verwendete Rechner HP2116B ist nicht mikroprogrammierbar und wurde daher mit einer FFT-Arithmetik-Unit ausgerüstet. Es handelt sich dabei um ein spezielles Rechenwerk für komplexe Rechenoperationen, welches 4 I/O-Kartenplätze im Rechner belegt. Der für NOASYS-2 verwendete Rechner HP2100A ist mikroprogrammierbar, so daß hier auf eine zusätzliche FFT-Hardware verzichtet werden konnte. Für die FFT, sowie für andere häufig benötigte Auswerterroutinen, wurden entsprechende Mikroprogramme erstellt /11/.

### 3.1 Hardware

#### 3.1.1 Rechner und Peripheriegeräte

Für die beiden Rauschanalysatoren NOASYS-1 und NOASYS-2 werden Kleinrechner vom Typ HP2116B und HP2100A eingesetzt. Die wichtigsten Daten dieser Kleinrechner sind in der folgenden Tabelle kurz zusammengefaßt.

	HP2116B	HP2100A
Wortlänge	16 bit	16 bit
Speichergröße	16K (max. 32K)	24K (max. 32K)
Zykluszeit	1,6 $\mu$ s	0,98 $\mu$ s
Zeiten für:	Festkomma / Gleitkomma	Festkomma / Gleitkomma
a.) Addition	3,2 $\mu$ s / 500 $\mu$ s	1,96 $\mu$ s / 24-60 $\mu$ s
b.) Multiplikation	20 $\mu$ s / 700 $\mu$ s	10,7 $\mu$ s / 33-41 $\mu$ s
DMA-Transfer-Rate	$5 \cdot 10^5$ Worte / s	$10^6$ Worte / s
I/O-Kanäle (intern verdrahtet)	16	14
Befehlsausführung	verdrahtete Logik	per Mikroprogramm
Mikrobefehlszeit	—	0,2 $\mu$ s

Die Konfiguration der beiden Systeme mit den wichtigsten Daten der Peripheriegeräte kann aus den Abbildungen 2 und 3 entnommen werden.

### 3.1.2 Die Meßdatenerfassung

Die analogen Meßsignale werden in einer Meßdatenerfassungseinheit (MDE) in zeit- und amplitudendiskrete Signale umgewandelt und in den Kernspeicher des Rechners übertragen. Die Eingangsstufen der MDE sind mit Sample+Hold-Stufen ausgerüstet, um eine simultane Abtastung der Meßsignale zu erreichen. Die Abtastfrequenz ist manuell oder über den Rechner einstellbar und kann maximal 100 kHz betragen. Die Datenwandlung erfolgt mit einem 12-bit ADC, der über einen Multiplexer mit den 16 Eingangsstufen verbunden ist. Die konvertierten Meßdaten werden über einen modifizierten DMA-Kanal direkt in den Speicher des Rechners übertragen. Die Daten werden dabei bereits während der Übertragung nach Kanalnummern sortiert und in getrennten Speicherblöcken abgelegt. Neben den 16 Analogkanälen werden noch 64 Digitaleingänge benötigt, um die Schalterstellungen der MDE und

der Steuer-Konsole abzufragen.

Abb. 4 zeigt das Blockschaltbild der MDE für NOASYS-2, deren wichtigsten Daten in der folgenden Tabelle zusammengefaßt sind:

Eingangskanäle:	a.) 8 Kanäle mit Sample+Hold-Stufen b.) 8 Kanäle ohne " " "
Eingangsbereiche:	a.) 8 V , 4 V , 2 V , 1 V b.) nur 4 V
Kopplung:	DC-Kopplung
Eingangswiderstand:	100 K $\Omega$
Datenwandlungsrate:	100 kHz/k , k = Anzahl der Kanäle;
Abtastfrequenz (intern):	$2^0, 2^1, \dots, 2^{15}$ Hz manuell oder über den Rechner einstellbar;
Abtastfrequenz (extern):	maximal 100 kHz/k , TTL-Pegel;
ADC-Kodierung:	12-bit , 2-er-Komplement;
Digitaleingänge:	30 für Abfrage der Steuer-Konsole; 30 für Bereichsabfrage der MDE; 4 für Abfrage der Abtastfrequenz;
Anzeigen für:	a.) ADC-Daten, bit 0 - 11 b.) Triggerpegelüberschreitung c.) Start Messung; e.) Übersteuerungsanzeige für jeden Kanal; f.) Abtastfrequenz zu hoch;

Die Datenwandlung in der MDE ist ständig aktiv, auch wenn kein Datentransfer zum Rechner stattfindet. Dadurch kann die Aussteuerung der einzelnen Kanäle jederzeit über den ADC-Display bzw. über die Übersteuerungsanzeigen überprüft werden. Der Datentransfer zwischen MDE und Kernspeicher wird durch Initialisierung des DMA-Kanals vorbereitet und kann wie folgt gestartet werden:

1.) TRIGGER AUS:

Der Datentransfer wird jeweils vom Rechner gestartet, wenn ein Pufferbereich zur Füllung freigegeben wurde.

2.) TRIGGER EIN:

Der Datentransfer kann jeweils durch ein Triggersignal gestartet werden nachdem ein Pufferbereich zur Füllung freigegeben wurde.

Triggerquelle: a.) Intern von Kanal 1

b.) Extern über BNC-Buchse

Der Triggerpegel kann zwischen -5V und +5V eingestellt werden.

3.2 NOASYS-Software

Die NOASYS-Software wurde ursprünglich für das lochstreifenorientierte Betriebssystem BCS entwickelt. Nachdem beide Rechner mit einem Plattenspeicher ausgerüstet waren, konnte die NOASYS-Software auf das Disc Operating System (DOS) umgestellt werden. Die NOASYS-Software besteht aus folgenden Komponenten:

- a.) Betriebssystem (DOS-M)
- b.) NOASYS-Monitor
- c.) NOASYS-Tasks
- d.) NOASYS-Unterprogramme
- e.) Standard-Unterprogramme

Abb. 5 zeigt die Speicherbelegung für NOASYS. Der speicherresidente Teil des Betriebssystems belegt etwa 5 K, für den NOASYS-Kern werden etwa 7 K benötigt, der Rest steht als Datenbereich zur Verfügung.

### 3.2.1 Monitor und Task-Liste

Die Kommunikation zwischen einem Benutzer und NOASYS erfolgt über den Monitor. Es handelt sich dabei um ein übergeordnetes Steuerprogramm zur Koordination und Parameterversorgung von Tasks. Der Benutzer kann mit Hilfe des Monitors aus einem Vorrat elementarer Tasks aufgabenspezifische Meßprogramme zusammenstellen und steuern. Abb. 6 zeigt ein vereinfachtes Flußdiagramm des Monitors. Der Monitor besitzt einen Task-Puffer für die Startadressen und Parametersätze von NOASYS-Tasks. Die Eingabe von Task-Parametern in diesen Puffer erfolgt über die Teletype oder den Lochstreifenleser mit folgendem Format:

Nummer, Name, P1, P2, ... , P5

#### Nummer:

Es handelt sich um eine Sequenznummer, die die Position der Task im Task-Puffer bestimmt. Durch die Sequenznummer wird die Ablaufreihenfolge einer Gruppe von Tasks, deren Parameter im Task-Puffer abgelegt sind, festgelegt. Die Sequenznummer darf Werte von 1 bis 50 annehmen, die Reihenfolge der Eingabe ist beliebig.

#### Name:

Es handelt sich um den Namen einer Task. Im Task-Puffer wird nur die Startadresse dieser Task zusammen mit maximal 5 Parametern abgelegt.

### P1 bis P5

Bei den Parametern P1 bis P5 handelt es sich um 16-bit Integer-Zahlen, die der Parameterliste der entsprechenden Task zugeordnet werden.

Die im Task-Puffer eingetragenen Tasks werden vom Monitor sequentiell aufgerufen. Eine Task kann dabei auch mehrfach mit unterschiedlichen Parametern in den Task-Puffer eingetragen werden. Die Reihenfolge der Eintragung ist beliebig, leere Plätze werden übersprungen.

Das Löschen einer Task erfolgt durch Eingabe der entsprechenden Sequenznummer. Erfolgt eine Task-Eingabe ohne Sequenznummer, so wird diese Task sofort aufgerufen (direkter Aufruf).

Der Monitor überprüft nach jeder Eingabe das Format und sucht anschließend in der Task-Liste die Startadresse der betreffenden Task. Bei einem Eingabefehler, oder wenn der Name nicht in der Task-Liste gefunden wird, druckt der Monitor die betreffende Eingabezeile mit einem vorangestellten Fragezeichen auf der Teletype aus. Eine Zurückweisung der Eingabe hat in der Regel folgende Gründe:

1. Der Task-Name ist nicht in der Task-Liste vereinbart.
2. Die betreffende Task wurde bei der Systemgenerierung durch den DOS-Loader nicht mitgeladen.
3. Formaler Eingabefehler

### 3.2.2 Tasks

Bei den NOASYS-Tasks handelt es sich um Assembler oder Fortran-Subroutinen, deren Name vor der Systemgenerierung in einer Task-Liste vereinbart wurde. NOASYS-Tasks können daher von einem Benutzer über die System-Konsole einzeln aufgerufen oder zu Meßprogrammen aneinandergeschaltet werden. Es gibt zwei Task-Listen:

1. Standart-Task-Liste
2. Erweiterungs-Liste

In der Standart-Task-Liste sind alle Taskvereinbart, die für ein Standart System benötigt werden. Eine Beschreibung der Standart-Tasks erfolgt in Abschnitt 3.4. Mit der Erweiterungs-Liste hat ein Benutzer die Möglichkeit, seine eigenen Tasks zu vereinbaren und in ein System zu integrieren (siehe Abschnitt 4.).

### 3.2.3 Sichtgerät und Steuerkonsole

Zur grafischen Darstellung einzelner Datenblöcke wird ein Sichtgerät mit variabler Nachleuchtdauer (0.1 - 10 s) verwendet. In Zeiten hoher Rechnerauslastung kann daher mit niedrigen Bildwiederholfrequenzen gearbeitet werden. Jedes Datenwort wird als Punkt in einem X-Y-Koordinatensystem dargestellt und kann durch Helltastung markiert werden. Die Daten des markierten Bildpunktes (z.B. Frequenz [Hz] u. SLD [ $V^2s$ ]) werden numerisch in das Bild eingeblendet.

Die Software für das Sichtgerät wurde so gestaltet, daß die Eingabe aller wichtigen Parameter zur grafischen Darstellung von Datenblöcken über eine Steuerkonsole möglich ist. Die Parameter für die Sichtgeräte-Task, die über die Steuerkonsole (Abb. 10) eingestellt werden können, haben folgende Bedeutung:

#### BLOCKNUMMER:

Nummer des Datenblockes, der auf dem Sichtgerät dargestellt werden soll.

#### OFFSET:

Lage der Nulllinie auf dem Bildschirm.



SCALE:

Skalierungsfaktor ( $2^0$  bis  $2^7$ ) für die Sichtgerätedarstellung.

R/B; Z; I/P:

Auswahl des Datentyps:

Z = Zeitbereichsdaten

R/B = Realteil bzw. Betrag

I/P = Imaginärteil bzw. Phase

LINKS, RECHTS:

Positionsveränderung des Helltastpunktes zur Markierung eines Datenwortes.

REQUEST:

Numerische Anzeige des markierten Datenworts auf dem Sichtgerät. Die Anzeige hat folgendes Format:

$$F = \dots \text{ HZ} \quad Y = \dots$$

Der Zahlenwert für F wird wie folgt berechnet:

$$F = n / (N \Delta t)$$

mit  $n = 0, 1, 2, \dots, N-1$  (Position des Helltastpunktes).

Für Y gilt:

$$Y = y(n) \cdot 2^{\text{SF}}$$

SF = Skalierungsfaktor (siehe Abschnitt 3.3).

/ΔF :

Der angezeigte Y-Wert wird mit dem Faktor  $2 \cdot N \Delta t$  multipliziert, (Normierung eines Spektrums auf die Analysierbandbreite  $\Delta f$ ).

Bei der Anwendung von Fensterfunktionen muß Y mit einem Korrekturfaktor k multipliziert werden. Für ein Hanning-Fenster bei Direkttransformation gilt  $k = 8/3$ .

Die restlichen Schalter und Taster auf der Steuerkonsole dienen zur Steuerung und Parameterversorgung häufig benötigter NOASYS-Tasks und haben folgende Bedeutung:

ERASE:

Der Inhalt des auf dem Sichtgerät dargestellten Datenblocks wird durch Aufruf der Task ERA gelöscht.

RUN:

Die Task RUN wird aufgerufen (starke Datenakkumulation). Der beim letzten Teletype-Aufruf von RUN verwendete Parameter a bleibt gültig (siehe Abschnitt 3.4.2).

STOP:

Eine durch die Task RUN gestartete Datenakkumulation wird vorzeitig beendet.

ZEITKONSTANTE:

Eingabe der Zeitkonstanten für die exponentielle Mittelwertbildung (siehe Task CPSD in Abschnitt 3.4.4).

### Anzeige-Lampe L1:

Sobald L1 aufleuchtet ist eine lückenlose Datenerfassung zwischen den einzelnen Blockfüllungen nicht mehr gewährleistet: Dies bedeutet, daß eine Auswertesequenz mehr Zeit benötigt als eine Blockfüllung.

Die Schalter S0-S3 und die Lampe L2 werden nicht verwendet.

### 3.3 NOASYS-Datenblöcke

Der NOASYS-Datenbereich wird beim Aufruf der Task MESS, BL in gleichgroße Datenblöcke der Länge BL unterteilt, (BL = 16, 32, ... , 2048). Jedem Datenblock, der durch seine Blocknummer (BNR) adressierbar ist, sind jeweils noch 2 Statusworte (ST1, ST2) und ein Skalierungsfaktor (SF) zugeordnet.

Es gibt 3 Typen von Datenblöcken:

- a.) Einfachblöcke, 16-bit Wortlänge
- b.) Einfachblöcke, 32-bit " (nur für SALD)
- c.) Doppelblöcke, 32-bit "

Bei Doppelblöcken enthält der erste Block die most significant bit's (msb's) und der zweite Block enthält die least significant bit's (lsb's). Bei Einfachblöcken vom Type b.) sind die msb's und die lsb's paarweise angeordnet.

Der Skalierungsfaktor ist ein Exponent zur Basis 2 und gilt für alle Datenworte eines Blockes. Der Wert W eines Datenwortes berechnet sich wie folgt:

$$W = w \cdot 2^{SF}$$

$$w = 16\text{-bit Integerwort} ; \quad -2^{15} \leq w < 2^{15}$$

Bei 32-bit Worten gilt:

$$w = w_m + 2^{-16} \cdot w_e$$

$w_m$  = 16-bit Integerwort (msb's)

$w_e$  = " " (lsb's)

Negative Zahlen werden im 2-er-Komplement dargestellt.

Bei Doppelblöcken sind jeweils SF, ST1 und ST2 des ersten Datenblocks maßgebend.

Das Statuswort ST2 wird als Zähler in Task's verwendet, in denen zwecks Mittelwertbildung mehrere Datenblöcke aufsummiert werden. Das Statuswort ST1 gibt Auskunft über Ursprung und Verarbeitungsmerkmale eines Datenblocks. Die einzelnen bits von ST1 haben folgende Bedeutung:

Bit 15, 14	: 0 0	Zahl der führenden Nullen $\geq 2$
	0 1	" " " " = 1
	1 x	" " " " = 0
Bit 13	: 0	Zeitbereichsdaten
	1	Frequenzbereichsdaten
Bit 12	: 0	Transformation ohne Hanning
	1	" mit "
Bit 11	: 0	Realteil / Imaginärteil
	1	Betrag / Phase
Bit 10	: 0	SKLD
	1	SALD
Bit 9, 8	: 0 0	Einfachblock, 16-bit
	0 1	Einfachblock, 32-bit
	1 0	Doppelblock, 32-bit
Bit 7 - 4	frei	
Bit 4 - 0	—	Status von 5471A-DOS Driver

x = 0 oder 1

Zeitbereichsdaten sind in natürlicher Reihenfolge in den NOASYS-Datenblöcken angeordnet. Die Anordnung der Frequenzbereichsdaten erfolgt in NOASYS-1 und NOASYS-2 unterschiedlich. Real- und Imaginärteil stehen bei beiden Systemen paarweise in zwei aufeinanderfolgenden Speicherzellen. In NOASYS-2 sind die Fourierkoeffizienten in natürlicher Reihenfolge angeordnet, so daß ihr Index einfach aus der Speicheradresse abgeleitet werden kann. Es besteht folgende Anordnung:

$$R_0, R_{N/2}, R_1, I_1, R_2, I_2, \dots, R_{N/2-1}, I_{N/2-1}$$

Für Betrag und Phase gilt:

$$B_0, O, B_1, P_1, B_2, P_2, \dots, B_{N/2-1}, P_{N/2-1}$$

R = Realteil ; I = Imaginärteil

B = Betrag ; P = Phase

In NOASYS-1 sind die Fourierkoeffizienten, bedingt durch die Verwendung der FFT-Arithmetik Unit, nach einer DFT ungeordnet. Zur Umrechnung der Frequenzindizes in die jeweiligen Speicheradressen kann die Subroutine "III71" verwendet werden.

Bei einem Zugriff zu den NOASYS-Datenblöcken mit den Subroutinen GET, SAVE und GETDB wird die unterschiedliche Anordnung der Frequenzbereichsdaten zwischen NOASYS-1 und -2 automatisch berücksichtigt.

### 3.3.1 Zugriff zu NOASYS-Datenblöcken

#### a.) direkter Zugriff

Der direkte Zugriff zu NOASYS-Datenblöcken ist nur für Tasks möglich. Bei

jedem Aufruf einer Task durch den MONITOR wird als 6. Parameter die Basisadresse des NOASYS-Datenbereichs übertragen. Es handelt sich dabei um die Adresse des 1. Wortes von Blocknummer 1. Dadurch ist ein direkter Zugriff zu allen NOASYS-Datenblöcken möglich. Die unterschiedliche Anordnung der Frequenzbereichsdaten muß dabei berücksichtigt werden. Der Zugriff zu Skalierungsfaktoren und zu weiteren NOASYS-Parametern kann mit der Subroutine

```
GIVE (IBNR, N, SFX, SFY, ISF)
```

erfolgen.

IBNR	=	Blocknummer, muß vor dem Aufruf definiert werden	
N	=	Blocklänge	
SFX	=	Frequenzauflösung $\Delta f$	} stehen nach Aufruf von GIVE zur Verfügung.
SFY	=	Dummy-Parameter	
ISF	=	Skalierungsfaktor von IBNR	

Die Änderung des Skalierungsfaktors kann über die Subroutine

```
ESCALE (IBNR, ISF)
```

vorgenommen werden.

Das folgende Beispiel zeigt eine Task, mit der ein Teilbereich eines Datenblocks ausgedruckt werden kann.

```
SUBROUTINE OUTP (IBNR, I1, I2, I3, I4, IB)
DIMENSION IB (1)
CALL GIVE (IBNR, N, SFX, SFY, ISF)
IANF = (IBNR-1) * N + I1
IEND = IANF + I2
CALL CLRIO
DO 10 I = IANF, IEND
```

```
F = IB(I) * 2 ** ISF
WRITE (6, 100) F
10 CONTINUE
CALL INPU
100 FORMAT (E12.4)
RETURN
END
```

Nach der Teletype-Eingabe:

OUTP, 10, 16, 30

werden die Worte 16 bis 30 von Block Nr. 10 auf der Teletype ausgegeben.

I3 und I4 sind Dummy-Parameter, damit IB an 6. Stelle steht.

Die Aufrufe CLRIO und INPU sind immer dann notwendig, wenn für Ein- oder Ausgabevorgänge die System-Konsole (Teletype) verwendet wird.

#### b.) Zugriff durch Konversion

Bei dieser Zugriffsart erfolgt eine Konversion zwischen NOASYS-Datenblöcken und Gleitkomma-Feldern. Die hierfür benötigten NOASYS-Unterprogramme GET, SAVE und GETDB haben folgende Funktion:

a.) GET(IBNR, F,K)

Ein NOASYS-Datenblock vom Type a wird in ein Gleitkomma-Feld umgewandelt.

b.) SAVE (IBNR, F, K)

Ein Gleitkomma-Feld wird in einen NOASYS-Datenblock (Type a) umgewandelt.

IBNR = Blocknummer

F = Feldname

K	NOASYS-Datenblock	Gleitkomma-Feld F
0	Zeitbereichsdaten	Zeitbereichsdaten
1	Frequenzbereichsdaten Realteil und Imaginärteil	Frequenzbereichsdaten paarweise nach Real- und Imaginärteil geordnet
2	Frequenzbereichsdaten mit Imaginärteil = 0	Nur Realteil in einem Feld der Länge N/2
3	Frequenzbereichsdaten Realteil und Imaginärteil	1. Felddhälfte: Realteil 2. Felddhälfte: Imaginärteil

Bei  $K = 2$  genügt ein Gleitkomma-Feld der Länge  $N/2$ , dies entspricht  $N$  Speicherzellen.

Anwendungsbeispiel für GET und SAVE

Der Realteil von Block IB1 soll mit dem Faktor 3.5 multipliziert werden, der Imaginärteil dagegen soll auf 0 gesetzt werden.

```
SUBROUTINE DEMO (IB1)
  DIMENSION F (256)
  CALL GET (IB1, F, 1)
  DO 10 I = 1, 256, 2
    F (I) = F (I) *3.5
    F (I+1) = 0.
  CALL SAVE (IB1, F, 1)
  RETURN
```

Beispiel 1

•  
•  
•



Handelt es sich bei der Subroutine DEMO um eine NOASYS-Task, so kann die Konversion auch direkt im NOASYS-Datenbereich erfolgen, dabei wird in Beispiel 1 das Feld F eingespart. Die Basis für das Gleitkomma-Feld F sollte so gewählt werden, daß sie mit Anfangsadressen von NOASYS-Datenblöcken übereinstimmt. Es ist dabei zu beachten, das jeweils 2 aufeinanderfolgende Blöcke für das Gleitkomma-Feld benötigt werden (Ausnahme K = 2). Beispiel 2 zeigt den Aufbau einer Task für eine Konversion im NOASYS-Datenbereich. Es soll dabei die gleiche Aufgabe wie in Beispiel 1 ausgeführt werden.

```
SUBROUTINE DEMO (IB1, IB2, I3, I4, I5, F)
  DIMENSION F(1)
  CALL GIVE (IBN, N, SFX, SFY, ISF)
  J = (IB2-1) * N/2 + 1
  CALL GET (IB1, F(J), 1)
  JE = J+N
  DO 10 I = J, JE, 2
    F(I) = F(I) * 3.5
  10 F(I+1) = 0.
  CALL SAVE (IB1, F(J), 1)
  RETURN
  END
```

Beispiel 2

In Beispiel 2 werden die Blöcke IB2 und IB2+1 als Gleitkommafeld verwendet. Für IB2 gelten folgende Einschränkungen:

```
für GET   :   IB2 ≠ IB1
für SAVE  :   IB2 ≠ IB1+1
```

Alle anderen Blöcke können für die Konversion verwendet werden.

### c.) Zugriff zu Doppelwort-Datenblöcken

Mit der Subroutine GETDB werden Doppelwort-Datenblöcke, wie sie z.B. von der Task CPSD erzeugt werden, normiert und im gleichen Block in ein Gleitkomma-Feld konvertiert.

Aufruf: CALL GETDB (IBNR, I)

IBNR = Blocknummer (Bei Doppelblöcken gilt die niedrigere Blocknummer  
I = Index des 1. Wortes im Gleitkommafeld F, (siehe Beispiel).

Ein Zugriff zu den konvertierten Daten zeigt folgendes Beispiel:

```
SUBROUTINE DEMO (IBNR, I2, I3, I4, I5, F)
DIMENSION F(1)
CALL GIVE (IBNR, N)
CALL GETDB (IBNR, J)
DO 10 IR = J, J + N/2
REAL = F(IR)
FIMAG = F(IR + N/2)
10 CONTINUE
.
```

### 3.4 Beschreibung der Standard-Tasks

NOASYS-Tasks können ihrer Funktion entsprechend in folgende Gruppen eingeteilt werden:

- a.) Monitor-Tasks
- b.) Steuer-Tasks für die Datenerfassung
- c.) Ein- Ausgabe-Tasks
- d.) Tasks für Datenblockoperationen
- e.) Benutzer-Tasks

#### 3.4.1 Monitor-Tasks

CLEAR

Lösche alle Eintragungen im Task-Puffer. Beim Start von NOASYS wird der Task-Puffer gelöscht.

LIST

Alle Eintragungen im Task-Puffer werden auf der Teletype gelistet.  
Format: Sequenz-Nr., Task-Name, Parametersatz.

PLIST

PLIST hat die gleiche Funktion wie LIST, die Ausgabe erfolgt jedoch auf Lochstreifen.

PTAPE

Einlesen von mit PLIST erzeugten Lochstreifen in den Task-Puffer.

ENDE

Ende einer Sequenz im Task-Puffer und Rücksprung in den Monitor.  
Mit ENDE können mehrere Task-Sequenzen voneinander getrennt werden.  
Beispiel:

- 1, ENDE
- 2, DKOH, 16, 5, 6, 7
- 3, WPT, 16
- 4, ENDE
- 5, FT, 1
- 6, BPH, 1
- 7, WPT, 1
- 8, ENDE

Mit dem Aufruf GOTO, 2 werden die Tasks DKOH und WPT einmal ausgeführt, entsprechendes gilt beim Aufruf GOTO, 5 für die Tasks FT, BPH und WPT.

Einen Sonderfall bildet der Eintrag ENDE, 0, der immer dann notwendig ist, wenn eine durch den Intervall-Timer gestartete Task-Sequenz abgeschlossen wird.

GOTO, n

Sprung in den Task-Puffer zur Sequenznummer n.

### 3.4.2 Steuer-Tasks für die Datenerfassung

MESS, n, k, m, s

Mit der Task MESS werden die wichtigsten Parameter für die Datenerfassung definiert.

n = Anzahl der Worte pro Datenblock (Blocklänge).

n = 16, 32, ... , 1024, 2048

k = Anzahl der benötigten Meßkanäle;

k = 1 bis 16 für NOASYS-2

k = 1 bis 4 für NOASYS-1

m = Puffermodus (0, 1, 2)

0 = Wechselpuffer

1 = Einfachpuffer

2 = Wechselpuffer mit Halbblocken

s = Samplefrequenzeinstellung (nur NOASYS-2)

s = 0 : manuelle Einstellung der Samplefrequenz

s = 1 bis 15 : Vorgabe der Samplefrequenz durch den Rechner;

$f_s = 2^s$  Hz

Beim Starten von NOASYS wird MESS, 128, 1, 0, 0 vom Monitor aufgerufen.

Fehlermeldungen:

-----

ME01 n < 16 oder n > 2048 oder  $\log_2 n$  keine ganze Zahl.

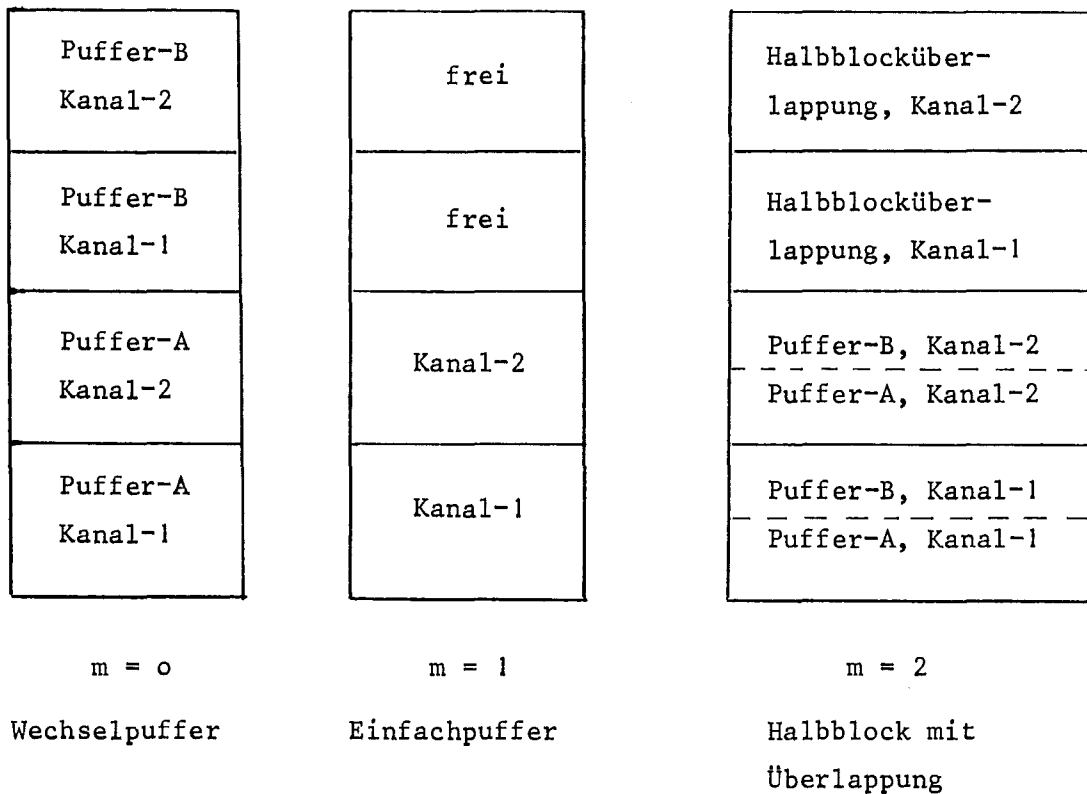
ME02 Der NOASYS-Datenbereich ist zu klein, um einen Pufferbereich für k Meßkanäle anzulegen.

Beim Aufruf von MESS werden alle Datenblöcke durch Aufruf der Task ERA gelöscht.

Anordnung der Meßdaten für  $m = 0, 1, 2$

---

Beispiel, für  $k = 2$



In allen 3 Fällen kann mit negativen Blocknummern zu den aktuellen Datenblöcken zugegriffen werden. Für  $m = 1$  und  $m = 2$  ist jedoch auch eine direkte Blockadressierung möglich.

RUN, a

Mit dem Aufruf RUN, a werden folgende Aktivitäten eingeleitet:

- 1.) Die als Zähler verwendete Variable @SCNT wird -a gesetzt.
- 2.) Die beiden Wechselpuffer-Flag's FA und FB werden gelöscht (beide Pufferbereiche leer).
- 3.) Die Meßdatenerfassungseinheit wird gestartet.

Mit RUN, a wird eine zwischen WAIT und NEXT stehende Task-Sequenz gestartet und a-mal durchlaufen (siehe NEXT). Bei Halblöcken (m = 2) werden a+1 Halblöcke bzw. a überlappende Ganzblöcke verarbeitet. Die Task RUN kann auch durch Knopfdruck von der Steuerkonsole gestartet werden. Dabei ist der zuletzt verwendete Parameter a gültig.

#### WAIT

WAIT hat die gleiche Funktion wie ENDE solange kein Pufferbereich gefüllt ist. Wenn ein gefüllter Pufferbereich bereits durch NEXT wieder freigegeben wurde, gilt dieser als nicht gefüllt. Sobald ein Pufferbereich gefüllt ist, wird zur nächsten Task verzweigt (Abb. 7).

#### NEXT, s

Ein gefüllter Pufferbereich wird nach seiner Verarbeitung für eine neue Füllung freigegeben. Bei jedem Aufruf von NEXT wird der durch RUN gesetzte Zähler@SCNT inkrementiert. Bei@SCNT=0 wird die Meßdatenerfassungseinheit ausgeschaltet und zur Sequenznummer s im Task-Puffer verzweigt. Bei @SCNT≠0 wird zur nächsten Task verzweigt. Wird NEXT ohne Parameter eingetragen, so erfolgt auch für @SCNT=0 nur eine Verzweigung zur nächsten Task im Puffer. Siehe Flußdiagramm für NEXT und WAIT, Abb. 7.

#### ZEIT, t, s

Starte Intervall-Timer.

t = Intervallzeit in Einheiten von 1/10 sek.

s = Task-Sequenznummer, zu der nach jedem Intervall verzweigt wird.

Mit der Task ZEIT, t, s wird ein Intervall-Timer gestartet. Am Ende eines jeden Intervalls t erfolgt ein Sprung in den Task-Puffer nach s.

Eine Task-Sequenz, die durch den Intervall-Timer gestartet werden soll, muß wie folgt aufgebaut sein:

- 1, ENDE
- 2, FT, 1
- 3, KKM, 1,1
- 4, ENDE, 0

Mit dem Aufruf Zeit, 20, 2 wird die Task-Sequenz im obigen Beispiel alle 2 sek. aufgerufen.

Mit Zeit, 0 kann der Intervall-Timer wieder ausgeschaltet werden.

### 3.4.3 Ein-Ausgabe Tasks

#### DSP

Ausgabe von NOASYS-Datenblöcken auf ein Sichtgerät. Die Parameter für DSP können über die Steuer-Konsole eingegeben werden. Die Task DSP wird vom Monitor so behandelt, als wäre sie an erster Stelle im Task-Puffer eingetragen. Mit SW15 kann dieser automatische Aufruf unterdrückt werden. DSP kann auch in eine Task-Sequenz eingereiht werden, um z.B. einen Datenblock unmittelbar vor b.z.w. nach seiner Verarbeitung auszugeben.

#### ERA, a

Lösche Block Nr. a

Alle Datenworte sowie ST1 und ST2 werden auf 0 gesetzt. Der Skalierungsfaktor SF wird auf -127 gesetzt.

#### TRA, s, d, m

Transport eines Datenblocks innerhalb des Kernspeichers oder zwischen Platte und Kernspeicher.

s = Blocknummer der Datenquelle

d = " " Datensenke

m = Index, hat nur für (s oder d) > 100 eine Bedeutung.

#### 1.) ( s oder d ) < 100

Für ( s oder d ) < 100 wird automatisch die Work-Area des Plattenspeichers benutzt, wenn die entsprechende Blocknummer im Kernspeicher nicht mehr ver-

füßbar ist. Die Work-Area des Plattenspeichers wird dabei ebenfalls in Blöcke aufgeteilt und mit Blocknummern adressiert.

Beispiel:

BLZA sei die maximale Zahl der Blöcke im Kernspeicher.

Für ( s und d )  $\leq$  BLZA Transfer innerhalb des Kernspeichers;  
" s  $>$  BLZA Transfer Platte — Kernspeicher;  
" d  $>$  BLZA Transfer Kernspeicher — Platte;  
" ( d und s )  $>$  BLZA keine Operation;

Der TRA-Aufruf mit  $BLZA < (s, d) \leq 100$  sollte nur dann verwendet werden, wenn der Plattenspeicher zur virtuellen Erweiterung des Kernspeicher- Datenbereichs benötigt wird.

2.) ( s oder d )  $>$  100  
-----

Transfer zwischen Platte und Kernspeicher.

Zur Blocknummer s oder d wird der Wert  $m \cdot x$  hinzuaddiert. Beim Faktor x handelt es sich um einen Index, der bei jedem Aufruf von NEXT inkrementiert wird. Mit RUN, a wird x auf 0 gesetzt.

1. Beispiel: Sequentielle Meßdatenerfassung von 3 Analogsignalen auf die Work-Area des Plattenspeichers.

1, WAIT  
2, TRA, -1, 101, 3  
3, TRA, -2, 102, 3  
4, TRA, -3, 103, 3  
5, NEXT, 1

Start mit RUN, a ; dabei werden  $3 \cdot a$  Datenblöcke sequentiell in der Work-Area abgelegt.



2. Beispiel: Einlesen der Datenblöcke von der Work-Area in den Kernspeicher zur weiteren Verarbeitung.

```
1, ENDE
2, TRA, 101, 1, 3
3, TRA, 102, 2, 3
4, TRA, 103, 3, 3
.
.
.
10, NEXT, 12
11, GOTO, 2
```

Verarbeitungs-Tasks

Start mit RUN, a und dann GOTO, 2. Es muß dafür gesorgt werden, daß die Meßdatenerfassungseinheit nicht gestartet wird, z.B. auf TRIG. EXT. schalten.

HTRA, s, d, h

Halbblock-Transfer

s = Quellblock

d = Zielblock

h = Blockhälfte

1.) s = positiv

-----

```
h = 1 : Transfer 2. Hälfte von s nach 1. Hälfte von d ;
h = 2 : " 1. " " " 2. " " " ;
h = 3 : " 1. " " " 1. " " "
        sowie Nullsetzen der 2. Hälfte von d ;
```

2.) s = negativ

-----

```
h = 1 : Transfer Wechselfuffer s nach 1. Hälfte von d ;
h = 2 : " " " " 2. " " " ;
h = 3 : " " " " 1. " " "
        sowie Nullsetzen der 2. Hälfte von d ;
```

WPT, b

Lochstreifenausgabe von Block Nr. b

RPT, b, x

Einlesen von Lochstreifen nach Block Nr. b

x = 0    Zeitbereichsdaten

x = 1    Frequenzbereichsdaten

Fehlermeldung : RPO1

Nach dem Vorspann:    Ungültige Blocklänge;

Bei Lochstreifenende: Lesefehler;

Lochstreifenformat

Der Datensatz auf dem Lochstreifen setzt sich zusammen aus Vorspann und Daten. Für jedes 16-bit Wort wird eine Doppelsprosse (DS) benötigt.

0	1	2	x	3	4	5	6	7
8	9	10	x	11	12	13	14	15

1. Sprosse bit 0 - 7

2. Sprosse bit 8 - 15

x = Transportloch

1. DS = -1 ; (Start Vorspann)
2. DS = Blocklänge (N)
3. DS = Skalierungsfaktor (SF)
4. DS = Statuswort ST 1
5. DS = Frequenzauflösung (  $1/N\Delta t$  )
6. DS

- 7. DS = SFY (Gleitkomma 1.)
- 8. DS
- 9. DS = -1 ; Trennung zwischen Vorspann und Daten
- 10. DS bis = Daten ; (16-bit Integer)
- (N + 10). DS
- (N + 11). DS = Prüfsumme
- (N + 12). DS = Statuswort ST2

ST2 wird nicht in die Prüfsumme mit einbezogen.

#### 3.4.4 Datenblock - Operationen

FT, b

Diskrete Fouriertransformation einer reellen Funktion in Block Nr. b.

Rechenzeit für N = 1024 : NOASYS-2 180 ms

NOASYS-1 230 ms

ST1 : bit-13 = 1

ST2 : unverändert

HA, b

Multiplikation von Block Nr. b mit einer Fensterfunktion  $F_n$  (Hanning).

$$F_n = ( 1 - \cos 2\pi n/N ) / 2$$

mit  $n = 0, 1, 2, \dots, N-1$

SF und ST2 unverändert

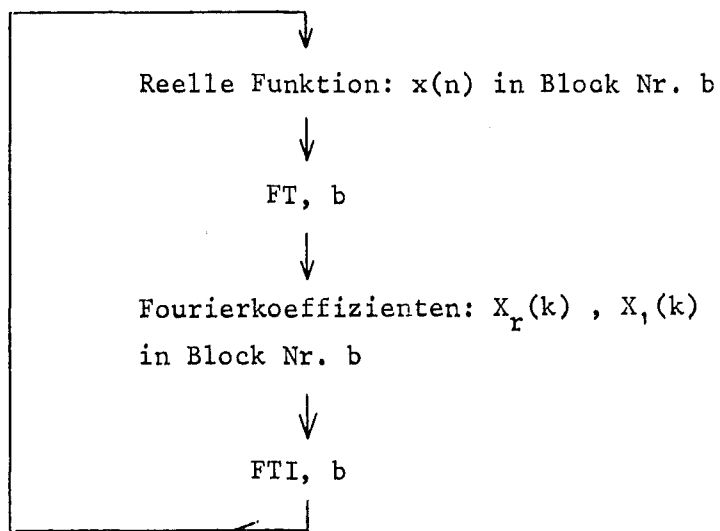
ST1 : bit 12 = 1

HFT, b

Zusammenfassung von HA, b und FT, b

FTI, b

Inverse Fouriertransformation von Block Nr. b.



$$n = 0, 1, \dots, N-1$$

$$k = 0, 1, \dots, N/2-1$$

ST1: bit 12 = 0

ST2: unverändert

KKM, b1, b2

Konjugiert komplexe Multiplikation der Blöcke b1 und b2.

$$(b1)^* * (b2) \rightarrow (b2) ; b1 = b2 \text{ zulässig.}$$

KMP, b1, b2

Komplexe Multiplikation der Blöcke b1 und b2.

$$(b1) * (b2) \rightarrow (b2) ; b1 = b2 \text{ zulässig}$$

TMP, b1, b2

Multiplikation der Blöcke b1 und b2

$$(b1) \cdot (b2) \rightarrow (b2) ; b1 = b2 \text{ zulässig}$$

TDI, b1, b2

Division der Blöcke b1 und b2.

$$(b1) / (b2) \rightarrow (b2)$$

BPH, b

Berechnung von Betrag und Phase aus einem komplexen Spektrum, in dem Real- und Imaginärteil paarweise angeordnet sind.

b = Blocknummer

Der Skalierungsfaktor SF hat für die Phase keine Bedeutung. Die Phase berechnet sich wie folgt:

$$P = \pi \frac{w}{4096} - 1$$

w = Phasenwert (16-bit Integer) aus Block Nr. b

Zur Ausgabe der Phase auf dem Sichtgerät sind folgende Einstellungen notwendig:

Offset = 0 ; Scale = 3

Die Phase wird dann im Bereich  $-\pi$  bis  $+\pi$  auf dem Sichtgerät dargestellt.

ST1 : bit-11 = 1

UPSC, b

Normalisierung von Blocknummer b, so daß keine führenden Nullen verbleiben.

ST1: bit 15 u. bit-14 werden aktualisiert.

ADD, b1, b2

Addition von zwei Datenblöcken.

$$(b1) + (b2) \rightarrow (b2)$$

CPSD, x, y, e, f, g

Die Task CPSD dient primär zur Berechnung von SALD und SKLD aus den linearen Spektren. Sie Spektralwerte werden durch arithmetische Mittelung (summation-mode) oder durch exponential-averaging mit jeweils doppelter Wortlänge berechnet.

Die Task CPSD kann auch zur Mittelwertbildung ohne vorherige konjugiert komplexe Multiplikation verwendet werden.

Parameterbeschreibung:

x : Blocknummer des linearen Spektrum x  
y : Blocknummer des linearen Spektrum y  
e : Blocknummer der SALD bzw. SKLD (Ergebnis)

Für  $x \neq y$  wird die SKLD in Blocknummer e und e + 1 gemittelt.

Für  $x = y$  wird die SALD in Blocknummer e gemittelt, e + 1 wird dabei nicht benötigt.

Für  $x \neq 0$  und  $y = 0$  erfolgt Mittelwertbildung der Datenblöcke x ohne konjugiert komplexe Multiplikation.

f : Anzahl der Spektralwerte, die bei der Mittelwertbildung ignoriert werden. Damit kann verhindert werden, daß hohe Gleichanteile die Dynamik des restlichen Spektrums beeinträchtigen.

g = 0 : summation-averaging  
g  $\neq$  0 : exponential-averaging  
g = -1 : Mittelungskonstante wird über die Steuerkonsole eingestellt;  
0 < g  $\leq$  15 : Mittelungskonstante = g ;

Für die Zeitkonstante  $\tau$  gilt:

$$\tau = T_b \cdot 2^g$$

$T_b$  = Zeit zwischen zwei aufeinanderfolgenden Mittelwertbildungen.

Normierung

-----

Beim 1. Durchlauf einer Task-Sequenz, die mit RUN gestartet wurde, werden die Blöcke für das Ergebnis (e bzw. e und e+1) gelöscht. Die Zahl der aufsummierten Einzelspektren wird im Statuswort ST2 von Block Nr. e gezählt. Bei exponential-averaging wird ST2 = 1 gesetzt.

Mit dem Aufruf CPSD, 0, 1, e erfolgt eine Normierung der SLD (Division durch ST2) und eine Umwandlung in einen Einfachblock mit 16-bit Wortlänge (Type a). Wird die Konversion in ein Gleitkommafeld mit der Subroutine GETDB gewünscht, so muß diese Normierung unterbleiben.

Die Daten in den Blöcken x und y werden durch die Task CPSD nicht verändert.

Abb. 9 zeigt die Anordnung der Spektralwerte in den Blöcken e und e+1 für eine SALD und eine SKLD.

Beispiel für die Berechnung der SALD und der SKLD von 2 Eingangssignalen im summation mode:

- 1, WAIT
  - 2, FT, -1
  - 3, FT, -2
  - 4, CPSD, -1, -1, 5, 0, 0
  - 5, CPSD, -2, -2, 6, 0, 0
  - 6, CPSD, -1, -2, 7, 0, 0
  - 7, NEXT, 10
  - 8, ENDE
  - 10, CPSD, 0, 1, 5
  - 11, CPSD, 0, 1, 6
  - 12, CPSD, 0, 1, 7
- } Normierung und Umwandlung  
im Einfachblöcke

Rechenzeiten: N = 1024

	NOASYS-1	NOASYS-2
SALD	90 ms	18 ms
SKLD	160 ms	34 ms



#### 4. Systemerweiterung

Ein Benutzer von NOASYS hat die Möglichkeit, eigene Tasks zu erstellen und in ein System zu integrieren. Die Programmierung solcher Tasks kann als Assembler- oder Fortran-Subroutine erfolgen. Der Name der Subroutine muß dabei in einer Task-Liste vereinbart werden.

Es gibt 2 Task-Listen:

1. Standard-Liste, sie enthält die Namen aller Tasks, die als Basis für jedes System benötigt werden.
2. Erweiterungs-Liste, sie enthält die Namen von Benutzer-Tasks.

Die Erweiterungs-Liste wird mit dem Programm TLIST erzeugt, dabei können maximal 15 Namen vereinbart werden.

Beispiel:

```
:PR, TLIST      Programm-Aufruf
NAME?
INTE           1. Name
NAME?
OUTP           2. Name
NAME?
/E             Ende
```

Als Namen müssen die ersten vier Zeichen des entsprechenden Subroutine-Namens verwendet werden.

Das Programm TLIST erzeugt einen object code, der in der JOB-Binary-Area abgelegt wird. TLIST kann innerhalb eines Job's gemeinsam mit anderen Sprachübersetzern aufgerufen werden.

#### Parameterversorgung

Eine Task kann bei ihrem Aufruf mit maximal 5 Integer-Parametern versorgt werden.

Beispiel:

Teletype-Eingabe: INTE, -1, 512, 13, -1000, 10

Die 5 Parameter werden entsprechend ihrer Reihenfolge, der Parameter-Liste I1 bis I5 der Task INTE zugeordnet.

```
SUBROUTINE INTE (I1, I2, I3, I4, I5, IB, IP, FB)
  DIMENSION  IB(1), IP(1), FB(1)
```

Die letzten 3 Parameter IB, IP und FB werden vom NOASYS-Monitor bereitgestellt, sie haben folgende Bedeutung:

IB, FB = Basisadresse des NOASYS-Datenbereichs  
IP = Basisadresse der NOASYS-Globalparameter-Liste  
(siehe Anhang)

Ein-Ausgabe auf die System-Konsole (Teletype)

---

Beispiel:

```
      .
      .
      .
CALL CLRIO
WRITE (1, 100) A, B, C
READ (1, *) E, F, G
      .
      .
CALL INPU
      .
      .
```

Der Aufruf CLRIO bewirkt eine Freigabe der System-Konsole vom Monitor, mit CALL INPU wird die Kontrolle der System-Konsole wieder an den Monitor zurückgegeben.

#### 4.1. Systemgenerierung mit dem DOS-Loader

Ein NOASYS-Load-Modul ist eine von DOS-Loader erzeugte stand-alone-Version von NOASYS.

Mit dem Aufruf:

: PR, DMONn

kann ein solcher Load-Modul in den Kernspeicher geladen und gestartet werden, ( n = 1 bis 7). Jeder Benutzer hat die Möglichkeit, sich spezielle anwendungsorientierte NOASYS-Load-Moduln zu erstellen. NOASYS-Tasks können nach den in Abschnitt 4, beschriebenen Regeln erstellt werden.

Für die Erzeugung eines Load-Moduls werden folgende User-Files benötigt:

1. RMONn      Monitor Version n ; n = 1 bis 7
2. FULIB      Tasks und Subroutinen für NOASYS
3. LIB00      Tasks und Subroutinen nur für NOASYS-2
4. LIB16      Tasks und Subroutinen nur für NOASYS-1

Abkürzungen

ADC	Analog-Digital-Converter
AKF	Autokorrelationsfunktion
DFT	Diskrete Fouriertransformation
DMA	Direct Memory Access
DOS	Disc Operating System
FFT	Fast Fourier Transform
KKF	Kreuzkorrelationsfunktion
lsb	least significant bit
MDE	Meßdatenerfassungseinheit
msb	most significant bit
NOASYS	Noise Analysis System
SALD	Spektrale Autoleistungsdichte
SKLD	Spektrale Kreuzleistungsdichte
SLD	Spektrale Leistungsdichte
SF	Skalierungsfaktor
ST1	Statuswort -1
ST2	Statuswort -2
SW	Switch-Register



Literaturverzeichnis

- /1/ W. Seifritz, D. Stegemann:  
Reactor Noise Analysis,  
Atomic Energy Review 9 (1), IAEA, Vienna, (1975)
- /2/ M. Edelmann:  
Two on-line Methods for Routine Testing of Neutron and Temperature  
Instrumentation of Power Reactors, KFK-Bericht Nr. 2316 (1976)
- /3/ J. Ehrhardt:  
Nachweis bandbegrenzter Komponenten in verrauschten Signalen durch  
Überwachung der spektralen Auto- und Kreuzleistungsdichten  
KFK-Bericht 2559 (1978)
- /4/ J. Ehrhardt, H. Massier:  
Überwachungsgerät für den schnellen Nachweis oszillatorischer Kompo-  
nenten in verrauschten Signalen,  
KFK-Bericht Nr. 2159 (1976)
- /5/ P. Hoppé, F. Mitzel:  
Berechnung, Messung und Analyse der Übertragungsfunktion zwischen  
der Reaktivität und der Leistung an der kompakten Natriumgekühlten  
Kernenergieanlage KNK-I,  
KFK-Bericht Nr. 2385 (1976)
- /6/ M. Audox, F.W. Katz:  
SEDAP-Rechnergestützte Auswertung technischer Versuche,  
KFK-Nachrichten 3/71, 24 (1971)
- /7/ J.S. Bendat, A.G. Piersol:  
Measurement and Analysis of Random Data, John Wiley and Sons,  
New York, (1966)

- /8/ W.T. Cochran et al.:  
What is the Fast Fourier Transform, Proc. of the IEEE 55,  
S. 1664 - 1674 (1967)
- /9/ L.R. Rabiner, B. Gold:  
Theory and Application of Digital Signal Processing, Prentice-Hall,  
Englewood Cliffs, New Jersey (1975)
- /10/ H. Massier:  
Anlage zur Real-time Signalanalyse mit einem Kleinrechner,  
(1974), unveröffentlicht.
- /11/ H. Massier:  
Ein Mikroprogramm zur schnellen Fouriertransformation für die on-line  
Signalanalyse, (1976), unveröffentlicht.
- /12/ P. Hoppé:  
Analyse des Neutronenrauschens an natriumgekühlten Reaktoren zur Er-  
kennung von Funktionsstörungen  
KFK-Bericht Nr. 2589 (1978)

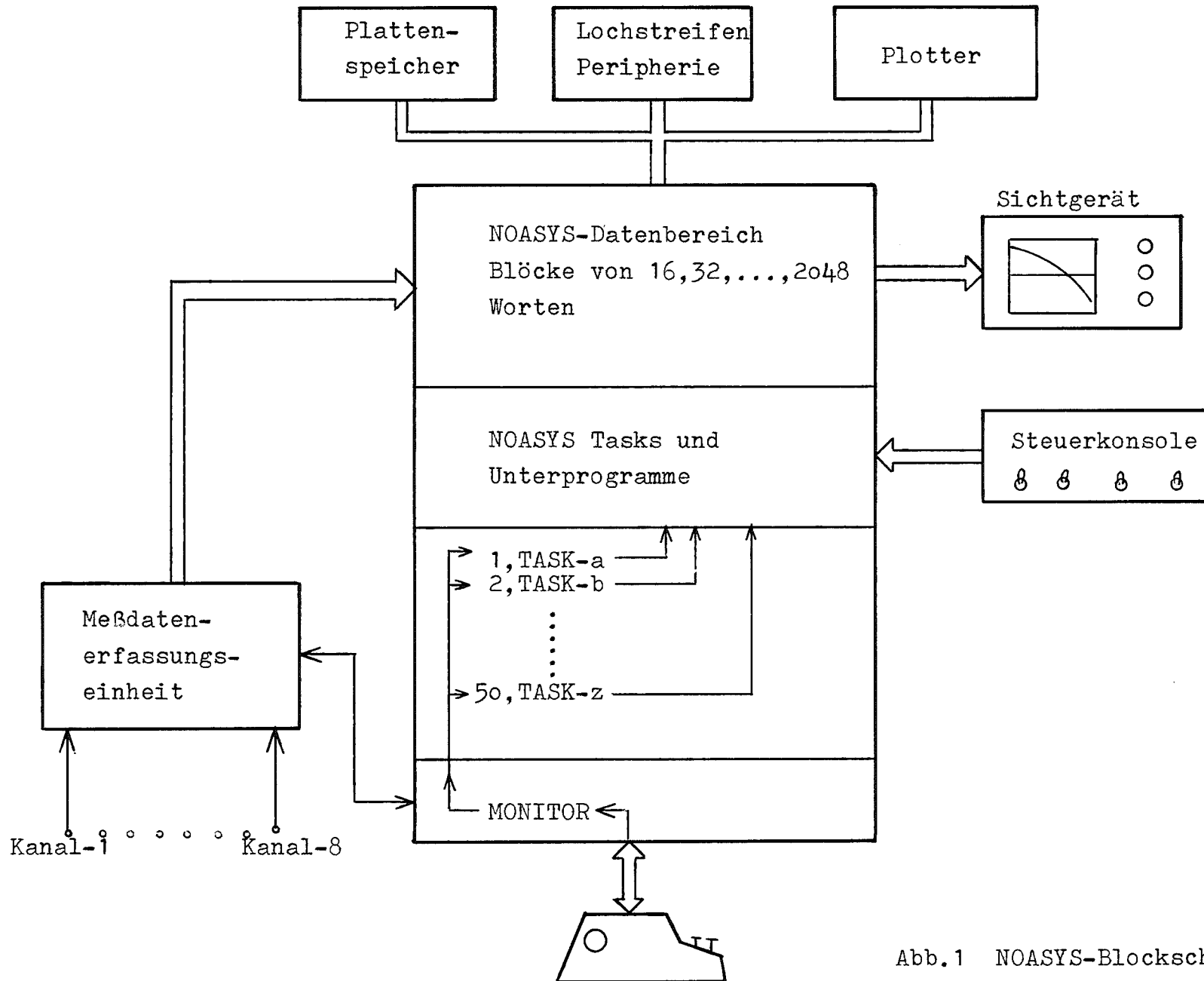


Abb.1 NOASYS-Blockschema



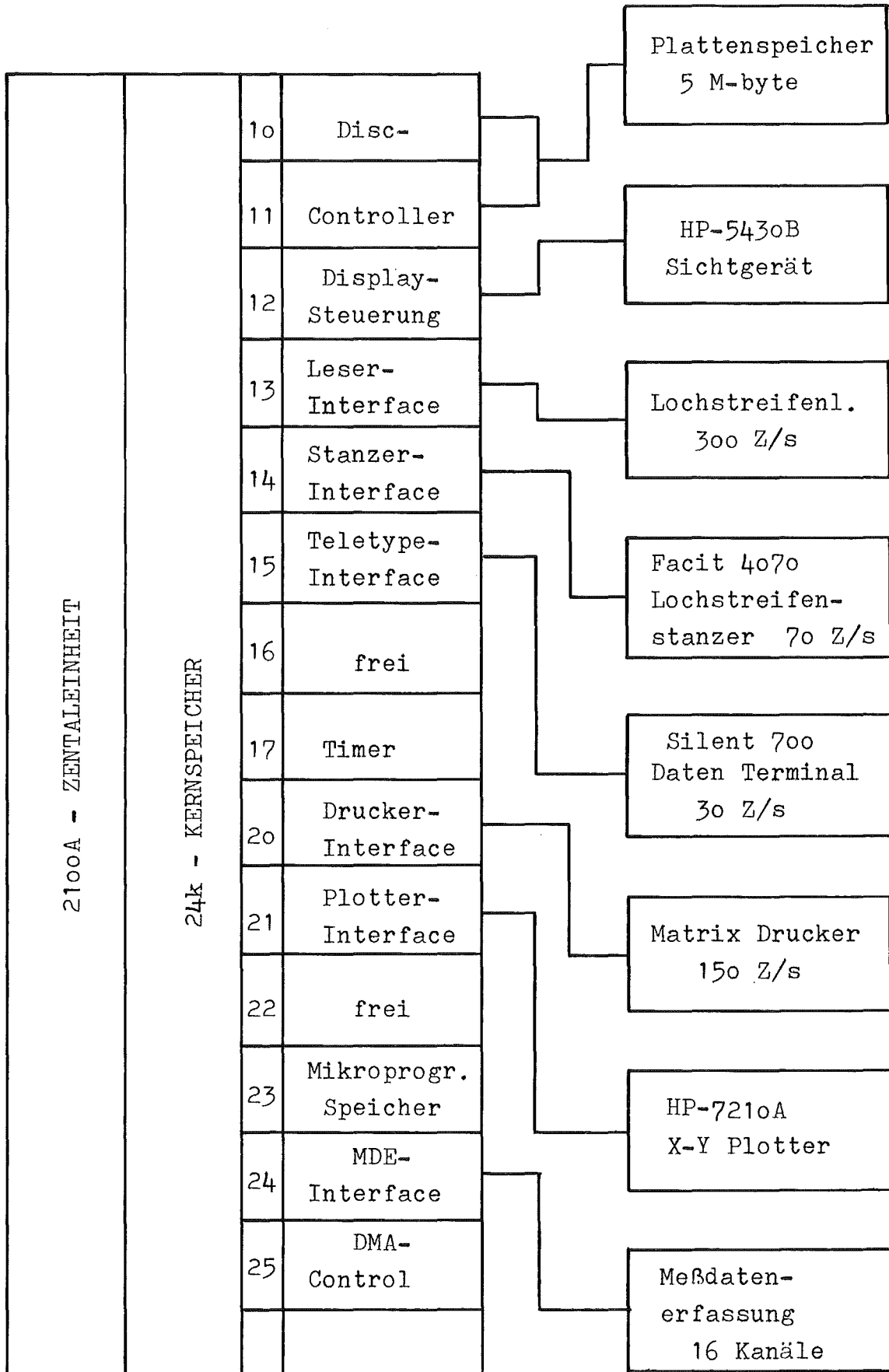


Abb.2 HP-2100 Konfiguration

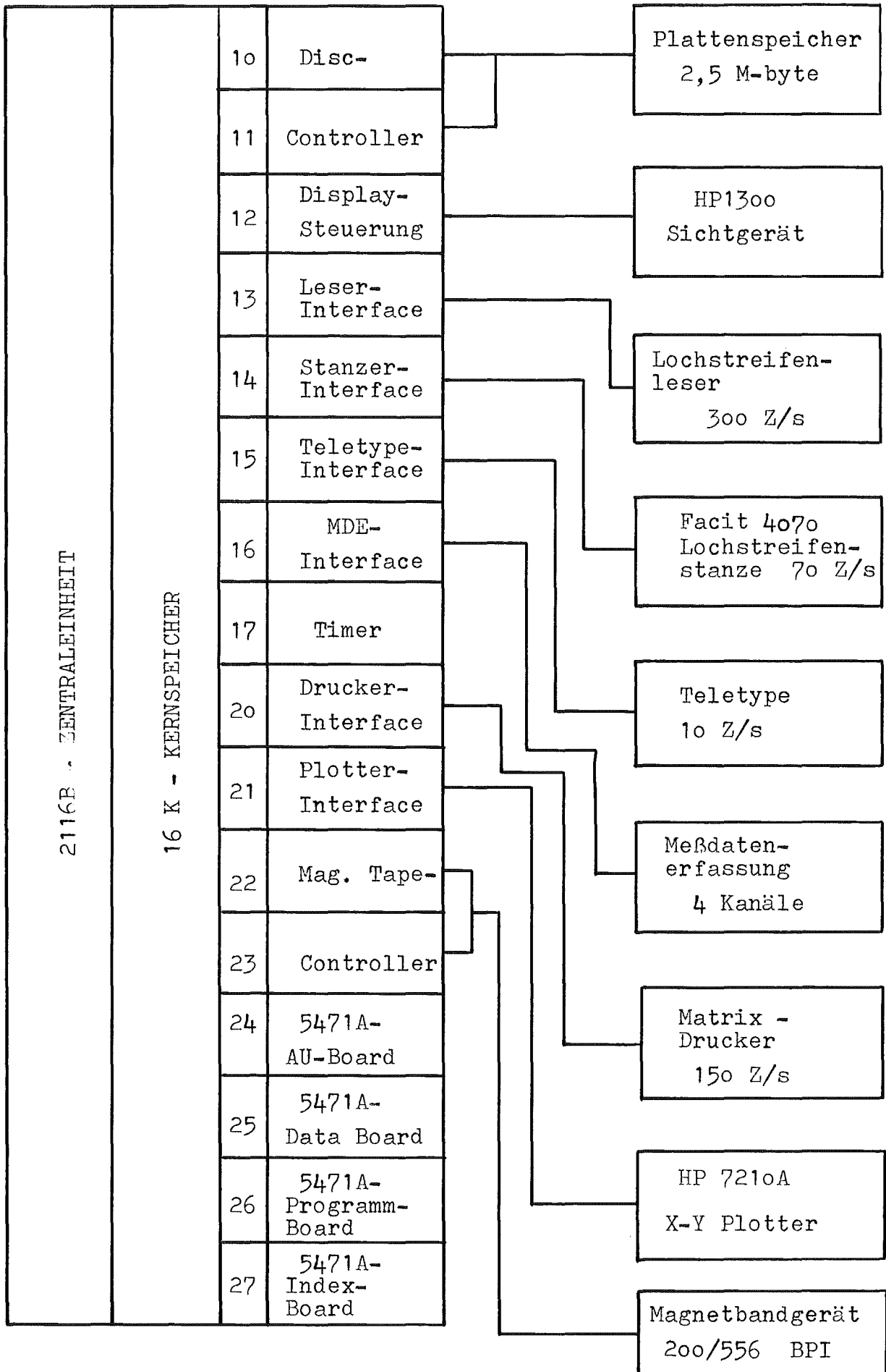


Abb. 3 HP-2116 Konfiguration

Abb.4 MDE-Blockschaltbild

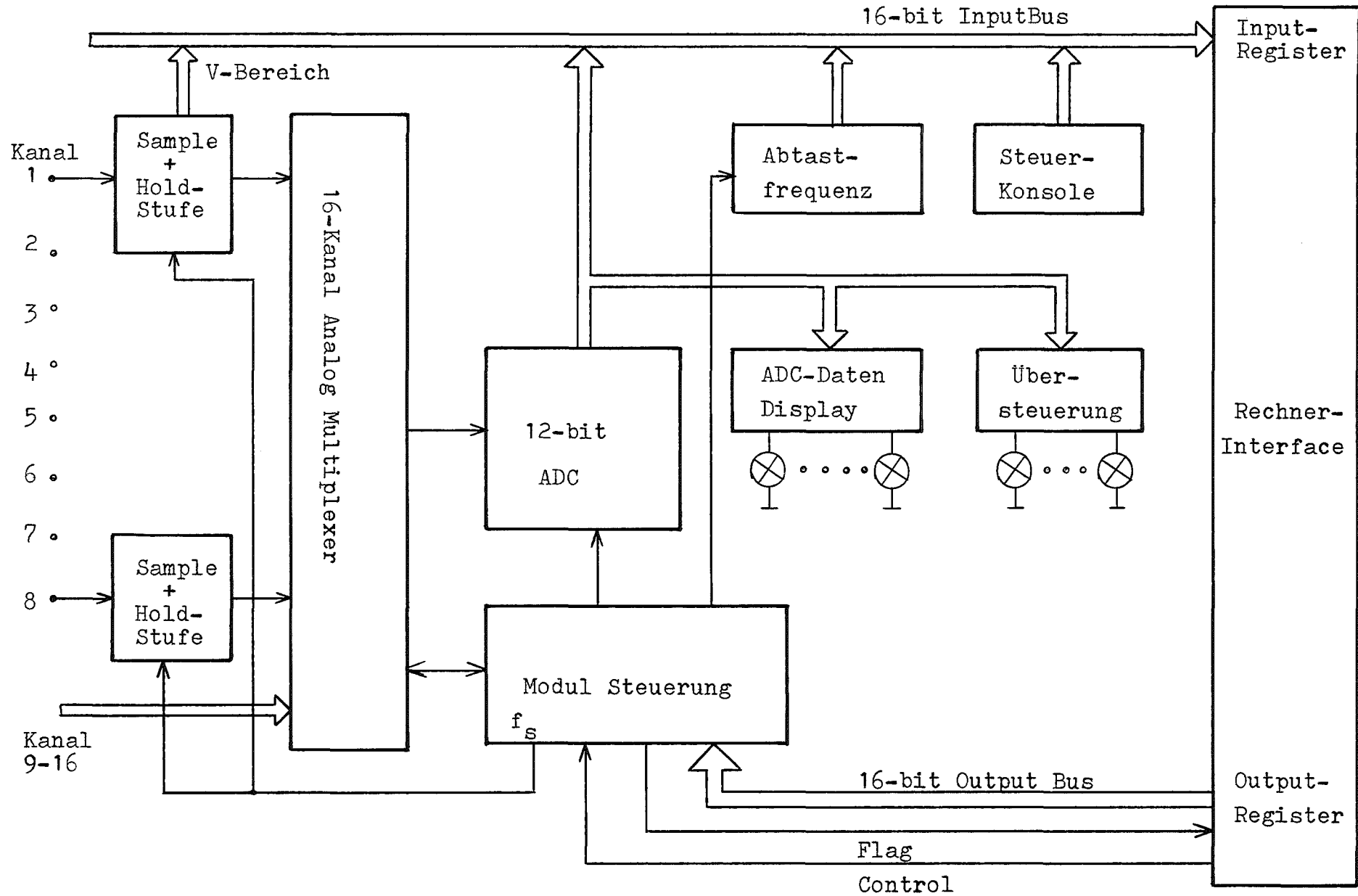
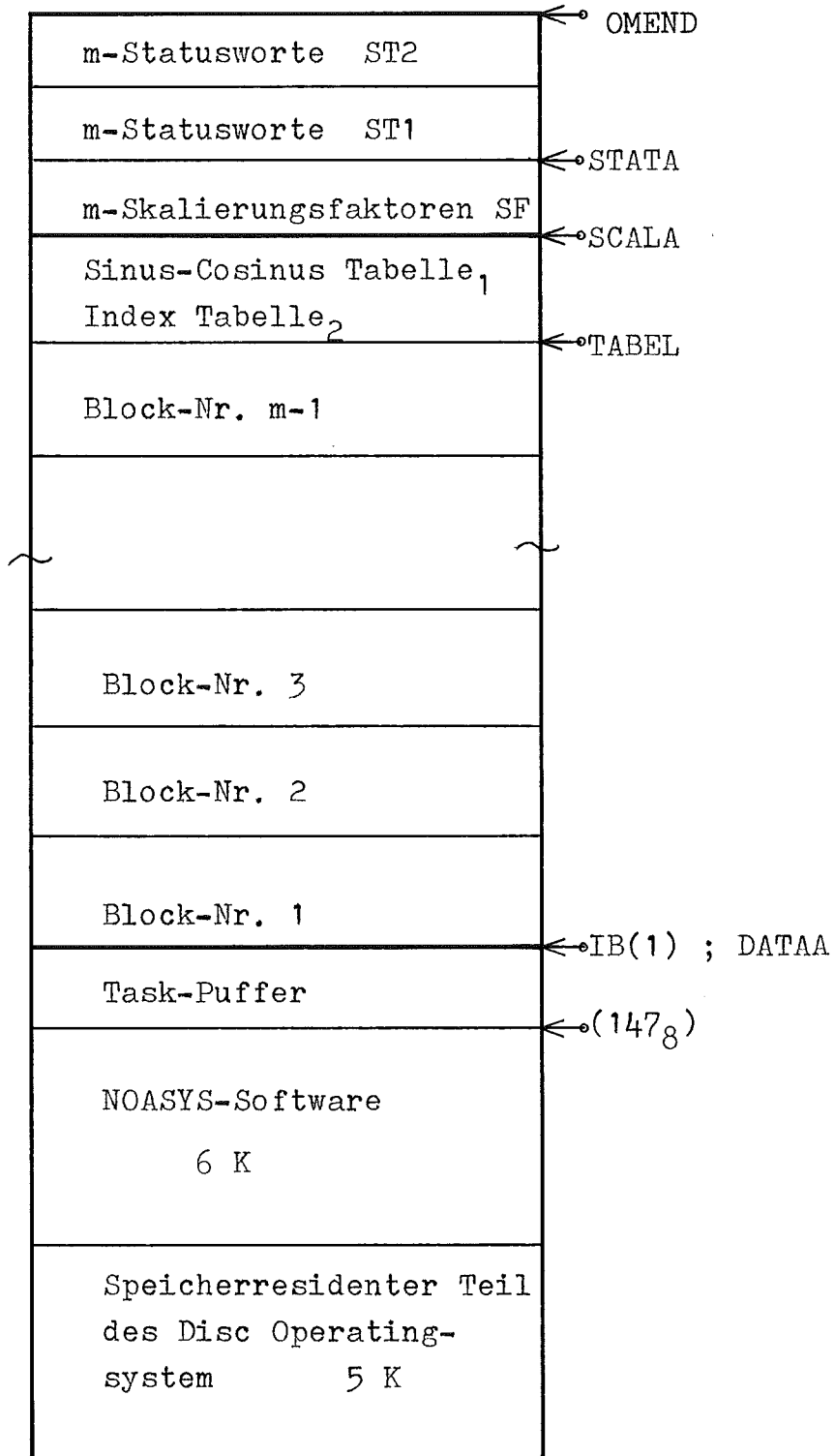
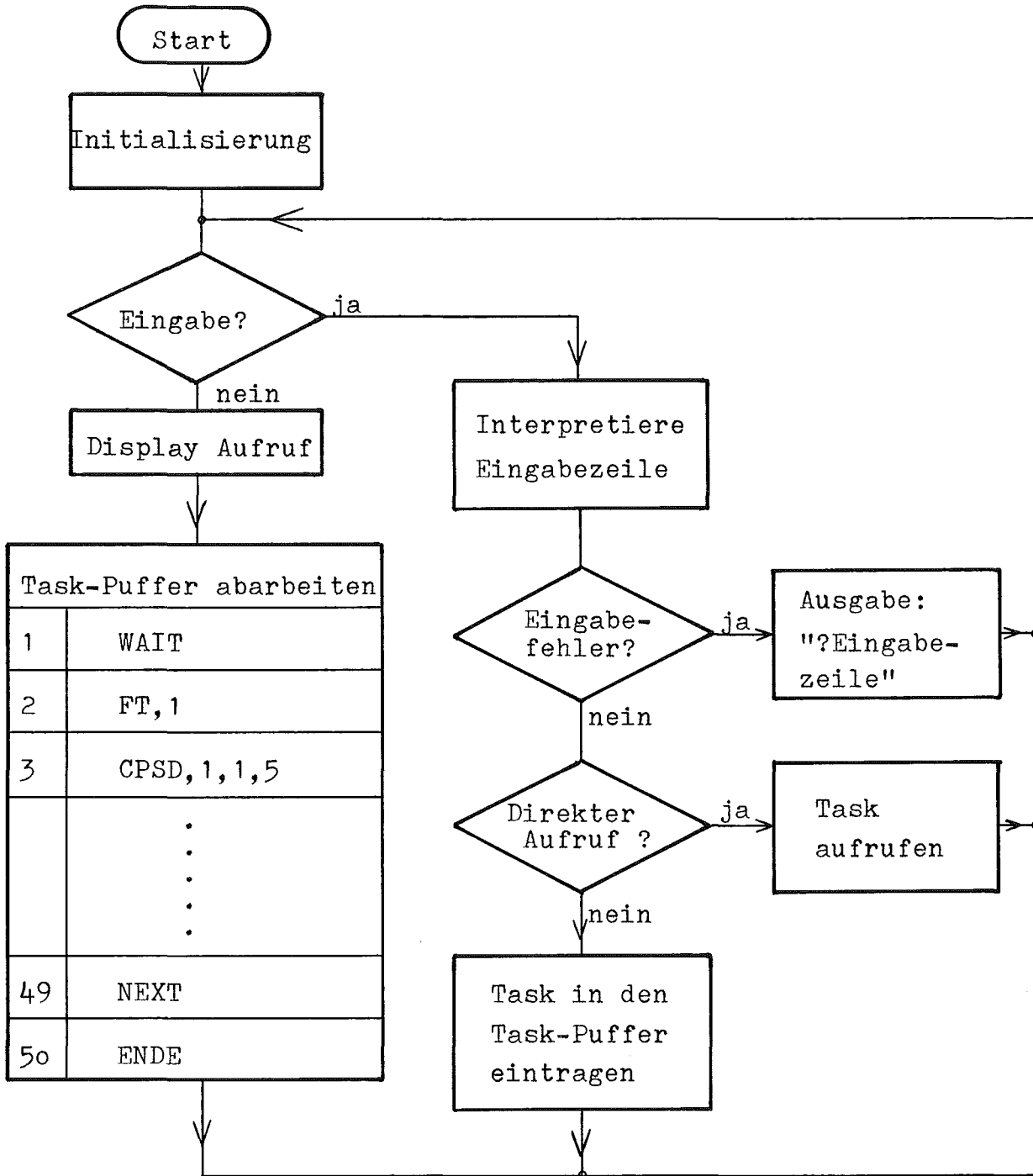


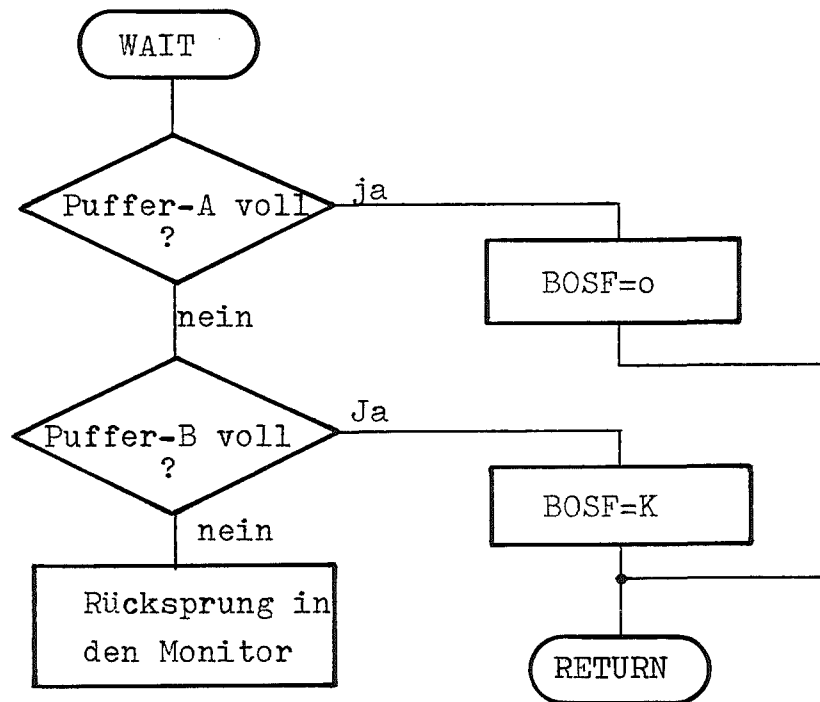
Abb.5 NOASYS-Speicherbelegung



1. NOASYS-2
2. NOASYS-1

Abb.6 MONITOR-Flußdiagramm





BOSF = Block-Offset für Wechselpuffer-adressierung

K = Kanalzahl

S = Sequenznummer im Taskpuffer

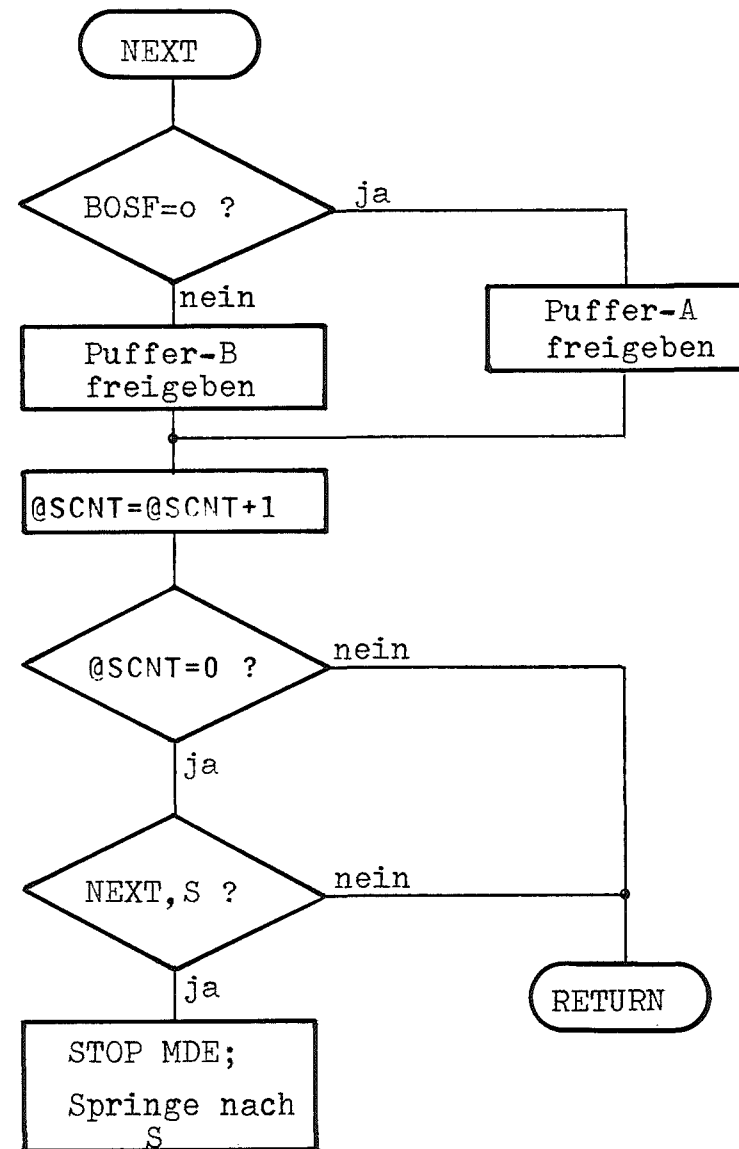
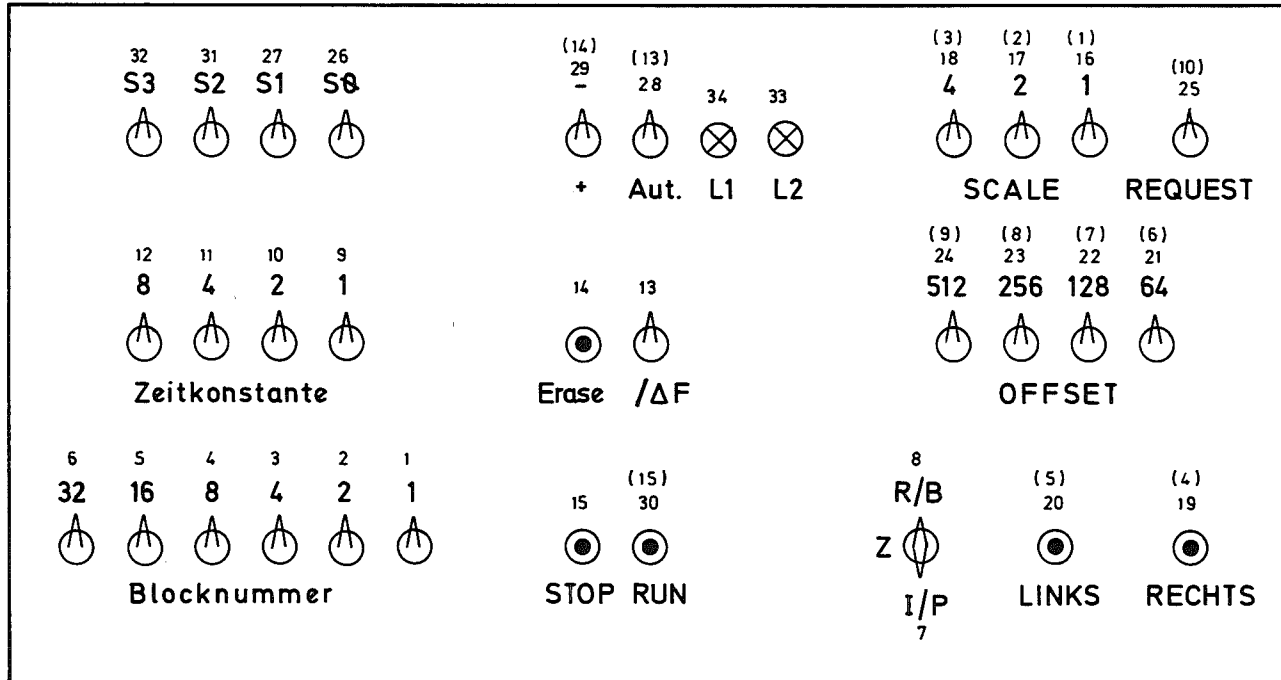


Abb.7 Flußdiagramm für WAIT und NEXT

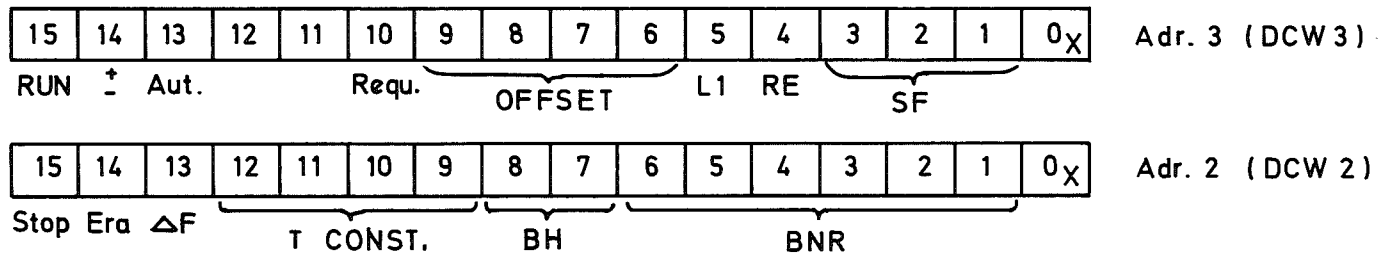
Abb. 8 STEUER - KONSOLE



mit ( ) = 36 pol. AMPHENOL  
 PIN 36 MASSE  
 " 34 + 5V

ohne ( ) = 50 pol. AMPHENOL  
 PIN 50 MASSE  
 " 48 + 5V

-  ≙ KIPPSCHALTER
-  ≙ KIPPSCHALTER mit MITTELSTELLUNG
-  ≙ TASTER
-  ≙ LEUCHTDIODE



SKLD	
I(N/2-1)	
R(N/2-1)	
Block-Nr. e+1	
lsb's	
I(1)	
R(1)	
R(N/2)	
DC	
I(N/2-1)	
R(N/2-1)	
Block-Nr. e	
msb's	
I(1)	
R(1)	
R(N/2)	
DC	

SALD	
Block-Nr. e+1	
frei	
R(N/2-1)	lsb
R(N/2-1)	msb
Block-Nr. e	
R(1)	lsb
R(1)	msb
DC	lsb
DC	msb

Abb.9 Anordnung der SKLD- bzw. SALD-Daten