

**Heinrich-Hertz-Institut
für Nachrichtentechnik
Berlin GmbH**

Technischer Bericht Nr. 203

LIFORT

ein Programmsystem zur
Umsetzung von LIDIA-Pro-
grammen nach FORTRAN IV

von

Dipl.-Inf. Herbert Gold

September 1978

Technischer Bericht Nr. 203

L I F O R T

ein Programmsystem zur Umsetzung von
LIDIA-Programmen nach FORTRAN IV

Zusammenfassung:

Der vorliegende Bericht beschreibt ein Programmsystem, mit dem Dialogprogramme, die in der Sprache LIDIA (Lernen im Dialog) realisiert wurden, in die Programmiersprache FORTRAN IV überführt werden können. Es wird dadurch möglich, Inhalte aus dem Bereich "Rechnergestützter Unterricht" für das Laborprojekt Zweiweg-Kabelfernsehen nutzbar zu machen.

Bearbeiter:

Herbert Gold

(Dipl.-Inf. Herbert Gold)

H. Ohnsorge

Dr. Horst Ohnsorge
(Wiss.Techn.Geschäftsführer)

Vöge

Dr. Karl Heinrich Vöge
(Abteilungsleiter)

Berlin-Charlottenburg, den 30.9.1978

ISSN 0342 - 6815

1 Einführung

Die Entwicklung der in den Bereichen Datenverarbeitung, kabelgebundene Breitband-Kommunikation und Audiovision verfügbaren Technologien haben seit Jahren zu einer breiten Erörterung individueller und interaktiver Dienste am erweiterten Heimfernseher geführt /2, 4, 8/. Es wurde dabei deutlich, daß ein Farbfernsehgerät ergänzt mit alphanumerischer Eingabetastatur, welches über ein Kabel an einen zentralen Rechner mit digitalen und audiovisuellen Speichern angeschlossen wird, in jedem Haushalt eine große Mannigfaltigkeit von neuen Diensten technisch möglich machen würde.

Bei der im Heinrich-Hertz-Institut für Nachrichtentechnik in Berlin (HHI) laufenden Entwicklung eines Zweiweg-Kabelfernseh-Laborsystems (ZKTVL) - als Vorläufer für Feldversuche - werden erste Experimente mit diesen neuen Telekommunikationsformen (TKFen) durchgeführt. Hierbei kommt es darauf an, möglichst viele vorhandene Erfahrungen und Inhalte, z.B. im Bereich "Lernen", zu nutzen. Eine Möglichkeit hierfür bot sich an durch das im Rahmen des 1. und 2. DV-Programms des BMFT in den Jahren 1971 bis 1975 am staatlichen Gymnasium bei St. Anna, Augsburg, geförderte Projekt zum computergestützten Unterricht (CUU). Dieses Projekt sollte den CUU im Schulalltag realisieren. Hierzu wurde auf einer Siemens-Anlage S4004/45 mit 64 kB unter BS1000 die Autorensprache LIDIA (Lernen im Dialog) /5/ angewandt.

Es wurden eine Reihe von CUU-Programmen aus den Gebieten

- Mathematik,
- Naturwissenschaften und
- Fremdsprachen

entwickelt und getestet /6/. Dabei entstanden Programmarten, die ein breites Spektrum umfassen:

- Übungsprogramme,
- tutorielle Programme,
- Spielprogramme,
- Simulation,
- "entdeckendes Lernen" u.v.m.

Durch die Unterstützung von Herrn Dr. Keil (Projektleiter des o.g. Projekts) erhielt das HHI einen Teil dieser Courseware, wofür wir ihm an dieser Stelle unseren Dank ausdrücken wollen.

Da im ZKTVL des HHI eine einheitliche und möglichst portable Implementation aller TKFen angestrebt wird, andererseits die Fülle (über 100 Lektionen) von LIDIA-Programmen nutzbar gemacht werden sollten, war eine Umsetzung in eine andere Implementierungssprache notwendig. Es war daher naheliegend, die Umsetzung mit Hilfe eines in FORTRAN IV geschriebenen Übersetzers so durchzuführen, daß das Produkt ein FORTRAN IV-Programm wird.

Dies leistet das Programmsystem LIFORT, das mit einem Aufwand von ca. 3 Mannmonaten erstellt wurde und Courseware im Umfang von über 100 Programmstunden einschließt. Die Umsetzung in eine andere Sprache (z.B. PASCAL) ist mit LIFORT relativ leicht zu bewerkstelligen.

Wir bedanken uns herzlich bei Herrn Dipl.-Ing. Mahnkopf für die Unterstützung in verschiedenen Phasen der Arbeit und bei Herrn cand. Inf. Geiß für einen großen Anteil der Implementierung.

2 LIFORT

2.1 Ausgangsbasis

Die Autorensprache LIDIA ist formatgebunden und zeilenorientiert /6/. Die wichtigsten Anweisungen sind:

- LADE - für die Zuweisung von Zahlen, logischen Werten, Text und Marken an entsprechende Register.
- BILD - für die Ausgabe von Texten am Terminal. Es besteht hierbei die Möglichkeit, den augenblicklichen Inhalt von Registern einzufügen ('splits').
- LIES - für Antworten/Eingaben des Benutzers, sie werden ebenfalls entsprechenden Registern zugewiesen.
- ANTW - temporäre Register halten in Verbindung mit logischen Variablen erwartete Antworten des Benutzers bereit.
- TEST - die Eingabe wird mit den Antwort-Registern verglichen und die log. Variablen (Antwortindikatoren) entsprechend gesetzt.
- WENN - erlaubt die Abfrage von Schaltern, Antwortindikatoren und Zahlenwerten. Die folgende LIDIA-Anweisung wird nur ausgeführt, wenn sich der Wert "ein" ergibt.
- NACH - führt einen Sprung zu einer Marke aus.
- TSPR - "Teste und springe" ist die Zusammenfassung der Funktionen LIES, ANTW, TEST, WENN und NACH.
- RTSPR - wie TSPR, jedoch ohne LIES.
- UPRO - Sprung in ein Unterprogramm.
- UEND - Rücksprung aus einem Unterprogramm.
- ENDE - Ende des LIDIA-Programms.

Bei dem großen Umfang der genannten Aufgabe bietet sich eine automatisierte Umsetzung oder ein Interpreter an. Obwohl die interpretative Struktur von LIDIA zu einer Laufzeitinterpretation zwingt, z.B. bei Sprüngen über Markenregister und bei Unterprogrammaufrufen ist der Aufwand bei einem Umsetzer durch geschickte Nutzung der Zielsprache geringer als bei einem vollständigen Interpreter. Auch ergeben sich kürzere Laufzeiten der Produkte. Der Umsetzer wird außerdem bei den syntaktisch und semantisch korrekten LIDIA-Quellen sehr einfach.

2.2 Konzept

Aus dem LIDIA-Programm und LIFORT-PRELUDE generiert der Übersetzer in zwei Läufen ein FORTRAN-Programm. Etwa auftretende Fehler werden am Terminal protokolliert und im FORTRAN-Programm markiert. Die PRELUDE enthält Programmteile, die bei allen Übersetzungen gleich sind, sie dient somit zur schnelleren Programmproduktion und leichten Adaptierbarkeit.

Das erzeugte FORTRAN-Programm muß übersetzt und anschließend mit den vorübersetzten Routinen aus dem Laufzeitsystem, der Grundsoftware des Terminals /1, 3/ und der FORTRAN-Standardbibliothek zusammengebunden werden. Das Resultat des gesamten Prozesses ist ein abgebundener, ausführbarer Modul, der semantisch das gleiche Verhalten wie die LIDIA-Quelle zeigt (s. Bild 1)./13/

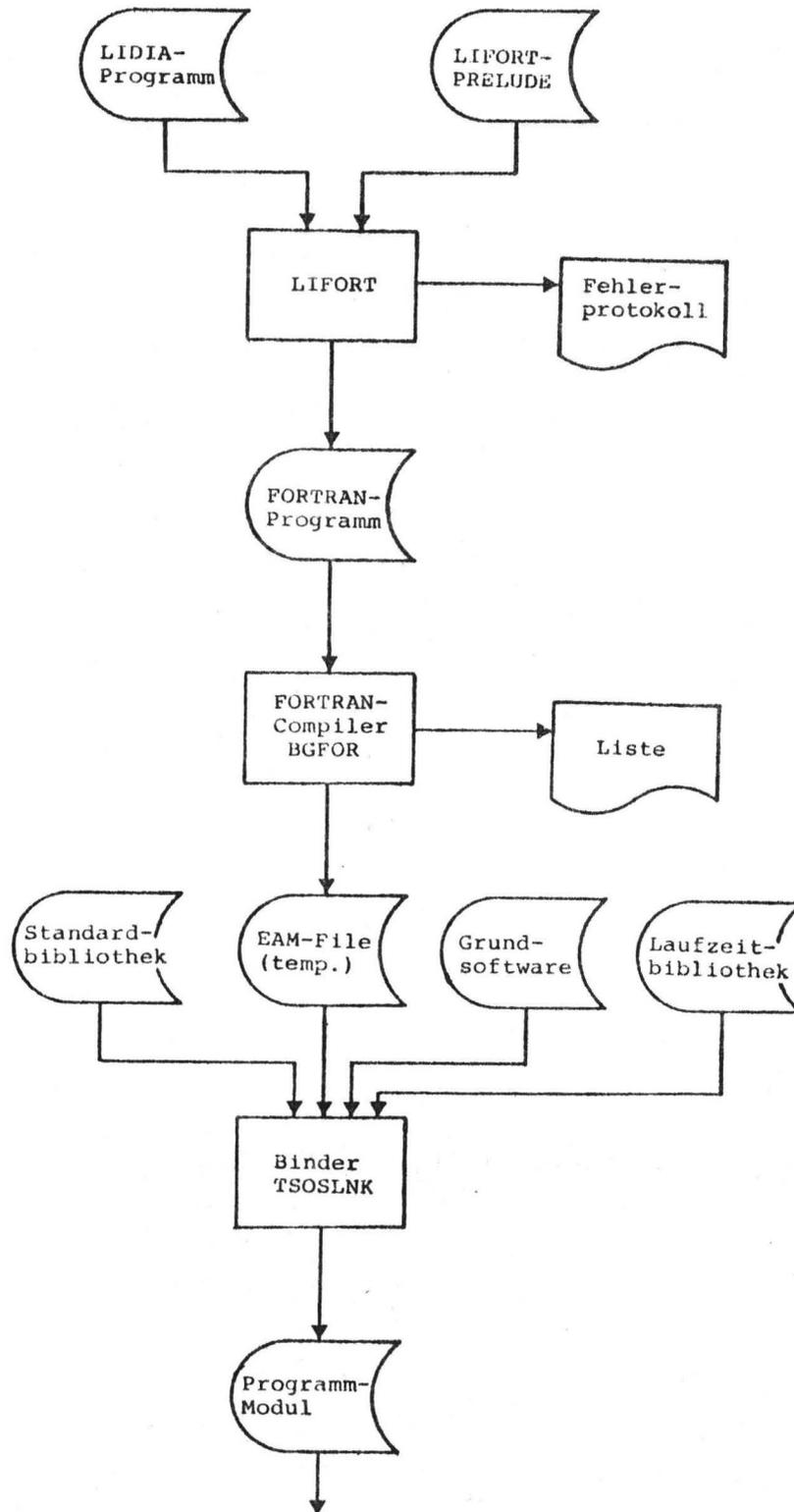


Bild 1: Ablaufschema von LIFORT /13/

2.3 Entwurfsaspekte

Für notwendige Anpassungen und Änderungen soll das FORTRAN-Programm die Quelle erkennen lassen. Leichte Adaptierbarkeit an veränderte Systembedingungen (Wechsel des Terminals) ist erwünscht und problemlose Portabilität wird durch ANSI-Standard FORTRAN IV gewährleistet.

Es wurden folgende Festlegungen für den LIFORT-Umsetzer getroffen:

- 1 : 1 Umsetzung, d.h. jedem LIDIA-Statement entspricht eine FORTRAN-Anweisung bzw. ein FORTRAN-Unterprogrammaufruf.
- Die Bezeichnungen der LIDIA-Befehle werden als Unterprogrammnamen beibehalten. Bei den stark kontextabhängigen Anweisungen LIES und LADE läßt sich dies allerdings nicht immer durchführen. Ausnahmen sind daher:
Lesen von Zahlen, als LIESZ
Laden von Markenregistern aus Textregistern als LADM.
Das Laden von Zählern, Schaltern und konstanten Marken ergibt direkte FORTRAN-Zuweisungen.
- Die Fortsetzungskarten der Bildanweisungen werden zu eigenen Bildaufrufen.
- Die Fortsetzungskarten der TSPR-Anweisung werden zu RTSPR-Anweisungen.
- Es wird kein Adressatenprotokoll erstellt.
- Die Möglichkeit, aus LIDIA andere LIDIA-Hauptprogramme zu rufen und die logische Funktion "implizit", werden nicht übersetzt, es erscheinen selbstverständlich Fehlermeldungen.
- Es ist kein Tischrechnermodus vorhanden.

2.4 Die Übersetzungsphasen

Der Übersetzer arbeitet in zwei Phasen: Im ersten Lauf werden alle Markendefinitionen des Quellprogramms in eine interne Liste übertragen. Der zweite Durchgang dient der eigentlichen Übersetzung. Er beginnt mit dem Kopieren der LIFORT-PRELUDE in die Produktfile. Sie enthält die Definitionen der Variablenbereiche, die den Anschluß an die Laufzeitroutinen erlaubt, der Arbeitsbereiche und den Aufruf einer Initialisierungsroutine.

Nach diesen Initialisierungen werden sequentiell alle LIDIA-Quellzeilen von der Plattendatei gelesen und übersetzt (Ausgabe in die Produktdatei). Die Bearbeitung geschieht im allgemeinen durch zeichenweise links-rechts-Parsieren. Eine Ausnahme bildet die WENN-Anweisung, wo ein Vorgriff um mehrere Zeilen notwendig werden kann.

Nach Umsetzung von ENDE wird ein Sprungverteiler für die im ersten Lauf gelesenen Marken und für die im Laufe der Übersetzung zusätzlich generierten Marken erzeugt. Die Ausgabe wird unterdrückt, wenn weder ein LIDIA-Unterprogrammssprung noch ein Sprung über ein Markenregister ("LADM") im Programm vorkommt.

Da LIDIA-Unterprogramme auch ohne Aufruf aus dem Hauptprogramm durchlaufen werden können, ist das automatische Abtrennen und getrennte Übersetzen der Unterprogramme nicht möglich, jedoch können manuell mittels eines Text-Editors LIDIA-Programmteile, die ein in sich geschlossenes Segment bilden, abgetrennt und als FORTRAN-Unterprogramm übersetzt werden. Jedes LIDIA-Hauptprogramm wird an der "START"-Anweisung am Programmanfang erkannt. Jeweils die erste Marke (nur 6 Zeichen) werden als Programm- bzw. Unterprogrammname verwendet. (Weitere Details in Punkt 3.4)

Hinter dem erzeugten Programm werden noch in einem "BLOCK DATA" die LIDIA-Marken ausgegeben, um bei einem Sprung über den textuellen Inhalt eines Markenregisters den Index für den Sprungverteiler auffinden zu können. Diese Ausgabe unterbleibt ebenfalls, falls kein

Sprung über ein Markenregister vorkommt, das seinen Wert aus dem Textstring der Marke (durch den Aufruf von LADM) erhalten hat. Das Layout des COMMON-Bereichs ist so, daß die Zuordnung von Markennamen und Labelnummer leicht getroffen werden kann.

2.5 Das Laufzeitsystem

Das Laufzeitsystem enthält neben den namensgleichen Unterprogrammen für LIDIA-Anweisungen verschiedene Routinen zur Umcodierung, z.B. Umwandlung einer Zahl in einen Text, sowie für die Ein- und Ausgabe.

2.6 Adaptierbarkeit

Anpassungen an eine andere Systemumgebung sind nur im Laufzeitsystem und der PRELUDE vorzunehmen und deshalb mit geringem Aufwand durchführbar. So wurde z.B. die Textausgabe auf einem Bildschirm von 21 mal 48 Zeichen /1/ gegenüber vorher 20 mal 54 Zeichen /7/ durch entsprechende Programmierung der BILD-Routine möglich. Zur Verbesserung des Layouts werden Trennungen im LIDIA-Text erkannt und unterdrückt, falls sie nicht mehr notwendig sind.

Alle Fehlermeldungen des Übersetzers werden durch die Ausgaberroutine ERRMSG durchgeführt. Damit erfolgen fast alle Terminalausgaben während der Übersetzung eines LIDIA-Programms von zentraler Stelle. Bei der Ausgabe des FORTRAN-Programms kommt nur das Unterprogramm PUCARD zur Anwendung (Ausnahme: Ausgabe der PRELUDE im UP FIRSTP). Diese Maßnahmen und die weitgehende Modularisierung des Umsetzers erleichtern die Übertragung auf andere Anlagen.

2.7 Ergänzungen

In einer Überarbeitung des Systems wurden einige Ergänzungen vorgenommen, die sich an /9/ orientieren. Diese sind die - oben bereits angesprochen - automatische Unterdrückung des Sprungverteilers und

des Block-Datums, sowie die Möglichkeit, aus LIDIA-Quellen auch FORTRAN-Unterprogramme zu generieren, wodurch beim praktischen Einsatz kürzere Übersetzungszeiten erreicht werden.

- Der Einbau von Realzahlen verbessert insbesondere die formatgebundene Ausgabe. Das Quellprogramm wird dabei deutlich kürzer und verständlicher.
- Um automatisches Löschen des Bildschirms zur Laufzeit zu verhindern, wird Spalte 9 der BILD-Anweisung interpretiert. Ein "*" bedeutet: Bildschirm nicht löschen, alle anderen Zeichen bedeuten: Bildschirm löschen.
Es wird die dynamische Reihenfolge von LIES- und BILD-Anweisungen betrachtet, so daß das erste Auftreten einer BILD-Anweisung während der Laufzeit maßgebend ist.
- Laufzeitfehler bei LADE-Anweisungen können im LIDIA-Programm ausgewertet werden, wenn in Spalte 15 ein "." steht. Die folgende Anweisung wird nur im Fehlerfall ausgeführt. (Entspricht den LIDIA-Konventionen in /9/.)
- Für die Nutzung einer Kommandotastatur, wie sie sich aus den Überlegungen für eine benutzerfreundliche Kommandosprache ergibt /10/, wurden die TSPR- und RTSPR-Anweisung ergänzt, abweichend von der LIDIA-Konvention. Das "#"-Zeichen kann in LIDIA eine Zeichenkette in den genannten Anweisungen abschließen. In Erweiterung der Syntax kann diesem Zeichen die interne Nummer der für den positiven Vergleich geforderten Kommandotaste folgen. (Die Zeichenkette darf auch leer sein.)
Enthält eine TSPR-Anweisung diese Ergänzung, so muß die Abfrage auf das Kommando "ENDE" explizit vorgenommen werden, in allen anderen Fällen wird diese Prüfung in der LIES-Routine durchgeführt und führt zur Beendigung des Programms.

2.8 Ausblick

Der Übersetzer kann leicht um neue Anweisungen - als LIDIA-Erweiterungen - ergänzt werden, z.B. im Projekt Zweiweg-Kabelfernsehen um Anweisungen für Ton, Dias und Film. Andererseits ist das Sprachkonzept für die Programmierung "sackgassenfreier" Dialoge und einer strukturierten Programmierung weniger geeignet, so daß modernere Sprachen mit gutem Prozedurkonzept vorzuziehen sind (ELAN, LEKTOR /11, 12/.

Das Laufzeitsystem wäre wesentlich effizienter, wenn die in 2.6 erwähnte Aufbereitung des Bildschirms im Übersetzer durchgeführt würde. Dabei könnten auch einige Abfragen auf Speicherüberlauf (für den Bildwiederholtspeicher), die bisher in der Grundsoftware durchgeführt werden, übernommen werden.

Die automatisierte Unterstützung für LIDIA-Programme mit Kommandotasten (s. 2.7, letzter Punkt) würde einen komplexen Analyselauf vor der eigentlichen Umsetzung erfordern und einige Probleme aufwerfen, z.B. Kellern von Variablen für die "ZURÜCK"-Funktion etc.

3 Details der Umsetzung

3.1 Umsetzung der Variablen

Es sollen die ursprünglichen Bezeichnungen für die Variablen erhalten bleiben. In der Datei "G.F.LIFORTPRELUDE" ist ein Common-Block /LIDIA/ definiert, in dem die Variablen wie folgt stehen:

- a) A1, A2, A3, A4, A32
Logische Variable für die Indikatoren bei Antwortanalysen.
- b) S1, S2, S3, S4, S32
Logische Variable als programmierbare Schalter.
- c) Z1, Z2, Z3, Z16
Integer*4 Variable für die Zähler.
- d) T1, LT1, T2, LT2, T3, LT3, T6, LT6, IT, LIT
Jedes Tx ist ein Integerfeld der Dimension 12, zur Aufnahme von Texten in gepackter Form(4 Zeichen/Wort).
Jedes LTx enthält die Länge des Textausdruckes Tx.
(IT und LIT sind für Texte ohne explizite Registerangabe bei LIES und TEST.)
- e) M1, M2, M3, M6
Variable, die Integerwerte beinhalten, die zur Verzweigung über computed GOTO's dienen. Sie entsprechen den Markenregistern.
- f) R1, R2, R3, R4, R32
Variable für Realzahlen.

3.2 Umsetzung der Anweisungen

Die Quellzeilen werden sequentiell gelesen und analysiert und in die entsprechenden FORTRAN IV-Anweisungen umgesetzt: zuerst wird die eingelesene Zeile daraufhin untersucht, ob es sich um einen Kommentar handelt. In diesem Fall wird eine FORTRAN IV-Kommentarkarte erzeugt. Enthält die Quellzeile dagegen eine Markendefinition, so wird die entsprechende FORTRAN IV-Marke erzeugt, anderenfalls bleibt das Markenfeld der Ausgabekarte leer. Anschließend wird der Operationscode bestimmt und auf die Programmstelle zur Übersetzung der entsprechenden LIDIA-Anweisung verzweigt.

Alle Texte werden grundsätzlich in zwei Parameter überführt, der erste gibt die Länge des Textes an, der zweite enthält den String selbst. Dabei kennzeichnet der Längenwert zusätzlich den Typ des Strings: bei positiver Zahl handelt es sich um eine Textkonstante, bei negativem Wert um einen Textausdruck, d.h. der String enthält Splits, die zur Laufzeit durch die aktuellen Registerwerte ersetzt werden müssen.

Die einzelnen LIDIA-Anweisungen werden wie folgt übersetzt:

BILD: Die BILD-Anweisung wird in einen Aufruf
`"CALL BILD (prv, pst, skp, spc, spl, len, lenHtext)"`
für jede Zeile der BILD-Anweisung umgesetzt. Dabei ist für prv beim 1. Aufruf und bei allen Aufrufen, bei denen skp oder spc größer als 0 sind "1", bei den anderen "-1" einzusetzen, für pst ist immer "-1" einzusetzen. Die beiden Werte sind dafür vorgesehen, in BILD-Aufrufen Steueranweisungen nach der Übersetzung leicht von Hand einbringen zu können, falls dies notwendig wäre. skp und spc erhalten den Wert der Skip- bzw. Space-Angabe der BILD-Anweisung, spl gibt die Position des ersten Splits im Text an, len erhält die Längenangabe des Textes der BILD-Anweisung und text den Text selbst.

- LIES:** Die LIES-Anweisung wird je nach ihrer Form in der LIDIA-Quellezeile in "CALL LIES (IT)" oder "CALL LIES (Tnn)" zum Einlesen von Texten bzw. in "CALL LIESZ (Znn)" oder "CALL LIESR (Rnn)" zum Einlesen von Zahlenwerten umgesetzt, wobei nn immer eine Registernummer bedeutet.
- ANTW:** Die ANTW-Anweisung wird in den FORTRAN IV-Aufruf "CALL ANTW (nn, len, lenHtext)" umgesetzt, wobei nn die Nummer des Antwortregisters, len die Länge des Textes und text den Text darstellen.
- TEST:** Die TEST-Anweisung wird je nach ihrer Form in der LIDIA-Quellezeile in "CALL TEST (IT)" bzw. "CALL TEST (Tnn)" umgesetzt, wobei die Aufrufe "CALL EXAKT", "CALL KONS", "CALL BLANKS", "CALL STW" und "CALL PROZ (prz)" vorausgehen können. Sie setzen entsprechende Parameter für den Vergleichsmodus.
- WENN:** Die WENN-Anweisung stellt eine Besonderheit bei der Umsetzung dar. Da jede LIDIA-Anweisung unter die WENN-Bedingung gestellt werden kann, und somit auch Anweisungen, deren Umsetzung mehrere FORTRAN IV-Zeilen erzeugen oder die mit einer Marke versehen sind, müssen daher je nach Form der LIDIA-Quelle zwei verschiedene Formen erzeugt werden. Außerdem kann hierbei ein Vorriff um einige Quellenweisungen notwendig werden.
- Die WENN-Anweisung wird in die IF-Anweisung überführt und ggf. eine Flagge gesetzt, die die Ausgabe eines " LABEL CONTINUE" vor der Umsetzung der übernächsten Quellenweisung veranlaßt.
- NACH:** Die NACH-Anweisung wird in eine GOTO-Anweisung überführt, wobei das Sprungziel entweder die der LIDIA-Marke entsprechende FORTRAN IV-Marke oder die Nummer des angegebenen Markenregisters ist.

- TSPR: Die TSPR-Anweisung wird nur in der ersten Zeile in die Form
 "CALL TSPR (&lab, len, lenHtext, koman)"
 übersetzt. Alle weiteren Zeilen werden wie die RTSPR-Anweisung behandelt, in der auch die Bedeutung der oben aufgeführten Parameter angegeben wird.
- RTSPR: Die RTSPR-Anweisung wird in eine Folge von
 "CALLRTSPR (&lab, len, lenHtext, koman)"
 für jede ihrer Quellzeilen übersetzt. Dabei bedeuten lab die der LIDIA-Marke entsprechende FORTRAN IV-Marke bzw. die Nummer des Markenregisters, len die Länge des Textes und text den Text. koman ist die Nummer der Kommandotaste, Ø für beliebig bzw. undefiniert.
- LADE: Die LADE-Anweisung wird in die Form
 "rnn = ..."
 überführt, wobei r eine Registerbezeichnung (Z, S, A, R, M) und nn eine Registernummer bedeutet. Die Anweisung zum Laden von Textregistern wird dagegen in die Form
 "CALL LADE (Tnn, len, lenHtext)"
 übersetzt, wobei nn die Registernummer, len die Länge des Textes und text den Text bezeichnet.
- LADE Mx Markenausdruck
 wird umgesetzt in Mx = generierter Labelwert (Schrittweite 10), wenn der Markenausdruck eine Marke ist. Falls LADE Mx = My wird daraus Mx = My. Soll das Markenregister mit einem Textregister geladen werden, so erfolgt die Umsetzung durch einen Funktionsaufruf der Form:
 Mx = LADM (Ty).

LADE FORMAT Formatspezifikation

wird umgesetzt in

CALL LADEF (ns, nz, nn, kn, kz, ks, ne).

ns, nz, nn sind die Anzahl der Sterne, Z-Zeichen und 9-Zeichen vor dem Dezimalkomma /-punkt.

kn, kz, ks die umgekehrte Reihenfolge der Zeichen nach dem Komma (entsprechend den LIDIA-Konventionen).

ne ist bei Exponentendarstellung größer - gleich 1.

UPRO: Die UPRO-Anweisung wird in die Form

"CALL UPRO (rlb, &lab)"

übersetzt, wobei rlb eine neu generierte FORTRAN IV-Marke für die Rücksprungadresse und lab entweder die der LIDIA-Marke entsprechende FORTRAN IV-Marke oder die Nummer des Markenregisters ist. Außerdem wird eine Flagge gesetzt, die die Ausgabe von

"rlb CONTINUE"

in der nächsten Zeile steuert.

UEND: Die UEND-Anweisung wird in einen Aufruf

"CALL UEND (&99998)"

übersetzt.

HALT: Da es keine einfache Lösung für die Umsetzung der HALT-Anweisung gibt, wird sie wie eine STOP-Anweisung übersetzt.

STOP: Die STOP-Anweisung wird in einen Sprung

"GOTO 99999"

übersetzt. Unter dieser FORTRAN IV-Marke steht im übersetzten Programm eine STOP-Anweisung.

PROG: Die PROG-Anweisung wird in einen Unterprogrammaufruf

"CALL" übersetzt.

Nach Abarbeitung aller Quellzeilen wird der Sprungverteiler für die umgesetzten und generierten Marken ausgegeben. Anschließend wird das "BLOCK DATA"-Unterprogramm zur Initialisierung des Markenfeldes ausgegeben und damit die LIDIA nach FORTRAN IV-Umsetzung beendet.

Der Umsetzer ist weitgehend modularisiert worden, so daß z.B. das Hauptprogramm im wesentlichen nur noch aus Steueranweisungen und Unterprogrammaufrufen besteht. Da in der Implementierungssprache FORTRAN IV keine Möglichkeit der zeichenweisen Ein/Ausgabe besteht, wird die Ein/Ausgabe zwischengepuffert und auf diese Puffer zum Teil mit entsprechenden Unterprogrammen, zum Teil aber auch direkt zugegriffen. Eine Übertragung auf andere Anlagen dürfte kaum Änderungen im Übersetzer und nur wenige Änderungen im Laufzeitsystem und dem PRELUDE erfordern.

3.3 Arbeitsbereiche

Das Laufzeitsystem benötigt einen permanenten Speicherbereich, über den die genannten Unterprogramme kommunizieren. Im Common-Block /LIDIA/ sind dazu neben den LIDIA-Variablen noch folgende Felder und Variablen definiert worden:

32 Antwortregister und die Länge der Antwort als ANT (13, 32),
 NAN die Zahl der gesetzten Antwortregister,
 20 Positionen für Adressateneingaben POS (20, 21).
 IP für die Anzahl der Positionen
 10 Rückkehradressen der Unterprogrammssprünge LABSAV (10),
 UPRCNT die Zahl der Unterprogrammaufrufe,
 SPVLAB der Index für den Sprungverteiler,
 EIN, AUS sind logische Konstanten, entsprechend "true" und "false",
 NUMLI, NUMCHR geben die Cursorposition Zeile/Spalte an,
 LCHR dient zur Steuerung aufeinanderfolgender BILD-Anweisungen,
 NPROZ Prozentzahl für TEST n%, sonst n=0,
 LEX, LKO, LBL, LST logische Werte zur Steuerung der TEST-Abfragen,
 ERR, POINT logische Werte zur Fehlerbehandlung,
 CLEAR zur dynamischen Steuerung des Bildschirmlöschens.

Beim Umsetzer sind die Common-Bereiche zu funktionalen Blöcken gruppiert.

Der Common-Block /CONSTS/ enthält alle Zeichen- und Stringkonstanten, wobei möglichst sprechende Bezeichnungen gewählt wurden, z.B. ANTW für die Zeichenkette "ANTW", die zur Erkennung der gleichnamigen LIDIA-Anweisung benötigt wird.

Im Common-Block /LIDVRS/ stehen alle diejenigen Konstanten, die durch den Variablenbereich einer bestimmten LIDIA-Version vorgegeben sind. Dies sind:

ZNMBRS	-	Zahl der Integerzahlen Z
SNMBRS	-	Zahl der Schalter S
ANMBRS	-	Zahl der Antwortregister A
TNMBRS	-	Zahl der Textregister T
MNMBRS	-	Zahl der Markenregister M
RNMBRS	-	Zahl der Realwerte R

Alle Prüfroutrinen des Umsetzers beziehen die Vergleichswerte für Bereichsabfragen aus diesem Common.

Der Block-Common /LABEL/ enthält das Feld LABELS zur Speicherung der Marken im LIDIA-Programm (Pass 1), LABMAX ist die Maximalzahl der Marken, LABCNT der Markenzähler, NSPVT steuert die Ausgabe des Sprungverteilers.

Die Common-Blöcke /IN/, /OUT/ und /MOUT/ enthalten Kanalnummern, Zähler und Felder jeweils für die Eingabe, die Ausgabe des FORTRAN-Programms und für die Meldungen am Terminal.

Alle Konstanten werden durch BLOCK DATA-Anweisungen gesetzt. Ebenso werden fast alle notwendigen Initialisierungen für Variable vorgenommen.

3.4 Anwendungshinweise

Zur Ausführung einer Umsetzung von LIDIA-Quellen nach FORTRAN IV-Programmen auf der Siemens-Anlage 7.748 muß die Quelle auf einer ISAM-Datei stehen und mindestens bis zur Spalte 72 Zeichen enthalten.

Ein LIDIA-Hauptprogramm ist für den Umsetzer an der START-Anweisung erkennbar. Der Programm- bzw. Unterprogrammname wird aus den ersten sechs Zeichen der ersten Marke gebildet. BLOCK-DATA-Anweisungen bei Unterprogrammen sind als DATA-Anweisungen in das Programm (Zeile 32) zu kopieren.

Umfangreiche Computed-GOTOs in den Programmen sind ebenfalls an den Programmanfang zu kopieren.

Einige Prozeduren, die die Durchführung einer Umsetzung erleichtern, erfordern feste Konventionen in der Benennung der Dateien:

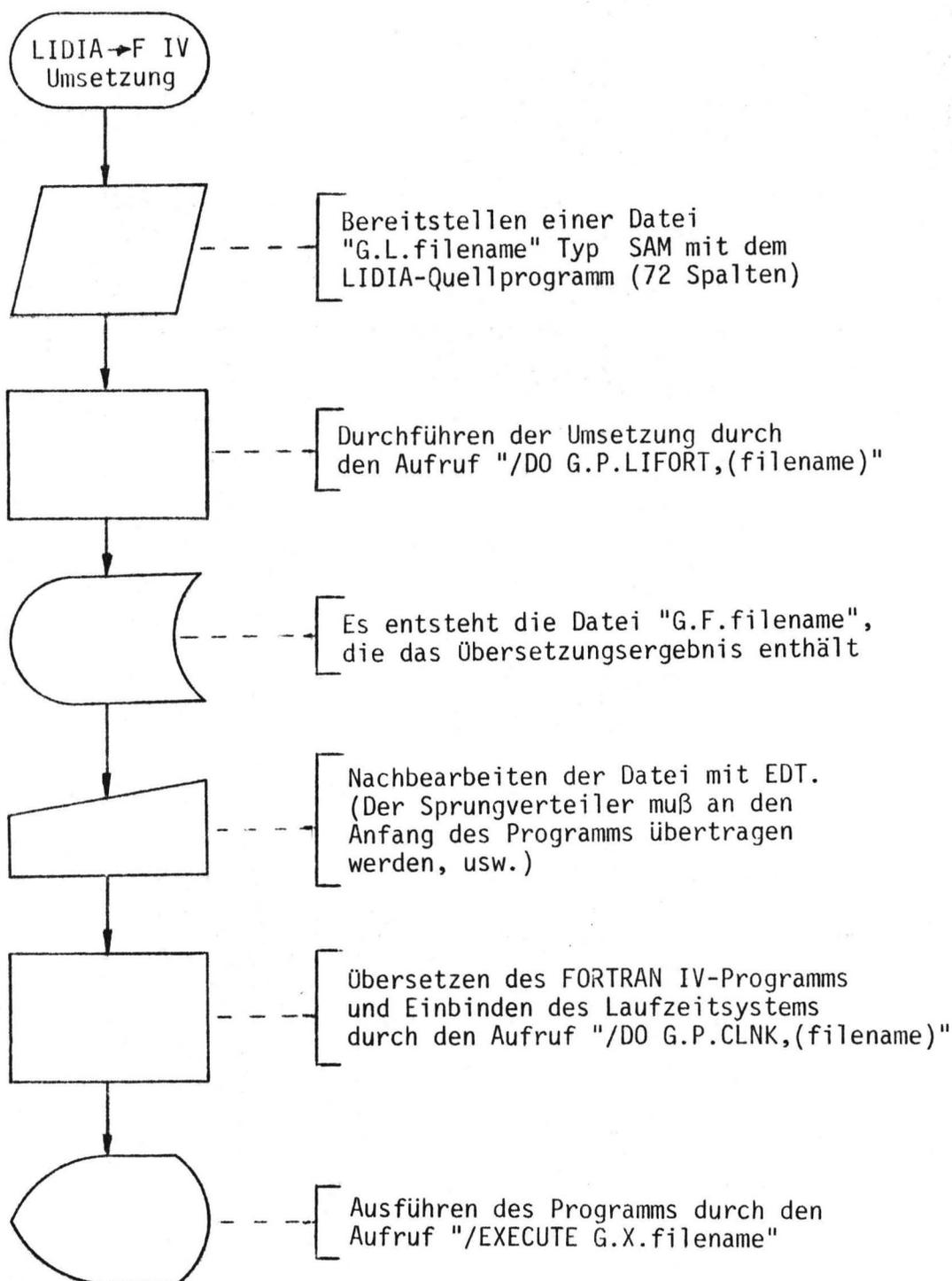
- G.L. filename für die LIDIA-Quellen
- G.F. filename für die FORTRAN-Dateien
- G.X. filename für die ausführbaren Moduln
- G.P. procname für die Prozeduren
- G.LIB.LIFORT für die LIFORT-Bibliothek

Den Ablauf der Umsetzung und die Ausführung zeigt Bild 2.

Der LIFORT-Umsetzer selbst kann mit `"/DO G.P.CLNK,(LIFORT),ID=GU."` aus der Quelldatei GU.F.LIFORT generiert werden.

Unterprogramme des Laufzeitsystems und aus LIDIA-Quellen werden mit `"/DO G.P.LIB.LIFORT,(filename)"` in die Bibliothek "G.LIB.LIFORT" gespeichert.

Bild 2 : Ablauf der Programmumsetzung LIDIA nach FORTRAN IV auf der Rechenanlage S 7.748 unter BS 2000



3.5 LIFORT-Fehlermeldungen

Fehlermeldungen erfolgen jeweils mit der fehlerhaften Zeile und einem Zeiger auf die wahrscheinliche Position des Fehlers.

- (1) "' : ' MISSING"
In einer TSPR- oder RTSPR-Anweisung fehlt der Doppelpunkt. Die Anweisung wird nicht übersetzt!
- (2) "' =' INSERTED"
Es wurde ein Gleichheitszeichen, das zur richtigen Syntax notwendig ist, eingefügt.
(Bei ANTW- u. LADE-Anweisungen)
- (3) "COMMENT AFTER 'WENN'"
Nach einer WENN-Anweisung darf keine Kommentarkarte folgen!
- (4) "DUPLICATE LABEL"
Mehrfaches Auftreten einer Marke in Spalte 1-8.
- (5) "END OF FILE REACHED WHILE TRANSLATION OF 'WENN'"
Unvollständiges LIDIA-Programm.
- (5b) "HALT OR STOP TRANSLATED TO STOP"
Hinweis, daß LIDIA-Anweisung HALT oder STOP als .ORTRAN-STOP übersetzt wurden.
- (6) "ILLEGAL LABEL CALL"
PULAB wurde mit einem ungültigen Markenwert aufgerufen (Systemfehler)
- (7) "ILLEGAL OPERATION"
Die Zeichen in Spalte 10-19 konnten nicht als LIDIA-Anweisung identifiziert werden.

- (8) "ILLEGAL SKIP"
Eintrag in Spalte 70 einer Bildanweisung (Übersprünge *Zeilen*) liegt nicht im Bereich
 \emptyset , 1 ... 9, A, B ... F
- (9) "ILLEGAL SPACE"
Wie bei SKIP, nur für Leerzeilen (Spalte 71)
- (10) "IMPLIZIT"
Ein Ausdruck enthält "IM" und wird nicht übersetzt.
- (11) "INDICATOR MISSING"
In der Antwortanweisung fehlt der Antwortindikator
A1 ... A32 oder liegt außerhalb des gültigen
Bereichs.
- (12) "LABEL NOT FOUND"
Eine undefinierte Marke wird angewendet.
- (13) "LIES IN IT ASSUMED"
Es wurde nach einer LIES-Anweisung eine nicht weiter
identifizierbare Zeichenkette gefunden und daher
LIES angenommen, d.h. CALL LIES (IT) produziert.
- (13b) "LOOK AHEAD STACK OVERFLOW, CHANGE 'SAVDAT'"
Der Keller für Eingabezeiten im Vorgriff bei der
Umsetzung der WENN-Anweisung ist zu klein. Änderung
in UPS/INCARD und SAVE. Führt zum Programmabbruch.
- (14) "NO ERROR EXIT USABLE"
Ein Punkt in Spalte 15 (Fehlerausgang) ist bei
dieser Anweisung nicht zulässig!
- (15) "NO LABEL GIVEN"
Es fehlt die Angabe einer Marke (Syntaxfehler).



- (16) "NO REGISTER GIVEN"
Nach 'LIES IN' fehlt eine gültige Registerangabe.
- (17) "OPERATION 'RTSPR' ASSUMED"
Es wurde nur "RTSP" erkannt und RTSPR angenommen.
- (18) "PROGRAM NAME MISSING"
Das Ziel einer PROG-Anweisung fehlt.
- (19) "QUOTATIONS NOT CLOSED"
Häufige "Fehlermeldung", wenn nur ein Apostroph gefunden wurde, jedoch ein weiterer z.B. für ein Textsplit erwartet wird.
- (20) "TEXTREG. MISSING"
Es fehlt die Angabe eines gültigen Textregisters (nach 'TEST IN').
- (21) "TEXTREGISTER NOT ALLOWED"
Unterprogramm sprung über Register nicht zulässig (z.B. weil keine Marken im Programm vorhanden sind).
- (22) "SKIP AND/OR SPACE NOT ALLOWED"
Einträge in Spalte 70 oder 71, bei anderen als der "BILD"-Anweisung.
- (23) "SPACE TOO SHORT FOR NUMBER"
Der Ausgabebereich für eine Zahl ist zu klein. Eine Systemfehlermeldung, die in der laufenden LIFORT-Version nicht auftritt. (UP:PUINT)
- (24) "STATEMENT ISN'T TRANSLATABLE"
Eine Anweisung kann nicht übersetzt werden.

(25) "TOO MANY LABELS. CHANGE LIFORT PROGRAM!"

Tabelle für die Speicherung der Marken (das Feld LABELS im Common/LABEL/ ist zu klein); führt zum Programmabbruch.

(26) "WENN FOLLOWS WENN"

Es folgen zwei WENN-Anweisungen aufeinander.
Dies ist in LIFORT nicht gestattet!

4 Beispiel

Anhand der Gegenüberstellung zweier Listenausschnitte soll die Arbeitsweise von LIFORT gezeigt werden/13/:

LIDIA-Programm

FORTRAN IV-Programm
(LIFORT-Ergebnis)

```

BILD DER COMPUTER BERECHNET IHNEN FUER VERSCHIEDENE ZEIT- 1
      YEN 7 DIE HOEHEN DES KOERPERS UEBER DEM BODEN (H=0).
      ES WERDEN MAX. 18 TABELLENWERTE ERRECHNET.
      WAELHEN SIE DIE SCHRITTWEITE, IN DER DIE 1
      TABELLE AUSGEGEBEN WERDEN SOLL. ZULAESSIG SIND
      SCHRITTWEITEN ZWISCHEN 0,1 UND 9,9 SEC.
      BITTE AUCH VOLLE SEKUNDEN MIT EINER STELLE HINTER 1
      DEM KOMMA ANGEBEN Z.B. 1,0 NICHT NUR 1 .

      UEND
TH *BILD SIE HABEN DIE ZEIT-HOEHE-TABELLE GEWAEHLT.
      UPRO TEXT
      LADE T6='T1'
          Z16=Z16+1
          Z14=Z15+Z12
      LIES IN T1
      LADE M3=TH
      UPRO EINNAHME
      LADE Z13=Z1
      NACH BILDTH

THDIRE LADE S32=AUS
      LADE Z14=Z15+Z12
BILDTH BILD H0='Z11' M, V0='Z12' M/SEC T H
      LADE Z10=Z10+1
      WENN Z10=19
      NACH BILDAUS
      LADE Z1=Z9
          Z3=Z1-Z1/10*10
          Z1=Z1/10
      WENN Z3<0
      LADE Z3=-Z3
      LADE T3='T1'
          Z8=Z11*10+Z12*Z9-5*Z9*Z9/10
      LADE Z2=Z8
          Z4=Z2-Z2/10*10
          Z2=Z2/10
      WENN Z4<0
      LADE Z4=-Z4
      LADE T2='T1'

      T1='T3'
      UPRO TABELLE2

      CALL BILD( 1,-1,0,1,0,51,
*51HDER COMPUTER BERECHNET IHNEN FUER VERSCHIEDENE ZEIT- )
      CALL BILD( -1,-1,0,0,0,52,
*52HTEM T DIE HOEHEN DES KOERPERS UEBER DEM BODEN (H=0). )
      CALL BILD( -1,-1,0,0,0,42,
*42HES WERDEN MAX. 18 TABELLENWERTE ERRECHNET. )
      CALL BILD( 1,-1,0,1,0,40,
*40HWAELHEN SIE DIE SCHRITTWEITE, IN DER DIE )
      CALL BILD( -1,-1,0,0,0,46,
*46HTABELLE AUSGEGEBEN WERDEN SOLL. ZULAESSIG SIND )
      CALL BILD( -1,-1,0,0,0,39,
*39HSCHRITTWEITEN ZWISCHEN 0,1 UND 9,9 SEC. )
      CALL BILD( 1,-1,0,1,0,49,
*49HBITTE AUCH VOLLE SEKUNDEN MIT EINER STELLE HINTER )
      CALL BILD( -1,-1,0,0,0,42,
*42HDEM KOMMA ANGEBEN Z.B. 1,0 NICHT NUR 1 . )
      CALL UEND( &9999B )
1080 CALL BILD( 1,-1,0,0,0,42,
*42HSIE HABEN DIE ZEIT-HOEHE-TABELLE GEWAEHLT. )

      CALL UPRO( 2480 ,&1070 )
2480 CONTINUE
      CALL LADE( T6,-4,4H'T1' )
          Z16=Z16+1
          Z14=Z15+Z12
      CALL LIES( T1 )
      M3=1080
      CALL UPRO( 2490 ,&2070 )
2490 CONTINUE
      Z13=Z1
      GOTO 1100
1090 S32=AUS
      Z14=Z15+Z12
1100 CALL BILD( 1,-1,0,0,0,-44,
*44H0='Z11' M, V0='Z12' M/SEC T H )
1110 Z10=Z10+1
      IF( Z10.EQ.19 )GOTO 1120
      Z1=Z9
      Z3=Z1-Z1/10*10
      Z1=Z1/10
      IF( Z3.LT.0 )Z3=-Z3
      CALL LADE( T3,-4,4H'T1' )
      Z8=Z11*10+Z12*Z9-5*Z9*Z9/10
      Z2=Z8
      Z4=Z2-Z2/10*10
      Z2=Z2/10
      IF( Z4.LT.0 )Z4=-Z4
      CALL LADE( T2,-4,4H'T1' )
      CALL LADE( T1,-4,4H'T3' )
      CALL UPRO( 2500 ,&1130 )
2500 CONTINUE

```

5 Einzelbeschreibung der Unterprogramme

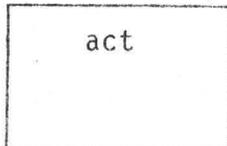
Teil I : Übersetzer (Kap. 5.1)

Teil II : Laufzeitsystem (Kap. 5.2)

Anmerkung: In jedem Teil sind die Programme alphabetisch geordnet und fortlaufend nummeriert.

Das komplexe Hauptprogramm wird durch Strukturdiagramme soweit verfeinert, daß die Grobstruktur sichtbar wird.

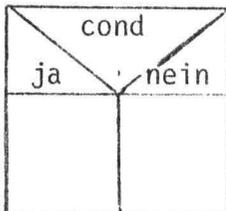
Für die Strukturdiagramme gelten folgende Vereinbarungen:



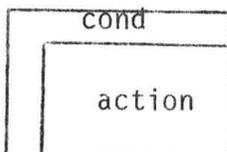
ACTION: ein Eingang (oben),
ein Ausgang (unten),
enthält Anweisungen oder andere
Strukturelemente.

act:

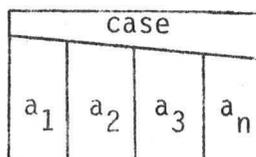
"UP" weist auf das wesentliche Unterprogramm hin
Blatt:3 weist auf neues Blatt mit weiterer Verfeinerung
hin.
Text Erläuternder Text
(leer) kennzeichnet "keine Aktion"



IF: Einfache Verzweigung
cond Erläutert die Bedingung
ja bedingte Aktionen
nein



WHILE: Wiederholung
solange Bedingung "cond" gilt
wird "action" ausgeführt.



CASE: Fallverzweigung
Eine der Alternativen " $a_1 \dots a_n$ "
wird durch "case" ausgewählt.

Deklaration der COMMON-Bereiche des Übersetzers

COMMON /CONST/

1 BILD, LIES, ANTW, TEST, WENN, NACH,
 2 TSPR, RTSP, LADE, UPRO, UEND, HALT,
 3 STOP, ENDE, PROG, SPACE, ASTRSK, APOSTR,
 4 POINT, EQUAL, COLON, CROSS, COMMA, OPEN2,
 5 CLOSE2, A, F, M, R, S,
 6 T, Z, DIGITS, START

INTEGER*4 ICHNL, INPOS, INNUM, INLIN(18), LIN(80)

COMMON /IN/

1 ICHNL, INPOS, INNUM, INLIN, LIN

INTEGER*4 LABMAX, LABCNT, LABELS(2048)

COMMON /LABEL/

1 LABMAX, LABCNT, LABELS, NSPVT

INTEGER*4 ZNMBS, SNMBS, ANMBS, TNMBS, MNMBS, RNMBS

COMMON /LIDVRS/

1 ZNMBS, SNMBS, ANMBS, TNMBS, MNMBS, RNMBS

INTEGER*4 MCHNL, LSTELN

COMMON /MOUT/

1 MCHNL, LSTELN

INTEGER*4 OCHNL, OUTPOS, OUTNUM, OUTLIN(72)

LOGICAL NEOCRD

COMMON /OUT/

1 OCHNL, OUTPOS, OUTNUM, OUTLIN, NEOCRD

INTEGER*4 SAVPTR, POS(18), NUM(18), LINES(18,18)

COMMON /SAVDAT/

1 SAVPTR, POS, NUM, LINES

Deklaration der COMMON-Bereiche des Laufzeitsystems

```
INTEGER*4  INPUT(12), IN(13), FORM(13)
DIMENSION  A(32), S(32), Z(16), R(32), T(13,7), M(6)
LOGICAL    A, S, LEX, LKO, LBL, LST, ERR, POINT, EIN, AUS, CLEAR
INTEGER    Z,T,ANT(13,32),POS(20,2),SPVLAB,UPRCNT,LABSAV(10)

COMMON /LIDIA/ A,S,Z,R,T,M,ERR,NPROZ,LEX,LKO,LBL,LST,
1  ANT,NAN,NTXT,IP,POS,NUMLI,SPVLAB,UPRCNT,LABSAV,POINT,
2  EIN,AUS,CLEAR,NUMCHR,LCHR

COMMON /$GSCOS/ KOMMAN,INPUT,IANBYT,ITIME,IANOUT

COMMON /VERW/ FORM
```

In den Einzelbeschreibungen der Routinen sind die jeweils verwendeten Werte unterstrichen.



T e i l 1

K a p i t e l 5.1



HHI Berlin	L I F O R T Übersetzer		Katalog-Nr. DG78915
Name	L I F O R T	HP	Nr. 1
Stichwort	Hauptprogramm		
Verfasser	Herbert Gold	Version vom 15.9.1978	
Parameter			
Unterprogramm- aufrufe	s. Kap. 6.1		
Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangsparameter &=Marke			
Name	Art	Typ	Zweck
COMMON			<u>/IN/</u> <u>ICOMPL, IERRUN, IIMIN, LIN</u> <u>/LABEL/</u> <u>LABLX, LABOBT, LABELS, NSLVT</u> <u>/LIDVRS/</u> <u>ZMBERS, SEMERS, ANMBRS, TFMERS, RUMBR</u> <u>/EOUT/</u> <u>ECHELE, LSTEIN</u> <u>/OUT/</u> <u>OCHELE, COUTO3, CUTHUM, OUTLIN, NEOCRD</u> <u>/SAVDAT/</u> <u>SAVATR, POS, NUM, LINES</u> <u>/CONSTS/</u> <u>BILD, LIES, ANTW, TEST, WENN, NACH,</u> <u>TSFR, RTSP, LADE, UPRO, UEND, HALT,</u> <u>STOP, ENDE, PROG, SPACE, ASTRSK, AFGSTR,</u> <u>POINT, EQUAL, COLON, CROSS, COMMA, OIEN2,</u> <u>CLOSER, A, F, M, R, S, T, Z, DIGITS, START</u>
<u>Beschreibung:</u>			
Übersetzt in zwei Läufen LIDIA-Programme in FORTRAN-Programme.			



Bild 3: Struktogramm LIFORT

Name: LIFORT-Hauptprogramm

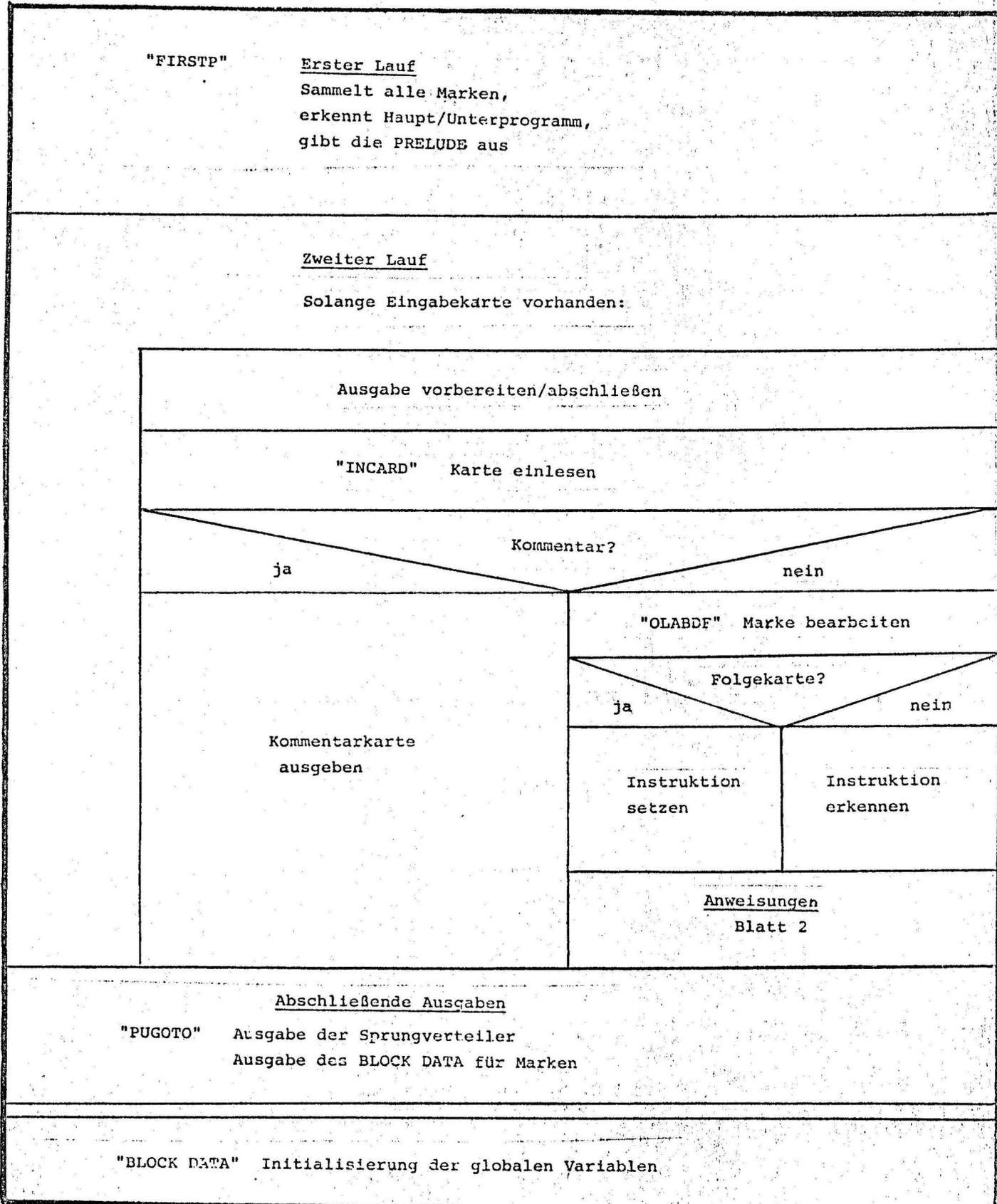


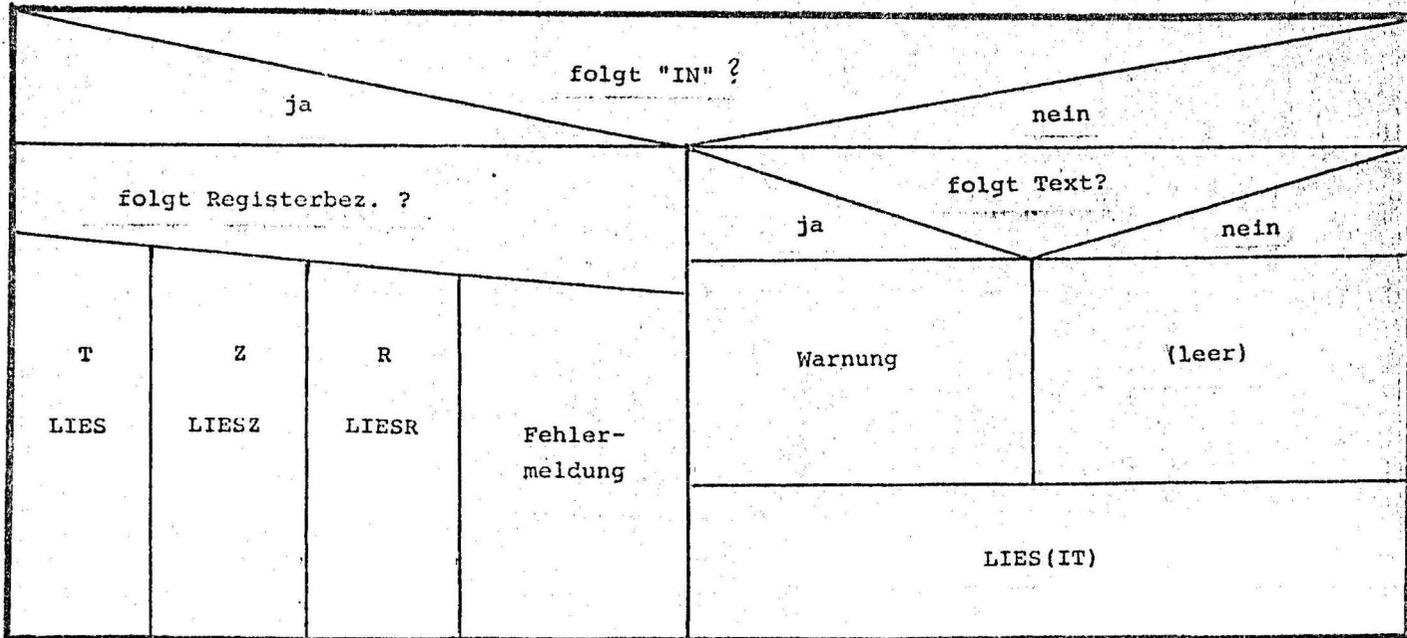
Bild 4: Struktogramm LIFORP
 Name: Anweisungen

Blatt 2

Verzweige nach Instruktion
 (Fehlermeldung, falls außer Bereich)

<u>BILD</u>	<u>LIES</u>	<u>ANTW</u>	<u>TEST</u>	<u>WENN</u>	<u>NACH</u>	<u>TSPR</u>	<u>RTSP</u>	<u>LADE</u>	<u>UPRO</u>	<u>ENDE</u>	<u>HALT</u>	<u>START</u>
									<u>UEND</u>		<u>STOP</u>	<u>PROG</u>
Marke 3100	3200	3300	3400	3500	3600	3700	3800	3900	4000 4100	4200	4200 4300	4400 4500
	Blatt 3	"PUTEXT"	Blatt 3	"SAVE" "SCNEXP"	"SCNLAB"		"SCNLAB"	Blatt 4				
							"PUSTR"					

Bild 5: Struktogramm LIFORT
Name: LIES (Übersetzer)



Name: TEST (Übersetzer)

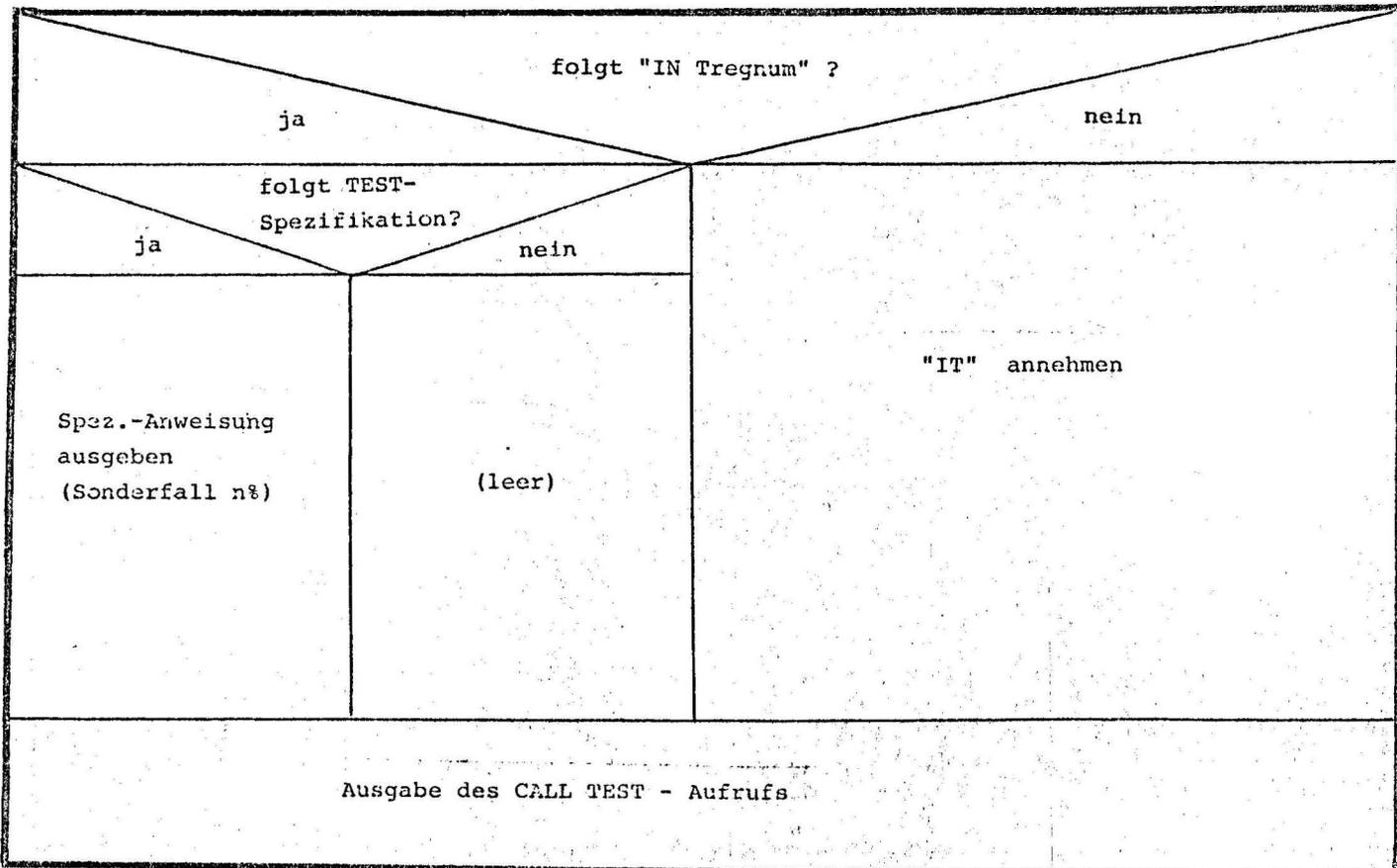
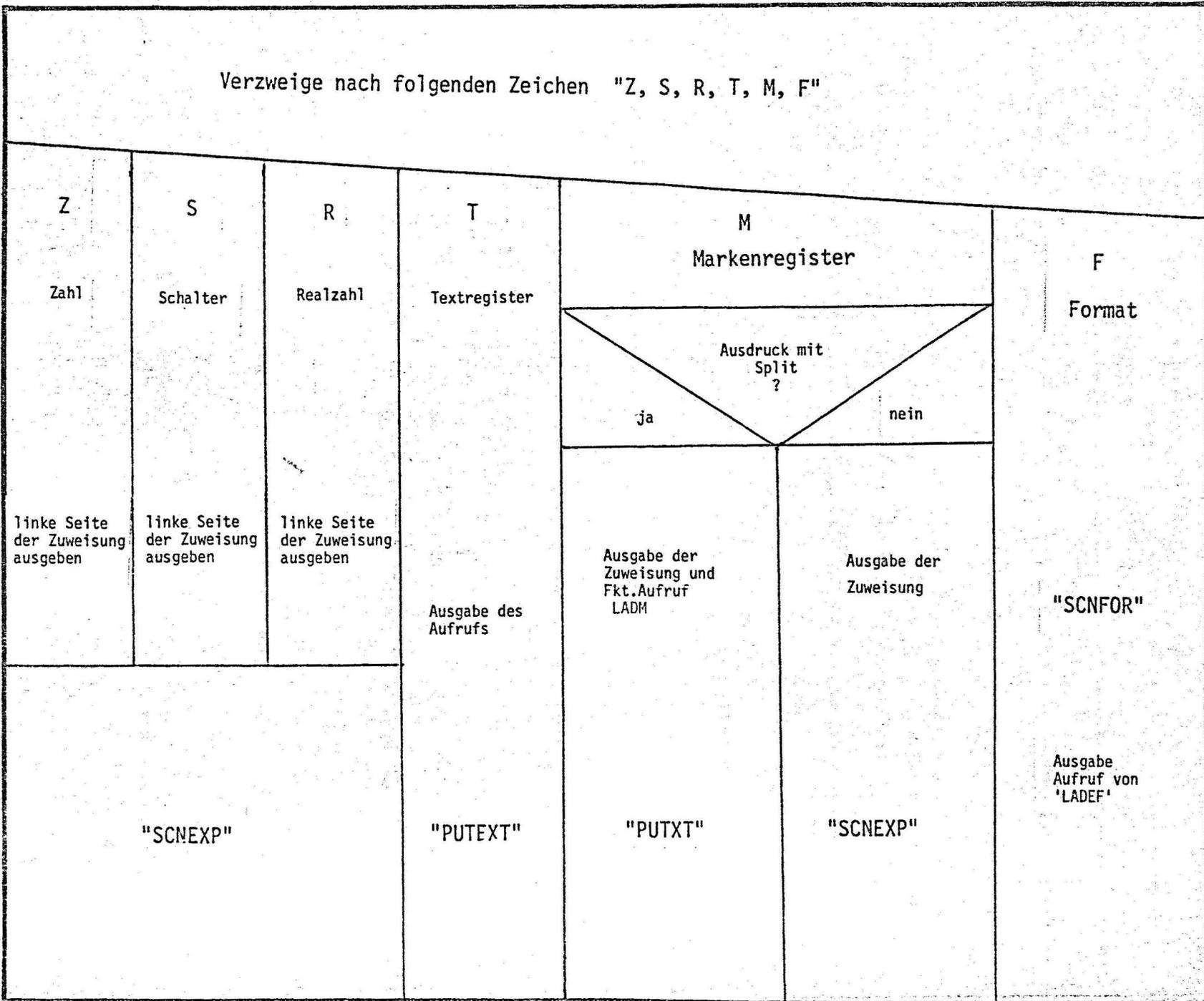


Bild 6: Struktogramm LIFORT

Name: LADE (Übersetzer)

Blatt 4



HHI Berlin	L I F O R T Übersetzer	Katalog-Nr. DG78915	
Name	ERRMSG	UP	Nr. 2
Stichwort	Errormessage, zentrale Fehlermeldung		
Verfasser	Herbert Gold	Version vom 15.9.1978	
Parameter	(string, length, pos)		
Unterprogramm- aufrufe	EXPATX		

Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangsparameter &=Marke

Name	Art	Typ	Zweck
string	E	I 4	enthält den Text der Fehlermeldung
length	E	I 4	die Länge von "string"
pos	E	I 4	die Position des Fehlers im Quelltext oder Steuerfunktion.

COMMON	<p>/IN/ ICHNL, LCHNL, ICHLN, LIN</p> <p>/LABEL/ LABELX, LABELT, LABELS, NSIPT</p> <p>/LDVRS/ ZMBERS, SMBERS, AMBERS, PMBERS, RMBERS</p> <p>/ROUT/ RCHNL, LSTEIN</p> <p>/OUT/ OCHNI, OUTPOS, OUTIRM, OUTLIN, NEOCRD</p> <p>/SAVDAT/ SAVI TR, POS, NUM, LINES</p> <p>/CONSTS/ BILD, LIES, ANTW, TEST, WENN, NACH, TSIR, RTSP, LADE, UIRO, UEND, HALT, STOP, ENDE, PROC, SPACE, ASTRSK, AFGSTR, POINT, EQUAL, COLOR, CROSS, COMMA, OLEN2, CLOCE2, A, F, M, R, S, T, Z, DIGITS, START</p>
--------	---

Beschreibung:

Wenn "pos" größer als null wird über "MCHNL" die Eingabenzeilennummer, Ausgabenzeilennummer und der expandierte "string" ausgegeben. Weiterhin wird eine Zeile, die das Zeichen "\$" an der Fehlerposition enthält und anschließend die fehlerhafte Eingabezeile ausgegeben. Auf alle Fälle (unabhängig von "pos") wird über den Ausgabekanal "OCHNL" eine Kommentarkarte "C++ERROR" gefolgt von der fehlerhaften Eingabezeile, ebenfalls als Kommentar, ausgegeben.

Laborprojekt Zweiweg-Kabelfernsehen



HHI Berlin	L I F O R T Übersetzer		Katalog-Nr. DG78915
Name	EVAL	UP	Nr. 3
Stichwort	übersetzt LADE-Aufrufe		
Verfasser	Herbert Gold	Version vom 15.9.1978	
Parameter	(char)		
Unterprogramm- aufrufe	FIND1C, PUCHAR, PUTXT, ERRMSG		
Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangsparameter &=Marke			
Name	Art	Typ	Zweck
char	E	I 4	Kennzeichen für LADE-Routine
COMMON	<u>/IN/</u> ICNCL, LCNCL, IBLIN, LIN <u>/LABEL/</u> LABELX, LABOPT, LABELS, NSPVT <u>/LIDVRS/</u> ZUMERS, SMDBRS, AMDBRS, TNMBRS, RHEBRS <u>/ROUT/</u> ECHNL, LSTELN <u>/OUT/</u> OCHNL, OUTPOS, OUTNAM, OUTLIN, NEOGRD <u>/SAVDAT/</u> SAVPTR, POS, NUM, LINES <u>/CONSTS/</u> BILD, LIES, ANTW, TEST, WENN, NACH, TSR, RTSP, LADE, UPRO, UEND, HALT, STOP, ERDE, PROG, SPACE, ASTRSK, AFSTR, POINT, EQUAL, COLOR, CROSS, COMMA, OIEN2, CLOSE2, A, F, M, R, S, T, Z, DIGITS, START		
<u>Beschreibung:</u>			
Erzeugt den Aufruf von LAD"char" mit der Nummer des Textregisters aus der Quellzeile.			

HHI Berlin	L I F O R T Übersetzer		Katalog-Nr. DG78915
Name	FIND1C	UP	Nr. 4
Stichwort	Finde 1. Character im String		
Verfasser	Herbert Gold	Version vom 15.9.1978	
Parameter	(&lab)		
Unterprogramm- aufrufe			
Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangsparameter &=Marke			
Name	Art	Typ	Zweck
&lab	A	&	Rücksprungmarke, falls nur Leerzeichen ab Zeigerposition INPOS
COMMON	<pre> /IN/ IC/NL, INNM, IN-IN, LIN /LABEL/ LAFIX, LABORT, LABELS, NSIAT /LIDVRS/ ZMBRS, SIMBRS, ANMBRS, TMBRS, RMBRS /ROUT/ MCHNL, LSTELN /OUT/ CCHNL, OUTPOS, OUTNAM, OUTLIN, NEOCRD /SAVDAT/ SAVTR, POS, NUM, LINES /CONSTS/ BIID, LIES, ANTW, TEST, WENN, NACH, TSPR, RTSP, LADE, UPRO, UEND, HALT, STOP, ENDE, PROG, SPACE, ASTRSK, AFOSTR, POINT, EQUAL, COLOR, CROSS, COMA, OLEN2, GLOE2, A, F, M, R, S, T, Z, DIGITS, START </pre>		
Beschreibung:			
<p>Setzt den Eingabezeiger auf das erste Zeichen ungleich Blank des Eingabepuffers (der Zeiger wird nicht bewegt, wenn er bereits auf einem Zeichen steht, das kein Leerzeichen ist).</p> <p>Enthält die Eingabezeile nur noch blanks, so wird auf die Rücksprungmarke verzweigt.</p>			

HHI Berlin		L I F O R T Übersetzer		Katalog-Nr. DG78915	
Name		FIRSTP		UP	Nr. 5
Stichwort		First Pass, erster Lauf des Übersetzers			
Verfasser		Herbert Gold		Version vom 15.9.1978	
Parameter		(&err, ISHP, BLOCK, LABGEN)			
Unterprogramm- aufrufe		INCARD, COMPTX, LABINX, ERRMSG, PUTXT, PUCARD, PULAB			
Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangsparameter &=Marke					
Name	Art	Typ	Zweck		
&err	A	&	Fehlerausgang bei LIDIA-Programmen mit mehr Marken als Feldgröße LABELS erlaubt (=1024)		
ISHP	A	L4	.TRUE. bei LIDIA-Hauptprogramm		
BLOCK	A	L4	.FALSE. Initialisiert Indikator für Ausgabe v. BLOCK DATA		
LABGEN	A	I4	Anfangswert zur Labelgenerierung		
COMMON		<u>/IN/</u> ICORNL, INKRN, IIN, IN, LIN <u>/LABEL/</u> LABELX, LABELT, LABELS, USEPVT <u>/LIDVRS/</u> ZUMBERS, SNUMBERS, ANMERS, TNMERS, RNUMBERS <u>/ROUT/</u> ICORNL, LSTEIN <u>/OUT/</u> OCORNL, OUTPOS, OUTRUM, OUTLIN, NEOCRD <u>/SAVDAT/</u> SAVETR, POS, NUM, LINES <u>/CONSTS/</u> BILD, LIES, ANTW, TEST, WENN, NACH, TSER, RTSP, LADE, UPRO, UEND, HALT, STOP, ERDE, PRG, SPACE, ASTRSK, APCSTR, POINT, EQUAL, COLON, CROSS, COLMA, OFEN2, CLOCE2, A, P, M, R, S, T, Z, DIGITS, START			
<u>Beschreibung:</u>					
Führt den ersten Lauf für die Umsetzung durch. Es werden alle Marken in den Spalten 1-8 erkannt und geprüft, ob es sich um ein Hauptprogramm oder Unterprogramm handelt. Die LIFORTPRELUDE wird eingelesen (Kanal 50) und nach dem Programmkopf ausgegeben.					

HHI Berlin		L I F O R T Übersetzer		Katalog-Nr. DG78915	
Name		INCARD		UP	Nr. 6
Stichwort		Karte einlesen			
Verfasser		Herbert Gold		Version vom 15.9.1978	
Parameter		(savrd, &lab)			
Unterprogramm- aufrufe		EXPATX			
Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangsparameter &=Marke					
Name	Art	Typ	Zweck		
savrd	E	I4	log. Steuerparameter, der angibt, ob aus dem "Rettungskeller" oder direkt von der Eingabe zu lesen ist.		
&lab	A	&	Rücksprungmarke, zu der zu verzweigen ist, wenn das Ende der Eingabe erreicht wurde.		
COMMON	<p> <u>/IN/</u> ICHNL, IRRUN, IRLIN, LIN <u>/LABEL/</u> LABELX, LABELT, LABELS, NSPVT <u>/LDVRS/</u> ZIMERS, SIMBERS, ANMBRS, TNMBRS, RNEBRS <u>/ROUT/</u> ICHNL, LSTELN <u>/OUT/</u> CCHNL, CUPPOS, CUTIUM, OUTLIN, NEOCRD <u>/SAVDAT/</u> SAVTR, POS, NUM, LINES <u>/CONSTS/</u> BIID, LIES, ANTW, TEST, WENN, NACH, TSFR, RTSP, LADE, UPRO, UEND, HALT, STOP, ERDE, PRG, SPACE, ASTRSK, AFGSTR, POINT, EQUAL, COLOR, CROSS, COMMA, OFEN2, CLOSE2, A, F, M, R, S, T, Z, DIGITS, START </p>				

Beschreibung:

Ist "savrd" wahr, so wird, falls der Rettungskeller nicht leer ist, die Eingabezeile aus dem Rettungskeller wiederhergestellt. Anderenfalls wird die Eingabezeile über den Kanal "ICHNL" eingelesen, die Eingabeposition auf 16 gesetzt und der Eingabetext expandiert. Falls das Ende der Eingabefile erreicht wurde, wird zur übergebenen Marke verzweigt.



HHI Berlin	L I F O R T Übersetzer	Katalog-Nr. DG78915	
Name	LABINX	UP	Nr. 7
Stichwort	Label-Index suchen		
Verfasser	Herbert Gold	Version vom 15.9.1978	
Parameter	(label, index)		
Unterprogramm- aufrufe			

Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangsparameter &=Marke

Name	Art	Typ	Zweck
label	E	Ix4	textuelle Darstellung der LIDIA-Marke
index	A	Ix4	Index im Markenfeld

COMMON	/IN/ ICONE, IRRCH, IRI IN, IIN /LABEL/ LABELX, LAPOUT, LABELS, NSPVT /LIDVRS/ ZIMERS, SEMERS, ANMERS, TMMERS, RMMERS /MOUT/ MOINL, LSTELN /OUT/ OCINI, CUTPOS, CUTHUM, OUTLIN, NEOORD /SAVDAT/ SAVLTR, POS, NUM, LINES /CONSTS/ BILD, LIES, ANTW, TEST, WENN, NACH, TSPR, RTSP, LADE, UPRO, UEND, HALT, STOP, ENDE, PROG, SPACE, ASTRSK, AFCSTR, POINT, EQUAL, COLOR, CROSS, COMMA, OIEN2, CLOSE2, A, F, M, R, S, T, Z, DIGITS, START
--------	---

Beschreibung:

Sucht die Marke im Markenfeld und gibt ihren Index zurück. Wird die Marke nicht gefunden, so ist "index" = 0.



HHI Berlin		L I F O R T Übersetzer		Katalog-Nr. DG78915	
Name		OLABDF		UP	Nr. 8
Stichwort		*			
Verfasser		Herbert Gold		Version vom 15.9.1978	
Parameter					
Unterprogramm- aufrufe		LABINX, PULAB, COMPTX			
Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangsparameter &=Marke					
Name	Art	Typ	Zweck		
			* output label definition, Umsetzung einer Marke		
COMMON			<u>/IN/</u> ICNHL, INRUM, INLIN, LIN <u>/LABEL/</u> LABINX, LABONT, LABELS, NSLVT <u>/LIDVRS/</u> ZUMERS, SEMBRS, ANMBRS, TMBRS, RMBRS <u>/EOUT/</u> ECHNL, LSTEIN <u>/OUT/</u> OCHNL, CURPOS, CUTHUM, OUTLIN, NEOCRD <u>/SAVDAT/</u> SAVPTR, POS, NUM, LINES <u>/CONSTS/</u> BILD, LIES, ANTW, TEST, WENN, NACH, TSR, RTSP, LADE, UPRO, UEND, HALT, STOP, ENDE, PRG, <u>SPACE</u> , ASTRSK, AFSTR, POINT, EQUAL, COLON, CROSS, COMMA, OPEN2, CLOSE2, A, F, M, R, S, T, Z, DIGITS, START		

Beschreibung:

Komprimiert die im Eingabepuffer stehende Marke, ruft LABINX, um ihren Index in der Markentabelle zu erhalten und gibt die entsprechende FORTRAN-Marke aus.

HHI Berlin	L I F O R T Übersetzer	Katalog-Nr. DG78915	
Name	PUCALL	UP	Nr. 9
Stichwort	punch call		
Verfasser	Herbert Gold	Version vom 15.9.1978	
Parameter	(string, length)		
Unterprogramm- aufrufe	PUTXT		

Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangsparameter &=Marke

Name	Art	Typ	Zweck
string	E	Ix4	Name des Unterprogramms
length	E	Ix4	Länge des Strings

COMMON	<pre> /IN/ LOBNL, INNUM, INLIN, LIN /LABEL/ J AEMAX, LABOFT, LABELS, NSIVT /LIDVRS/ ZIMERS, SIMERS, ANMERS, TIMERS, RNUMERS /ROUT/ MOHNE, LSTEIN /OUT/ COHNI, CUPPOS, CUTIAM, OUTLIN, NEOCRD /SAVDAT/ SAVTR, POS, NUM, LINES /CONSTS/ BIID, LIES, ANTW, TEST, WENN, NACH, TSFR, RTSP, LADE, UPRO, UEND, HALT, STOP, ENDE, PRGG, SPACE, ASTRSK, AFCSTR, POINT, EQUAL, COLON, CROSS, COMMA, OFEN2, CLOSE2, A, F, M, R, S, T, Z, DIGITS, START </pre>
--------	--

Beschreibung:

PUCALL dient zur Ausgabe eines Unterprogrammaufrufs.

HHI Berlin		L I F O R T Übersetzer		Katalog-Nr. DG78915	
Name		PUCARD		UP	Nr. 10
Stichwort		punch card, Ausgabe einer Karte			
Verfasser		Herbert Gold		Version vom 15.9.1978	
Parameter		CALL PUCARD			
Unterprogramm- aufrufe					
Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangsparameter &=Marke					
Name	Art	Typ	Zweck		
COMMON			<u>/IN/</u> ICHNL, ICHNL1, ICHLN, LIN <u>/LABEL/</u> J AEMAX, LABORT, LABELS, NSIVT <u>/LIDVRS/</u> ZIMERS, SIMERS, ANMERS, TMMERS, RNMERS <u>/EOUT/</u> ECHNL, LSTELN <u>/OUT/</u> OCHNL, OUTPOS, CUTIRM, OUTLIN, REGRD <u>/SAVDAT/</u> SAV, TR, POS, NUM, LINES <u>/CONSTS/</u> BILD, LIES, ANTW, TEST, WENN, NACH, TSIR, RTSP, LADE, UPRO, UEND, HALT, STOP, ENDE, PROG, SPACE, ASTRSK, AFCSTR, POINT, EQUAL, COLON, CROSS, COMMA, OPEN2, CLOSE2, A, F, M, R, S, T, Z, DIGITS, START		
<u>Beschreibung:</u>					
Komprimiert den Ausgabezeichenpuffer und gibt ihn anschließend auf dem Ausgabe- kanal "OCHNL" aus. Der Ausgabepuffer wird mit Blanks initialisiert und die aktuelle Position auf 7 gesetzt.					

Laborprojekt Zweiweg-Kabelfernsehen



HHI Berlin	L I F O R T Übersetzer		Katalog-Nr. DG78915
Name	PUCCRD	UP	Nr. 11
Stichwort	punch continuation card, Ausgabe einer Karte mit Fortsetzung		
Verfasser	Herbert Gold	Version vom 15.9.1978	
Parameter			
Unterprogramm- aufrufe	PUCARD		
Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangsparameter &=Marke			
Name	Art	Typ	Zweck
COMMON			/IN/ ICONE, LNRBR, IRLIN, LIN /LABEL/ I NERLX, LAOBT, LABELS, NSVVT /LIDVRS/ ZIMERS, SEMERS, ANMERS, TNEBRS, RHMERS /ROUT/ MOENL, LSTELN /OUT/ OCHNL, CUMPOS, CUTIUM, <u>OUTLIN</u> , <u>NEOGRD</u> /SAVDAT/ SAVPTR, POS, NUM, LINES /CORSTS/ BILD, LIES, ANTW, TEST, WENN, NACH, TSFR, RTSP, LADE, UPRO, UEND, HALT, STOP, ERDE, PRG, SPACE, <u>ASTRSK</u> , <u>AFGSTR</u> , POINT, EQU/L, COLON, CROSS, COMMA, OFEN2, GLOCE2, A, P, M, R, S, T, Z, DIGITS, START
<u>Beschreibung:</u>			
Ruft PUCARD und trägt anschließend einen Stern in Position 6 der neuen Karte als Fortsetzungskarte ein.			



HHI Berlin	L I F O R T Übersetzer		Katalog-Nr. DG78915
Name	PUCHAR	UP	Nr. 12
Stichwort	punch character, Ausgabe eines Zeichens		
Verfasser	Herbert Gold	Version vom 15.9.1978	
Parameter	(char)		
Unterprogramm- aufrufe	PUCARD		
Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangsparameter &=Marke			
Name	Art	Typ	Zweck
char	E	Ix4	auszugebendes Zeichen
COMMON	<p> <u>/IN/</u> ICENE, INERT, III IN, LIN <u>/LABEL/</u> LABELX, LABORT, LABELS, NSPVT <u>/LIDVRS/</u> ZIMERS, SIMERS, AMERS, TNEBRS, RNEBRS <u>/ROUT/</u> ROINE, LSTEIN <u>/OUT/</u> OONE, OUTPOS, OUTIUM, <u>OUTLIN</u>, NEOCRD <u>/SAVDAT/</u> SAVTR, POS, NUM, LINES <u>/CONSTS/</u> BILL, LLES, ANTW, TEST, WENN, NACH, TSER, RTSP, LADE, UPRO, UEND, HALT, STOP, ERDE, PROG, SPACE, <u>ASTRSK</u>, ATOSTR, POINT, EQUAL, COLOR, CROSS, COLMA, OIEN2, GLOCE2, A, F, N, R, S, T, Z, DIGITS, START </p>		
<u>Beschreibung:</u>			
<p>Ist die aktuelle Ausgabeposition größer 72, so wird PUCARD gerufen und ein Stern in Position 6 der neuen Karte eingetragen, und die aktuelle Ausgabeposition auf 7 gesetzt. Dann wird "char" in den Zeichenpuffer an der aktuellen Position eingetragen und die Position um eins erhöht.</p>			

Laborprojekt Zweiweg-Kabelfernsehen



HHI Berlin		L I F O R T Übersetzer		Katalog-Nr. DG78915	
Name		PUGOTO		UP	Nr. 13
Stichwort		punch goto, Ausgabe des Sprungverteilers			
Verfasser		Herbert Gold		Version vom 15.9.1978	
Parameter		(LABGEN)			
Unterprogramm- aufrufe		PUCARD, PULAB, PUTXT, PUCHAR, PUCCRD, PUINT			
Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangsparameter &=Marke					
Name	Art	Typ	Zweck		
labgen	D	Ix4	größtes Label		
COMMON	<pre> /IN/ ICNCL, IERRH, IHI IN, LIN /LABEL/ IABEL, X, LABGNT, LABELS, NSLVT /LIDVRS/ ZMBRS, SMBRS, AMBRS, TMBRS, RMBRS /ECUT/ ECHNL, LSTELN /OUT/ OCHNL, OUTPOS, CUTIRUM, OUTLIN, NEOCRD /SAVDAT/ SAVPTR, POS, NUM, LINES /CONSTS/ BILD, LIES, ANTW, TEST, WENN, NACH, TSER, RTSP, LADE, UPRO, UEND, HALT, STOP, ENDE, PRG, SPACE, ASTRSK, AFOSTR, POINT, EQUAL, COLOR, CROSS, COMMA, OPEN2, CLOCE2, A, F, M, R, S, T, Z, DIGITS, START </pre>				
<u>Beschreibung:</u>					
Generiert den Sprungverteiler von 1000 bis LABGEN in Schritten von 10. Jeweils 100 Marken für ein computed GOTO. Der Sprungverteiler wird gleichzeitig ausgegeben.					



HHI Berlin		L I F O R T Übersetzer		Katalog-Nr. DG78915	
Name		PUI NT		UP	Nr. 14
Stichwort		punch integer, Ausgabe einer ganzen Zahl			
Verfasser		Herbert Gold		Version vom 15.9.1978	
Parameter		(number, length)			
Unterprogramm- aufrufe		ERRMSG, P UCHAR			
Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangparameter &=Marke					
Name	Art	Typ	Zweck		
number	E	Ix4	die auszugebende Zahl		
length	E	Ix4	der Absolutwert der Feldlänge, in die number ausgegeben wird, Steuerung durch das Vorzeichen.		
COMMON		/IN/ IOENL, IINRU, IININ, LIN /LABEL/ LABMAX, LABONT, LABELS, NSPVT /LIDVRS/ ZUMERS, SEMERS, ANNERS, TMMERS, RMMERS /EOUT/ ECHNL, LSTEIN /OUT/ OCHNL, OUTPOS, OUTIUM, OUTLIN, NEOCRD /SAVDAT/ SAVPTR, POS, NUM, LINES /CONSTS/ BILD, LIES, ANTW, TEST, WENN, NACH, TSPR, RTSP, LADE, UPRO, UEND, HALT, STOP, ERDE, PRG, SPACE, ASTRSK, AFCSTR, POINT, EQUAL, COLOR, CROSS, COLMA, OFEN2, CLOSE2, A, F, M, R, S, T, Z, DIGITS, START			
<u>Beschreibung:</u>					
<p>Die Zahl "number" wird ab der aktuellen Ausgabeposition in ein Feld der Länge "length" ausgegeben. Negatives Vorzeichen von "length" gibt an, daß abs (length) als maximale Feldlänge anzusehen ist, d.h. es werden keine zusätzlichen Blanks mit ausgegeben.</p>					

Laborprojekt Zweifweg-Kabelfernsehen



HHI Berlin	L I F O R T Übersetzer	Katalog-Nr. DG78915	
Name	PULAB	UP	Nr. 15
Stichwort	punch label, Ausgabe einer Marke		
Verfasser	Herbert Gold	Version vom 15.9.1978	
Parameter	(label, pos)		
Unterprogramm- aufrufe	PUCARD, ERRMSG, PUCAR		

Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangsparameter &=Marke

Name	Art	Typ	Zweck
label	E	Ix4	Enthält die auszugebende FORTRAN-Marke als Zahl
pos	E	Ix4	Enthält die Ausgabeposition oder 0.

COMMON	<pre> /IN/ ICNCL, INRCH, IMLIN, LIN /LABEL/ LAEMAX, LABOPT, LABELS, NSLVT /LIDVRS/ ZEMERS, SEMERS, ANMERS, TFMERS, RHEBRS /ROUT/ MCHNL, LSTBLN /OUT/ CCNCL, CUMPOS, CUMTRM, OUTLIN, REOGRD /SAVDAT/ SAVPTR, POS, NUM, LINES /CONSTS/ BILD, LIES, ANTW, TEST, WENN, NACH, TSPR, RTSP, LADE, UPRO, UEND, HALT, STOP, ENDE, PROG, SPACE, ASTRSK, AFOSTR, POINT, EQUAL, COLOR, CROSS, COMHA, OFEN2, CLOSE2, A, F, M, R, S, T, Z, DIGITS, START </pre>
--------	---

Beschreibung:

Ab der angegebenen Position "pos" ist die Marke auszugeben. Ist die Position gleich null, so wird die aktuelle Ausgabeposition +1 genommen.

HHI Berlin	L I F O R T Übersetzer	Katalog-Nr. DG78915	
Name	PUSTR	UP	Nr. 16
Stichwort	punch string, Ausgabe einer Zeichenkette		
Verfasser	Herbert Gold	Version vom 15.9.1978	
Parameter	(pos)		
Unterprogramm- aufrufe	PUCARD, PUTXT, PUCAR, PUINT, PUCRD		

Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangparameter &=Marke

Name	Art	Typ	Zweck
POS	E	Ix4	Anfangsposition für Textausgabe und Steuerparameter durch das Vorzeichen
COMMON			<p><u>/IN/</u> ICNCL, INRUL, IRLIN, LIN</p> <p><u>/LABEL/</u> LABELX, LABELT, LABELS, NSPVT</p> <p><u>/LIDVRS/</u> ZUMERS, SEMERS, ANMERS, TNEBRS, RNMBS</p> <p><u>/MOUT/</u> MCNCL, LSTEIN</p> <p><u>/OUT/</u> CCNCL, CUPPOS, CUTUM, OUTLIN, NEOCRD</p> <p><u>/SAVDAT/</u> SAVPTR, POS, NUM, LINES</p> <p><u>/CONSTS/</u> BILD, LIES, ANW, TEST, WENN, NACH, TSR, RTSP, LADE, UPRO, UEND, HALT, STOP, ENDE, PRGG, SPACE, ASTRSK, AFOSTR, POINT, EQUAL, COLOR, <u>CROSS</u>, COMMA, OPEN2, <u>CLOSE2</u>, A, F, N, R, S, T, Z, <u>DIGITS</u>, START</p>

Beschreibung:

Dieses UP entspricht in seiner Funktion PUTEUT, jedoch prüft es auf # am Ende des Strings und setzt einen weiteren Parameter ab (nach dem Text), der zu 0 initialisiert ist und den Wert der Zahl nach dem # Zeichen besitzt, falls eine Zahl folgt.

Dies ist die Ergänzung für die Erkennung einer Kommandotaste.



HHI Berlin	L I F O R T Übersetzer	Katalog-Nr. DG78915	
Name	PUTEXT	UP	Nr. 17
Stichwort	püncn text, Ausgabe einer Zeichenkette		
Verfasser	Herbert Gold	Version vom 15.9.1978	
Parameter	(pos)		
Unterprogramm- aufrufe	PUCARD, ERRMSG, PUTXT, PUCAR, PUINT, PUCCRD		

Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangsparameter &=Marke

Name	Art	Typ	Zweck
pos	E	Ix4	Anfangsposition für Textausgabe und Steuerparameter durch das Vorzeichen
COMMON			<p><u>/IN/</u> IOENL, INRUN, IRLIN, LIN</p> <p><u>/LABEL/</u> LABELX, LABELT, LABELS, NSPVT</p> <p><u>/LIDVRS/</u> ZUMERS, SMBERS, AMBERS, TMBERS, RMBERS</p> <p><u>/EOUT/</u> ECHNL, LSTEIN</p> <p><u>/OUT/</u> OCHNL, OUTPOS, OUTNUM, OUTLIN, NEOCRD</p> <p><u>/SAVDAT/</u> SAVFTR, POS, NUM, LINES</p> <p><u>/CONSTS/</u> BILD, LIES, ANTW, TEST, WENN, NACH, TSFR, RTSP, LADE, UPRO, UEND, HALT, STOP, ENDE, PRG, <u>SPACE</u>, <u>ASTRSK</u>, <u>APOSTR</u>, POINT, EQUAL, COLON, <u>CROSS</u>, <u>COMMA</u>, <u>OLEN2</u>, <u>CLOSE2</u>, A, F, M, R, S, T, Z, <u>DIGITS</u>, START</p>

Beschreibung:

Die LIDIA-Zeile wird ab abs (pos) untersucht und in der Form "l,lHt" ausgegeben. Zunächst wird die Länge des Textes "l" bestimmt, wobei die Position des ersten Apostroph gemerkt wird. Anschließend wird, wenn "pos" negativ war, die Position des ersten Apostroph ausgegeben. Danach werden die Länge des Textes und der Text selbst, gefolgt von einer schließenden Klammer, ausgegeben. Enthielt die Zeile am Schluß die Zeichenfolge "./.", so wird auf einer weiteren Karte (PUCARD) der Text "ip=o" ausgegeben.

(COMMON wie PWSTR)

HHI Berlin	L I F O R T Übersetzer		Katalog-Nr. DG78915
Name	PUTXT	UP	Nr. 18
Stichwort	punch text-string, Text-Ausgabe		
Verfasser	Herbert Gold	Version vom 15.9.1978	
Parameter	(string, length)		
Unterprogramm- aufrufe	EXPATX, PUCCHAR		
Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangsparameter &=Marke			
Name	Art	Typ	Zweck
string	E	Ix4	der auszugebende Text
length	E	Ix4	dessen Länge
COMMON	<pre> /IN/ IOINL, IORUN, IOLIN, LIN /LABEL/ LABELX, LABCNT, LABELS, NSIAT /LIDVRS/ ZIMERS, SIMERS, ANMERS, TMERS, RNMERS /ECUT/ EOHNL, LSTEIN /OUT/ OOHNI, OUPPOS, OUTHUM, OUTLIN, NEOCRD /SAVDAT/ SAVPTR, POS, NUM, LINES /CONETS/ BILD, LIES, ANTW, TEST, WENN, NACH, TSFR, RTSP, LADE, UPRO, UEND, HALT, STOP, ENDE, PROC, SPACE, ASTRSK, AFGSTR, POINT, EQUAL, COLON, CROSS, COLMA, OFEN2, GLOSE2, A, F, M, R, S, T, Z, DIGITS, START </pre>		

Beschreibung:

PUTXT gibt den "string" der Länge "length" aus. Der Text wird dazu expandiert und die einzelnen Zeichen hintereinander mit PUCCHAR ausgegeben.



HHI Berlin	L I F O R T Übersetzer		Katalog-Nr. DG78915
Name	SAVE	UP	Nr. 19
Stichwort	Kellern der Eingabezeile		
Verfasser	Herbert Gold	Version vom 15,9.1978	
Parameter			
Unterprogramm- aufrufe	ERRMSG, COMPTX		
Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangparameter &=Marke			
Name	Art	Typ	Zweck
COMMON			/IN/ IOINL, INCHI, ININ, LIN /LABEL/ LABELX, LABELT, LABELS, INSLVT /LIDVRS/ ZUMERS, SUMERS, ANGBRS, TNMBRS, RMMBRS /ROUT/ MOHNL, LSTELR /OUT/ COHNL, CUMOG, CUMRM, OUTLIN, NEOCRD /SAVDAT/ SAVPTR, POS, NUH, LINES /CONSTS/ BILD, LIES, ANTW, TEST, WENN, NACH, TSFR, RTSP, LADE, UPRO, UEND, HALT, STOP, ENDE, PROG, SPACE, ASTRSK, AFSTR, POINT, EQVAL, COLOR, CROSS, COLMA, OFEN2, CLOSE2, A, P, N, R, S, T, Z, DIGITS, START
Beschreibung: Komprimiert die aktuelle Eingabezeile und kopiert sie, die aktuelle Eingabe- position und die Nummer der Eingabekarte in einen Rettungskeller. Falls der Keller überläuft, wird eine Fehlermeldung abgegeben.			



HHI Berlin	L I F O R T Übersetzer	Katalog-Nr. DG78915	
Name	SCNFOR	UP	Nr. 20
Stichwort	scan format, Umsetzung der Format-Anweisung		
Verfasser	Herbert Gold	Version vom 15.9.1978	
Parameter	(IPAR)		
Unterprogramm- aufrufe	FIND1C, SCANT		

Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangparameter &=Marke

Name	Art	Typ	Zweck
IPAR	A	I4(7)	Sieben Parameterwerte in einem Feld beschreiben das gefundene Format.
COMMON			<p><u>/IN/</u> ICHNL, INRUL, INLR, LIN</p> <p><u>/LABEL/</u> LABELX, LABELT, LABELS, NSPVT</p> <p><u>/LIDVRS/</u> ZIMERS, SIMERS, AMERS, TNMERS, RNEBRS</p> <p><u>/RCUT/</u> MOENL, ISTEELN</p> <p><u>/CUT/</u> CCHLE, CUTPOS, CUTIRM, OUTLIN, REORD</p> <p><u>/SAVDAT/</u> SAVTR, POS, NUM, LINES</p> <p><u>/CONSTS/</u> BILD, LIES, ANTW, TEST, WENN, NACH, TSR, RTSP, LADE, UPRO, UEND, HALT, STOP, ENDE, PROG, SPACE, ASTRSK, AFOSTR, POINT, EQUAL, COLOR, CROSS, COMMA, OPER2, CLOSE2, A, F, N, R, S, T, Z, <u>DIGITS</u>, START</p>

Beschreibung:

Setzt eine LIDIA-FORMAT-Anweisung in sieben Parameter um: ns, nz, un, kn, kz, ks, ne

Bei Exponentendarstellung ist $ne \geq 1$ (Länge der Mantisse) s, z, n die Anzahl von Sternen, Zeichen -Z und Zeichen 9 n - vor, k - nach dem Komma.

Es wird auch die "Standard"-Einstellung erkannt.



HHI Berlin	L I F O R T Übersetzer		Katalog-Nr. DG78915
Name	SCNEXP	UP	Nr. 21
Stichwort	scan expression, Übersetzung von Ausdrücken		
Verfasser	Herbert Gold	Version vom 15.9.1978	
Parameter	(char)		
Unterprogramm- aufrufe	FINDIC, ERRMSG, EVAL, SCANT, INCARD, PUTXT, PCHAR		
Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangsparameter &=Marke			
Name	Art	Typ	Zweck
char	E	Ix4	steuert den Aufruf von EVAL für den Typ des jeweiligen Ausdrucks, z.B. R, M, Z etc.
COMMON	<p> <u>/IN/</u> ICHNE, IANUM, IHLIN, LIN <u>/LABEL/</u> LABELX, LABELT, LABELS, NSLVT <u>/LIDVRS/</u> ZIMERS, SIMERS, ANMERS, TMMERS, RMMERS <u>/ROUT/</u> MOHNE, LSTEEN <u>/OUT/</u> COHNI, CUTI03, CUTIUM, OUTLIN, NEOCRD <u>/SAVDAT/</u> SAVPTR, POS, NUM, LINES <u>/CONSTS/</u> BILD, LIES, ANTW, TEST, WENN, NACH, TSER, RTSP, LADE, UPRO, UEND, HALT, STOP, ENDE, PROG, SPACE, ASTRSK, AFOSTR, POINT, EQUAL, COLON, CROSS, COMMA, OFEN2, CLOSE2, A, F, M, R, S, T, Z, DIGITS, START </p>		
<u>Beschreibung:</u>			
<p>SCNEXP setzt arithmetische und log, Ausdrücke um, wobei die LIDIA-Operatoren zu FORTRAN-Operatoren bzw. Funktionsaufrufen übersetzt werden. Enthält die LIDIA-Eingabe eine Folgekettenkennzeichnung, so wird die nächste Karte eingelesen und wie oben angegeben übersetzt.</p>			



HHI Berlin	L I F O R T Übersetzer		Katalog-Nr. DG78915
Name	SCNLAB	UP	Nr. 22
Stichwort	scan label, erkenne Marke		
Verfasser	Herbert Gold	Version vom 15.9.1978	
Parameter	(label, index)		
Unterprogramm- aufrufe	SCANR, SCANT, LABINX, ERRMSG, COMPTX		
Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangsparameter &=Marke			
Name	Art	Typ	Zweck
label	A	I4	enthält die textuelle Darstellung einer gefundenen Marke
index	A	I 4	den Index in der Markentabelle oder eine Kennung
COMMON	/IN/ ICNLI, INRUH, IRLIN, LIN /LABEL/ LABELX, LABORT, LABELS, NSPVT /LIDVRS/ ZUMERS, SIMERS, ANBERS, TNEERS, RNEERS /MOUT/ MOHNL, LSTEIN /OUT/ OCHNL, CUMTOS, CUTIUM, OUTLIN, NEOCRD /SAVDAT/ SAVATR, POS, NUM, LINES /CONSTS/ BILD, LIES, ANTW, TEST, WENN, NACH, TSPR, RTSP, LADE, UPRO, UEND, HALT, STOP, ENDE, PROG, SPACE, ASTRSK, AFCSTR, POINT, EQUAL, COLOR, CROSS, COMMA, OPEN2, CLOSE2, A, F, N, R, S, T, Z, DIGITS, START		

Beschreibung: SCNLAB untersucht den Eingabetext ab dem aktuellen Eingabezeiger auf das Vorhandensein einer Marke oder eines Markenregisters. Da hiermit die Applikationen gefunden werden, wird im Fehlerfall eine entsprechende Meldung abgesetzt.

Wird ein Markenregister gefunden, so enthält "index" die Nummer als negative Zahl. Wird eine Marke gefunden, so erhält "label" ihre textuelle Darstellung und "index" ihren Index in der Markentabelle. Enthält die Eingabe eine Markenregisterbezeichnung (in Apostrophe eingeschlossen), so wird als "index" 0 und in "label" die Markenregisternummer geliefert.

Keine Marke ergibt die Ausgabewerte 0, 0; wird die Marke dagegen nicht in der Markentabelle gefunden, so erhält "label" den Text, "index" den Wert -999.

HHI Berlin	L I F O R T Übersetzer	Katalog-Nr. DG78915	
Name	SCANT logische Funktion	UP	Nr. 23
Stichwort	vergleicht Zeichenkette		
Verfasser	Herbert Gold	Version vom 15.9.1978	
Parameter	(String, Länge)		
Unterprogramm- aufrufe	FIND1C, EXPATX		

Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangsparameter &=Marke

Name	Art	Typ	Zweck
String	E	I 4	enthält den Vergleichsstring
Länge	E	I 4	ist dessen Länge

COMMON

```

/IN/      ICHNL, ICHRT, ICHTH, LIN
/LABEL/   ICHLX, ICHRT, LABELS, NSIVT
/LIDVRS/  ZUMERS, SUMERS, ANMERS, FNMBRS, RUMBERS
/ICUT/    ICHNL, LSTELI
/OUT/     OCHNL, OUTPOS, OUTNUM, OUTLIN, NEOCRD
/SAVDAT/  SAVETR, POS, NUM, LINES

/CONSTS/  BILD, LIES, ANTW, TEST, WENN, NACH,
          TSFR, RTSP, LADE, UPRO, UEND, HALT,
          STOP, ENDE, PRG, SPACE, ASTRSK, AFOSTR,
          POINT, EQUAL, COLOR, CROSS, COMMA, OPEN2,
          CLOSE2, A, P, M, R, S, T, Z, DIGITS, START

```

Beschreibung:

SCANT bewegt den Eingabezeiger (INPOS in /IN/) auf das nächste Zeichen ungleich Blank durch Aufruf von FIND1C. Anschließend wird die Zeichenkette auf Gleichheit mit "String" untersucht.

Bei Ungleichheit liefert SCANT "false", sonst "true". Der Eingabezeiger wird hinter das letzte Zeichen der gefundenen Kette gesetzt.

HHI Berlin	L I F O R T Übersetzer	Katalog-Nr. DG78915
Name	SCANR logische Funktion	UP Nr. 24
Stichwort	scan number, vergleicht Buchstabe und Zahl	
Verfasser	Herbert Gold	Version vom 15.9.1978
Parameter	(char, maxnmb, nmb)	
Unterprogramm- aufrufe	FINDIC	

Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangsparameter &=Marke

Name	Art	Typ	Zweck
char	E	I 4	gesuchter Buchstabe einer Registerbezeichnung
maxnmb	E	I 4	höchste Kennzahl des Registers
nmb	A	I 4	Kennzahl der gefundenen Registerbezeichnung

COMMON	<p><u>/IN/</u> ICHNE, ICHUN, ICHIN, LIN</p> <p><u>/LABEL/</u> LABELX, LABELT, LABELS, HSPVT</p> <p><u>/LIDVRS/</u> ZIMERS, SIMBRS, ANMBRS, TMBRS, RMBRS</p> <p><u>/KOUT/</u> KCHNL, LSTELN</p> <p><u>/OUT/</u> OCHNI, OUTPOS, CUTNUM, OUTLIN, NEOCRD</p> <p><u>/SAVDAT/</u> SAVPTR, POS, NUM, LINES</p> <p><u>/CONSTS/</u> BILD, LIES, ANTW, TEST, WENN, NACH, TSFR, RTSP, LADE, UPRO, UEND, HALT, STOP, ENDE, PROG, SPACE, ASTRSK, AFOSTR, POINT, EQUAL, COLOR, CROSS, COMMA, OFEN2, CLOSE2, A, P, M, R, S, T, Z, DIGITS, START</p>
--------	--

Beschreibung:

SCANR bewegt den Eingabezeiger (INPOS in /IN/) auf das nächste Zeichen ungleich Blank durch Aufruf von FINDIC. Ist dieses Zeichen gleich dem angeforderten Zeichen "char" und folgt darauf eine Zahl kleiner gleich "maxnmb", so liefert SCANR "true". "nmb" ist die gefundene Zahl, der Eingabezeiger steht dahinter.



T e i l 2

K a p i t e l 5.2

HHI Berlin	L I F O R T Laufzeitsystem		Katalog-Nr. DG78915
Name	ANTW	UP	Nr. 1
Stichwort	speichert antizipierte Antwort		
Verfasser	Herbert Gold	Version vom 15.9.1978	
Parameter	(ind, ltxt, txt)		
Unterprogramm- aufrufe	LADE		
Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangsparameter &=Marke			
Name	Art	Typ	Zweck
ind	E	I 4	Index des Antwortindikators
ltxt	E	I 4	Länge des Textes
txt	E	I 4	Vorgesehener Antworttext
COMMON	COMMON /LIDIA/ A,S,Z,R,T,M,FRR,I,PROZ,LEX,LKO,LBL,LST, 1 ANT,NAN,NTXT,IF,PGS,NUMLI,SPVLAB,UPRCNT,LABSAY,POINT, 2 EIN,AUS,CLEAR,NUMCHR,LCHR COMMON /SGSC03/ KOMMAN,INPUT,IANLYT,ITIME,IANOUT COMMON /VERW/ FORM		
<u>Beschreibung:</u> ANTW speichert mittels LADE den Antworttext und seine Länge in das entsprechende (durch ind markierte) Antwortregister um. Dabei wird für den nächsten Aufruf von TEST eine Markierung gesetzt.			



HHI Berlin	L I F O R T Laufzeitsystem	Katalog-Nr. DG78915	
Name	BILD	UP	Nr. 2
Stichwort	Ausgabe auf dem Bildschirm		
Verfasser	Herbert Gold	Version vom 15.9.1978	
Parameter	CALL BILD (iprec, iposc, skip, leerz, lue, len, txt)		
Unterprogramm- aufrufe	REST, LUECKE, FSCCR, FILTXT, MZART, INSERT, IBYTE		

Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangparameter &=Marke

Name	Art	Typ	Zweck
iprec	E	I 4	Steuerparameter für Layout vor dem Aufruf
iposc	E	I 4	Steuerparameter für Layout nach dem Aufruf
skip	E	I 4	Zahl der zu überspringenden Zeilen
leerz	E	I 4	Zahl der Leerzeilen
lue	E	I 4	falls größer null = Index des (ersten) Antwortsplits
len	E	I 4	Länge des Textes, neg. Vorzeichen zeigt Split(s) an
txt	E	I 4	Text in Hollerith-Schreibweise

COMMON

COMMON /LIDIA/ A,S,Z,R,T,M,FRR,RPROZ,LEX,LKO,LBL,LST,
 1 ANT,NAN,WTXT,IP,POS,NUMLI,SPVLAB,UPCNT,LASAV,PCINT,
 2 EIN,AUS,CLEAR,NUMCHR,LCHR

COMMON /SGSCOS/ KOMMAN,INPUT,IANYT,ITIME,IANOUT

COMMON /VERW/ FORM

Beschreibung:

Bild generiert die Textausgabe auf dem Bildschirm



HHI Berlin	L I F O R T Laufzeitsystem		Katalog-Nr. DG78915
Name	BLANKS, STW, EXAKT, KONS		UP Nr. 3
Stichwort	Steuerung des Antwort-Tests		
Verfasser	Herbert Gold	Version vom 15.9.1978	
Parameter			
Unterprogramm- aufrufe			
Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangsparemeter &=Marke			
Name	Art	Typ	Zweck
COMMON			COMMON /LIDIA/ A,S,Z,R,T,M,FRR,I,PROZ,LEX,LKO,LBL,I,ST, 1 ANT,NAN,I,XTT,IP,POS,NUCLI,SPVLAB,UPRONT,LABSAY,PCINT, 2 EIN,AUS,CLEAR,KUNCHR,LCHR COMMON /\$GSCOS/ KOMMAN,INPUT,I,ANFYT,ITIME,I,ANOUT COMMON /VERW/ FORM
<u>Beschreibung:</u>			
Diese UP's setzten entsprechende logische Variable im Common, über welche die VERGLEICHs-Routine gesteuert wird.			

Laborprojekt Zweiweg-Kabelfernsehen

HHI

HHI Berlin	L I F O R T Laufzeitsystem	Katalog-Nr. DG78915	
Name	FSPLIT	UP	Nr. 4
Stichwort	find split, finde Split		
Verfasser	Herbert Gold	Version vom 15.9.1978	
Parameter	(text, ian, iart, num)		
Unterprogramm- aufrufe	IBYTE, IZAHL		

Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangparameter &=Marke

Name	Art	Typ	Zweck
text	E	I 4	Eingabetext, der untersucht wird
ian	D	I 4	Zeiger auf den Anfang, wird weiter gesetzt bei pos. Ausgang
iant	A	I 4	zeigt die Art des Splits an: T=1, Z=2, R=3
num	A	I 4	Registernummer des Splits
COMMON	COMMON /LIDIA/ A,S,Z,R,T,M,FRR,NPROZ,LEX,LKO,LBL,LST, 1 ANT,NAN,NTXT,IP,POS,NUMLI,SPVLAB,UPRCNT,LASAV,POINT, 2 EIN,AUS,CLEAR,KU,CHR,LCHR COMMON /SGSCOB/ KOMMAN,INPUT,IANYT,ITIME,IANCUT COMMON /VERW/ FORM		

Beschreibung:

Findet Split in "text" ab "ian".



HHI Berlin	L I F O R T Laufzeitsystem		Katalog-Nr. DG78915
Name	INITAL	UP	Nr. 5
Stichwort	Initialisierung		
Verfasser	Herbert Gold	Version vom 15.9.1978	
Parameter			
Unterprogramm- aufrufe	EXAKT, LADEF, NORMIR		
Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangparameter &=Marke			
Name	Art	Typ	Zweck
COMMON			COMMON /LIDIA/ <u>A</u> , <u>S</u> , <u>Z</u> , <u>R</u> , <u>T</u> , <u>M</u> , <u>ERR</u> , <u>KPROZ</u> , <u>LEX</u> , <u>LKO</u> , <u>LBL</u> , <u>LST</u> , 1 <u>ART</u> , <u>NAN</u> , <u>ITXT</u> , <u>IP</u> , <u>POS</u> , <u>NUKLI</u> , <u>SPVLAB</u> , <u>UPRONT</u> , <u>LABEAV</u> , <u>POINT</u> , 2 <u>EIN</u> , <u>AUS</u> , <u>CLEAR</u> , <u>KURCHR</u> , <u>ICHR</u> COMMON /\$GSC03/ <u>KOMMAN</u> , <u>INPUT</u> , <u>IANLYT</u> , <u>ITIME</u> , <u>IANOUT</u> COMMON /VERW/ <u>FORM</u>
<u>Beschreibung:</u>			
Initialisiert Variable, Felder und Ausgabeinheit. (Schreibt den "Eröffnungstext" umgesetzter Programme)			

HHI Berlin	L I F O R T Laufzeitsystem		Katalog-Nr. DG78915
Name	IZAHL	UP	Nr. 6
Stichwort	Konv. Text zu Zahl		
Verfasser	Herbert Gold	Version vom 15.9.1978	
Parameter	(text, ian, iw, istel)		
Unterprogramm- aufrufe	MZART		
Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangsparameter &=Marke			
Name	Art	Typ	Zweck
text	E	I 4	Eingabetext
ian	D	I 4	Zeiger auf den aktuellen Anfang, wird bei pos. Durchgang erhöht.
iw	A	I 4	gefundenener Integerwert
istel	A	I 4	Stellenzahl des Wertes
COMMON	<pre>COMMON /LIDIA/ A,S,Z,R,T,M,FRR,KPROZ,LEX,LKO,LBL,LST, 1 ANT,RAN,RTXT,IP,POS,NUMLI,SPVLAB,UPRNT,LBEAV,PCINT, 2 EIN,AUS,CLEAR,NUCHR,LCHR COMMON /\$GSC03/ KOMMEN,INPUT,IANEY,ITIME,IANOUT COMMON /VERW/ FORM</pre>		
<u>Beschreibung:</u> Der Eingabetext wird auf eine Integerzahl hin untersucht. Der Wert und die Stellenzahl werden ausgegeben. Im Fehlerfall ist "istel" null.			

HHI Berlin	L I F O R T Laufzeitsystem	Katalog-Nr. DG78915	
Name	LADE	UP	Nr. 7
Stichwort	Laden des Textregisters		
Verfasser	Herbert Gold	Version vom 15.9.1978	
Parameter	(tx, len, text)		
Unterprogramm- aufrufe	LABS, FSPLIT, IBYTE, INSERT, PUTNR, PUTRZ		

Erklärung: · E=Eingangs- A=Ausgangs- D=Durchgangsparameter &=Marke

Name	Art	Typ	Zweck
tx	A	I 4	Ausgabefeld (nach dem Laden)
len	E	I 4	Länge des Eingabetextes
text	E	I 4	Eingabetext
COMMON	COMMON /LIDIA/ A,S,Z,R,T,M,FRR,I,PROZ,LEX,LKO,LBL,LST, 1 ANT,MAN,HTXT,IP,POS,NUCLI,SPVLAB,UPRONT,LBEAV,POINT, 2 EIN,AUS,CLEAR,NUMCHR,LCHR COMMON /\$GSC0\$/ KOMMAN,INPUT,IANEXT,ITIME,IANOUT COMMON /VERW/ FORM		

Beschreibung:

Das Vorzeichen von "len" steuert das (Um)-LADEn. Bei negativem Vorzeichen wird der Eingabetext auf Splits untersucht. Die Länge des Ausgabefeldes steht in tx (13).



HHI Berlin	L I F O R T Laufzeitsystem		Katalog-Nr. DG78915
Name	LADEF	UP	Nr. 8
Stichwort	Lade Format		
Verfasser	Herbert Gold	Version vom 15.9.1978	
Parameter	(ns, nz, nn, kn, kz, ks, ne)		
Unterprogramm- aufrufe			
Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangsparameter &=Marke			
Name	Art	Typ	Zweck
ns	E	I 4	Zahl der "*" vor dem Komma
nz	E	I 4	Zahl der "z"s vor dem Komma
nn	E	I 4	Zahl der "g"en vor dem Komma
Kn	E	I 4	Zahl der "g"en nach dem Komma
Kz	E	I 4	Zahl der "z"s nach dem Komma
Ks	E	I 4	Zahl der "*" nach dem Komma
ne	E	I 4	ne - 1 bei Exponentendarstellung
COMMON	COMMON /LIDIA/ A,S,Z,R,T,M,FRR,I,PROZ,LEX,LKO,LBL,IST, 1 ANT,LAN,ITXT,IP,PGS,NUCLI,SPVLAB,UFRONT,LABEV,PCINT, 2 EIN,AUS,CLEAR,KUNCHR,LCNR COMMON /SGSCG/ KOMMAN,INPUT,IANYT,ITIME,IANOUT COMMON /VERW/ FORM		
<u>Beschreibung:</u> Lädt die Formatparameter zur Ausgabe von Realzahlen in den COMMON/VERW/. Werte wurden in PUTRZ verwendet.			

HHI Berlin	L I F O R T Laufzeitsystem	Katalog-Nr. DG78915	
Name	LADM /Integer Funktion/	UP	Nr. 9
Stichwort	Lade Marke		
Verfasser	Herbert Gold	Version vom 15.9.1978	
Parameter	LADM (text), LABCNT, LABELS)		
Unterprogramm- aufrufe	LADE		

Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangsparameter &=Marke

Name	Art	Typ	Zweck
text	E	I 4	Eingabetext
Labcnt	E	I 4	Anzahl der LABELS
LABELS	E	I 4	Feld mit den Originalstrings der Label
COMMON	<pre>COMMON /LIDIA/ A,S,Z,R,T,M,FRR,IPROZ,LEX,LKO,LBL,LST, 1 ALT,NAN,ITXT,IP,POG,NUMLI,SPVLAB,UPRNT,LABAV,POINT, 2 EIN,AUS,CLEAR,KUCHR,LCHR COMMON /\$GSGO\$/ KOMPLAN,INPUT,IANLYT,ITIME,IANOUT COMMON /VERV/ FORM</pre>		

Beschreibung:

Für den Eingabetext wird untersucht, ob es sich um ein Label handelt. Bei pos. Resultat wird der Index in der Markentabelle mit der Funktion übergeben, andernfalls gibt es eine Laufzeitfehlermeldung mit Pause (dies kann bei korrekten Programmen nicht geschehen). Diese Routine wurde erweitert, um verschiedene Felder mit Marken bearbeiten zu können.



HHI Berlin	L I F O R T Laufzeitsystem		Katalog-Nr. DG78915
Name	LADR	/ Real Fkt. /	UP Nr. 10
Stichwort	Lade Realzahl		
Verfasser	Herbert Gold		Version vom 15.9.1978
Parameter	LADR (TX)		
Unterprogramm- aufrufe	FLOAT, IBYTE, IZAHL		
Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangparameter &=Marke			
Name	Art	Typ	Zweck
TX	E	I 4	Textfeld
COMMON	COMMON /LIDIA/ A,S,Z,R,T,M, <u>FRR</u> ,I,PROZ,LEX,LKO,LBL,LST, 1 ART,KAN,NTXT,IP,POS,NUKLI,SPVLAB,UPRCNT,LASAV, <u>POINT</u> , 2 EIN,AUS,CLEAR,NUCHR,LCHR COMMON /\$GSC03/ KOMPLAN,INPUT,IANYT,ITIME,IANOUT COMMON /VERW/ FORM		
<u>Beschreibung:</u>			
Lädt Realzahl aus dem Textfeld und gibt den Wert als Funktionswert ab.			



HHI Berlin		L I F O R T Laufzeitsystem		Katalog-Nr. DG78915	
Name		LADZ /interger Funktion/		UP	Nr. 11
Stichwort		Lade Integerzahl			
Verfasser		Herbert Gold		Version vom 15.9.1978	
Parameter		(tx)			
Unterprogramm- aufrufe		IZAHL, MZART			
Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangsparameter &=Marke					
Name	Art	Typ	Zweck		
tx	E	I 4	Eingabetext		
COMMON		<pre>COMMON /LIDIA/ A,S,Z,R,T,M,FRR,I,PROZ,LEX,LKO,LBL,LST, 1 ANT,NAN,NTXT,IP,POS,NUKLI,SPVLAB,UPRINT,LABEAV,PCINT, 2 EIN,AUS,CLEAR,NUMCHR,LCHR COMMON /\$GSCOS/ KOMMAN,INPUT,IANYT,ITIME,IANOUT COMMON /VERW/ FORM</pre>			
<p><u>Beschreibung:</u></p> <p>In "tx" werden zunächst führende Leerzeichen übersprungen, danach durch Aufruf von IZAHL der Integerwert an die Funktion übergeben.</p> <p>ACHTUNG: Im Zusammenhang mit der Fehlerbehandlung in LIDIA sind evtl. noch Fehler denkbar. Die Funktion läuft dann auf eine Pause:</p> <p>"Fehler in LADZ". (=runtime Fehlermeldung)</p>					

HHI Berlin	L I F O R T Laufzeitsystem	Katalog-Nr. DG78915	
Name	LIES	UP	Nr. 12
Stichwort	Lies Benutzereingabe		
Verfasser	Herbert Gold	Version vom 15.9.1978	
Parameter	(in)		
Unterprogramm- aufrufe	IBYTE, FSCCR, NORMIR, SENDZ, GSREAD, INSERT		

Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangsparameter &=Marke

Name	Art	Typ	Zweck
in	A	I 4	Ausgabe des eingelesenen Textes
COMMON	<pre>COMMON /LIDIA/ A,S,Z,R,T,M,ERR,H,PROZ,LEX,LKO,LBL,LST, 1 ANT,RAN,RTXT,IP,POS,NUM,LI,SPVLAB,UPRCNT,IABSAV,POINT, 2 EIN,AUS,CLEAR,RUN,CHR,LCNR COMMON /%GSCOB/ KOMMAN,INPUT,IANLYT,ITIME,IANOUT COMMON /VERW/ FORM</pre>		

Beschreibung:

LIES sendet das aufbereitete "Bild" an das Endgerät und erwartet Eingabe (en).

HHI Berlin		L I F O R T Laufzeitsystem		Katalog-Nr. DG78915	
Name		LIESR		UP	Nr. 13
Stichwort		Lies Realzahl			
Verfasser		Herbert Gold		Version vom 15.9.1978	
Parameter		(realz)			
Unterprogramm- aufrufe		LIES, LADR			
Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangsparameter &=Marke					
Name	Art	Typ	Zweck		
realz	A	R 4	Wert der gelesenen Realzahl		
COMMON		<pre>COMMON /LIDIA/ A,S,Z,R,T,M,FRR,I,PROZ,LEX,LKO,LBL,LST, 1 ART,NAN,TEXT,IP,POS,NUMLI,SPVLAB,UPRNT,LABELV,POINT, 2 EIN,AUS,CLEAR,NUMCHR,LCHR COMMON /\$GSC03/ KOMMAN,INPUT,IANYT,ITIME,IANCUT COMMON /VERV/ FORM</pre>			
<u>Beschreibung:</u>					
<p>Erwartet eine Realzahl als Eingabe. Zunächst wird die Eingabe des Teilnehmers mit LIES als Zeichenkette gelesen, anschließend mit der Funktion LADR eine Realzahl aus der Zeichenkette erkannt.</p>					



HHI Berlin		L I F O R T Laufzeitsystem		Katalog-Nr. DG78915	
Name		LIESZ		UP	Nr. 14
Stichwort		Lies Zahl			
Verfasser		Herbert Gold		Version vom 15.9.1978	
Parameter		(izahl)			
Unterprogramm- aufrufe		LIES, LADZ			
Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangsparameter &=Marke					
Name	Art	Typ	Zweck		
izahl	A	I 4	Wert der gelesenen Integerzahl		
COMMON		<pre>COMMON /LIDIA/ A,S,Z,R,T,M,FRR,I,PROZ,LEX,IKO,LBL,IST, 1 ART,KAN,I,XT,IP,POS,NUM,LI,SPVLAB,UPRONT,LABSAY,POINT, 2 EIN,AUS,CLEAR,KUNCHR,LOHR COMMON /SCSCO3/ KOMMAN,INPUT,IANYT,ITIME,IANOUT COMMON /VERW/ FORM</pre>			
<u>Beschreibung:</u> Erwartet eine Integerzahl als Eingabe. Zunächst wird die Eingabe als Zeichenkette gelesen (LIES), anschließend mit LADZ umgewandelt.					

HHI Berlin	L I F O R T Laufzeitsystem		Katalog-Nr. DG78915
Name	LUECKE	UP	Nr. 15
Stichwort	Bearbeitet Antwort-Lücken		
Verfasser	Herbert Gold	Version vom 15.9.1978	
Parameter	(tx, k, ll, l, lue, ic, quot, ibl)		
Unterprogramm- aufrufe	REST, FILTXT, IBYTE		
Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangparameter &=Marke			
Name	Art	Typ	Zweck
Tx	E	I 4	Text der Ausgabe
k	D	I 4	Anfangszeiger in TX für Ausgabe
ll	E	I 4	Endezeiger des Textes
l	D	I 4	Endezeiger für laufende Zeile
lue	D	I 4	Zeiger auf Antwortlücke
ic	D	I 4	Steuerparameter
quot	E	I 2	' in Halbwortdarstellung
ibl	E	I 2	blank in Halbwortdarstellung
COMMON	<pre>COMMON /LIDIA/ A,S,Z,R,T,M,FRR,HPROZ,LEX,LKO,LBL,LST, 1 AKT,HAN,UTXT,IP,POS,NUCLI,SPVLAB,UFRONT,LABELV,POINT, 2 EIN,AUS,CLEAR,KUNCHR,LCHR COMMON /SCSCO3/ KOMMAN,INPUT,IANYT,ITIME,IANOUT COMMON /VERW/ FORM</pre>		
<u>Beschreibung:</u> Füllt den Text in BILD mit Antwortlücken und entspr. steuert das Layout, speichert Position für Teilnehmereingabe.			

HHI Berlin	L I F O R T Laufzeitsystem	Katalog-Nr. DG78915	
Name	PROZ	UP	Nr. 16
Stichwort	Prozentangabe für Antwortangabe		
Verfasser	Herbert Gold	Version vom 15.9.1978	
Parameter	(n)		
Unterprogramm- aufrufe			

Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangsparameter &=Marke

Name	Art	Typ	Zweck
n	E	I 4	Prozentzahl für die Übereinstimmung beim pos. Vergleich der Teilnehmereingabe
COMMON			COMMON /LIDIA/ A,S,Z,R,T,M,ERR,I,PROZ,LEX,LKO,L3L,LST, 1 ANT,KAN,TEXT,IP,POS,KUHLI,SPVLAE,UFRONT,LABELV,PCINT, 2 EIN,AUS,CLEAR,KUNCHR,LCHR COMMON /SGSCOS/ KOMMEN,INPUT,IANLYT,ITIME,IANOUT COMMON /VERV/ FORM

Beschreibung:

Im Common-Bereich wird die Variable NPROZ auf n gesetzt.
Dies dient zur Steuerung der Routine VERGL. NPROZ ist andernfalls null.

HHI Berlin	L I F O R T Laufzeitsystem	Katalog-Nr. DG78915	
Name	PUTNR	UP	Nr. 17
Stichwort	put number, Bildschirmausgabe einer Zahl		
Verfasser	Herbert Gold	Version vom 15.9.1978	
Parameter	(number, length, field, pos. &lab, zero)		
Unterprogramm- aufrufe	IBYTE, INSERT		

Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangsparameter &=Marke

Name	Art	Typ	Zweck
number	E	I 4	auszugebende Zahl
length	E	I 4	Länge des Ausgabebereichs > 0 alle Stellen sind zu nutzen (Festformat) < 0 führende Blanks werden nicht ausgegeben.
field	D	I 4	Ausgabefeld
pos	E	I 4	Anfangsposition für Ausgabe
&lab	A	&	Fehlerausgang
zero	E	I 4	logische Variable, bei .TRUE. werden führende Nullen als solche ausgegeben, sonst als Blanks
COMMON	COMMON /LIDIA/ A,S,Z,R,T,M,FRR,WPROZ,LEX,LKO,LBL,LST, 1 ANT,NAN,NTXT,IP,POS,NUCLI,SPVLAE,UERCNT,LABEAV,POINT, 2 EIN,AUS,CLEAR,KUNCHR,LCHR COMMON /SGSCO3/ KOMMEN,INPUT,IANYT,ITIME,IANOUT COMMON /VERW/ FORM		

Beschreibung:

Ausgabe einer Integerzahl in ein Feld für die Bildschirmausgabe.



HHI Berlin	L I F O R T Laufzeitsystem	Katalog-Nr. DG78915	
Name	PUTRZ	UP	Nr. 18
Stichwort	Ausgabe einer Realzahl		
Verfasser	Herbert Gold	Version vom 15.9.1978	
Parameter	CALL PUTRZ (reale, length, field, pos, &lab)		
Unterprogramm- aufrufe	PUTNR, IBYTE, INSERT		

Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangsparameter &=Marke

Name	Art	Typ	Zweck
realz	E	R 4	Realzahl zur Ausgabe
length	E	I 4	Länge des Ausgabebereichs
field	D	I 4	Ausgabefeld
pos	D	I 4	Ausgabeposition
&lab	A		Fehlerausgang
COMMON	COMMON /LIDIA/ A,S,Z,R,T,M,FRR,I,PROZ,LEX,LKO,LBL,LST, 1 ANT,KAN,UTXT,IP,POS,NUKLI,SPVLAB,UERCNT,IABEV,POINT, 2 EIN,AUS,CLEAR,KUCHR,LOHR COMMON /\$GSCG/ KOMMEN,INPUT,IANLYT,ITIME,IANCUT COMMON /VERW/ <u>FORM</u>		

Beschreibung:

Ausgabe einer Realzahl in ein Feld.

Im COMMON/VERW/ stehen die Parameter, die das Format steuern.

Ist die Zahl im Format nicht darstellbar, so wird auf die Standardeinstellung übergegangen, ist dies auch zu klein, erfolgt Standard-Exponenten-Darstellung.

HHI Berlin		L I F O R T Laufzeitsystem		Katalog-Nr. DG78915	
Name		REST		UP	Nr. 19
Stichwort					
Verfasser		Herbert Gold		Version vom 15.9.1978	
Parameter		CALL REST (ic)			
Unterprogramm- aufrufe		FILTXT, FSCCR			
Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangparameter &=Marke					
Name	Art	Typ	Zweck		
ic	E	I 4	Steuerparameter		
COMMON		<pre>COMMON /LIDIA/ A,S,Z,R,T,M,FRR,WPROZ,LEX,LKO,LBL,LST, 1 AKT,HAN,RTXT,IP,POS,NUMLI,SPVLAR,UERCNT,LABELV,POINT, 2 EIN,AUS,CLEAR,KUNCHR,LCHR COMMON /\$GSC03/ KOMMAN,INPUT,IANLYT,ITIME,IANOUT COMMON /VERW/ FORM</pre>			
<u>Beschreibung:</u> Füllt den Rest einer Ausgabezeile aufgrund verschiedener Parameter ((im Common) mit Blanks bzw. setzt den Cursor auf den Anfang der folgenden Zeile.					

HHI Berlin	L I F O R T Laufzeitsystem	Katalog-Nr. DG78915	
Name	RTSPR	UP	Nr. 20
Stichwort	Antwortvergleich mit Sprung		
Verfasser	Herbert Gold	Version vom 15.9.1978	
Parameter	(&lab, len, tx, Komman)		
Unterprogramm- aufrufe	LADE, VERGL		

Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangsparameter &=Marke.

Name	Art	Typ	Zweck
&lab	A	&	Rücksprungadresse bei pos. Vergleich
len	E	I 4	Länge des Textes, neg. Vorz. zeigt Split an
tx.	E	I 4	Vergleichstext
Komman	E	I 4	Interne Kodierung für gewünschte Kommandotast, Ø = jede

COMMON

```
COMMON /LIDIA/ A,S,Z,R,T,M,FRR,NPROZ,LEX,LKO,LBL,IST,
1 ANT,KAN,TEXT,IP,PGS,RUHLI,SPVLAB,UPRCNT,LABSAY,POINT,
2 EIN,AUS,CLEAR,RUNCHR,LCHR
```

```
COMMON /$GSC03/ KOMMAN,INPUT,IANYT,ITIME,IANOUT
```

```
COMMON /VERW/ FORM
```

Beschreibung:

Der Vergleichstext wird umgeladen, wodurch eventuell vorhandene Splits entwickelt werden. Verglichen wird dieser Text mit der letzten Antwort durch den Aufruf des UP's VERGL entsprechend den aktuellen Vergleichsparametern. Ist ein Kommando (Komman \neq Ø) angegeben, so muß beim Vergleich auch die gedruckte Kommandotaste stimmen. Für die Angabe von Kommandos kann tx nur aus "#" bestehen.

HHI Berlin	L I F O R T Laufzeitsystem	Katalog-Nr. DG78915	
Name	TEST	UP	Nr. 21
Stichwort	Eingabevergleich mit antizipierten Antworten		
Verfasser	Herbert Gold	Version vom 15.9.1978	
Parameter	(tx)		
Unterprogramm- aufrufe	VERGL		

Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangsparameter &=Marke

Name	Art	Typ	Zweck
tx	E	I 4	letzter Eingabetext
COMMON	COMMON /LIDIA/ A,S,Z,R,T,M,FRR,I,PROZ,LEX,LKO,LBL,LST, 1 ANTW,KAN,RTXT,IP,PCS,NUKLI,SPVLAB,UPRCNT,LABEAV,PCINT, 2 EIN,AUS,CLEAR,KU,CHR,LCHR COMMON /\$GSCOB/ KOMMAN,INPUT,IANYT,ITIME,IANOUT COMMON /VERW/ FORM		

Beschreibung:

Der Eingabetext wird mit Texten der Antwortregister mit VERGL verglichen, die Antwortindikationen entsprechend gesetzt. Die Antwortregister erhalten bei ihrer Zuweisung in ANTW eine Markierung, sind jedoch keine Register markiert bei einem Aufruf von TEST, so wird mit allen Antwortregistern verglichen.

HHI Berlin		L I F O R T Laufzeitsystem		Katalog-Nr. DG78915	
Name		TIME / Integer Function		UP	Nr. 22
Stichwort		Abfrage der Uhrzeit			
Verfasser		Herbert Gold		Version vom 15.9.1978	
Parameter		IZ = TIME (i)			
Unterprogramm- aufrufe					
Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangsparameter &=Marke					
Name	Art	Typ	Zweck		
i	E	I 4	Steuerparameter (gedacht für verschiedene Zeitaufrufe) z.Zt. als Dummy verwendet.		
COMMON	<pre>COMMON /LIDIA/ A,S,Z,R,T,M,FRR,NPROZ,LEX,LKO,L3L,LST, 1 AKT,NAN,NTXT,IP,POS,NUMLI,SPVLAB,UFCNT,LABELV,POINT, 2 EIN,AUS,CLEAR,KUNCHR,ICHR COMMON /SGSCO3/ KOMMAN,INPUT,IANYT,ITIME,IANOUT COMMON /VERW/ FORM</pre>				
<u>Beschreibung:</u>					
Die Integerfunktion TIME liefert die Antwortzeit des Adressaten. Dazu wird der entsprechende Wert aus dem COMMON-Bereich der Grundsoftware gelesen.					



HHI Berlin	L I F O R T Laufzeitsystem	Katalog-Nr. DG78915	
Name	TSPR	UP	Nr. 23
Stichwort	Eingabe, Antwortvergleich mit Sprung		
Verfasser	Herbert Gold	Version vom 15.9.1978	
Parameter	(&lab, len, tx, Komman)		
Unterprogramm- aufrufe	LIES, RTSPR		

Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangparameter &=Marke

Name	Art	Typ	Zweck
&lab	A	&	Rücksprung bei Übereinstimmung von "tx" und Eingabe
len	E	I 4	Länge des Textes "tx"
tx	E	I 4	Text der erwarteten Antwort
Komman	E	I 4	erwartete Kommandotaste (interne Kodierung)
COMMON	<pre>COMMON /LIDIA/ A,S,Z,R,T,M,FRR,NPROZ,LEX,LKO,LBL,LST, 1 ANT,KAN,NTXT,IP,POS,NUCLI,SPVLAB,UPRCNT,LABAV,POINT, 2 EIN,AUS,CLEAR,KUNCHR,LCHR COMMON /SGSC03/ KOMMAN,INPUT,IANYT,ITIME,IANOUT COMMON /VERV/ FORM</pre>		

Beschreibung:

TSPR liest eine Antwort, testet und springt entsprechend.
Ist Komman $\neq \emptyset$ so muß beim Vergleich auch die Kommandotaste stimmen.



HHI Berlin	L I F O R T Laufzeitsystem	Katalog-Nr. DG78915	
Name	UEND	UP	Nr. 24
Stichwort	Ende eines Unterprogrammes (LIDIA)		
Verfasser	Herbert Gold	Version vom 15.9.1978	
Parameter	(&lab)		
Unterprogramm- aufrufe			

Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangsparameter &=Marke

Name	Art	Typ	Zweck
&lab	A	&	Rückkehradresse nach Unterprogrammaufruf.
COMMON			<pre>COMMON /LIDIA/ A,S,Z,R,T,M,FRR,UPROZ,LEX,LKO,LBL,LST, 1 ANT,IAN,ITXT,IP,POS,NUKLI,SPVLAB,UPRNT,LABSXV,POINT, 2 EIN,AUS,CLEAR,RUNCHR,LCNR COMMON /\$GSCG/ KOMMAN,INPUT,IANYT,ITIME,IANCUT COMMON /VER/ FORM</pre>

Beschreibung:

UPRO entkellert die letzte Rückkehradresse und legt sie im Common ab. Mit einem Rücksprung zum zentralen Sprungverteiler kann auf die neu generierte Marke nach dem Unterprogrammaufruf gesprungen werden. Marke steht in SPVLAB des COMMONS.



HHI Berlin		L I F O R T Laufzeitsystem		Katalog-Nr. DG78915	
Name		UPRO		UP	Nr. 25
Stichwort		LIDIA Unterprogrammaufruf			
Verfasser		Herbert Gold		Version vom 15.9.1978	
Parameter		(retlab, &lab)			
Unterprogramm- aufrufe					
Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangparameter &=Marke					
Name	Art	Typ	Zweck		
retlab	E	I 4	Die Rückkehradresse (als Integerzahl) des LIDIA-Unterprogrammaufrufs wird gekellert.		
&lab	A	&	Sprungadresse in das Unterprogramm		
COMMON		COMMON /LIDIA/ A,S,Z,R,T,M,KRR,NPROZ,LEX,LKO,LBL,LST, 1 ANT,NAN,UTXT,IP,POS,KUKLI,SPVLAB,UPRCNT,LABELV,POINT, 2 EIN,AUS,CLEAR,KUNCHR,LCHR COMMON /\$GSC03/ KOMMEN,INPUT,IANEYT,ITIME,IANOUT COMMON /VERW/ FORM			
<u>Beschreibung:</u>					
UPRO kellert "retlab" und verzweigt zum Unterprogramm.					

HHI Berlin		L I F O R T Laufzeitsystem		Katalog-Nr. DG78915	
Name		VERGL		UP	Nr. 26
Stichwort		Vergleich von Texten			
Verfasser		Herbert Gold		Version vom 15.9.1978	
Parameter		(tx, at, l, aia, K)			
Unterprogramm- aufrufe		IBYTE, FLOAT, VOCAL			
Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangparameter &=Marke					
Name	Art	Typ	Zweck		
tx	E	I 4	Teilnehmer eingabetext		
at	E	I 4	Antworttext		
aia	A	I 4	log. Antwortindikator		
K	E	I 4	Kommandotaste		
COMMON		COMMON /LIDIA/ A,S,Z,R,T,M,FRR,J,PROZ,LEX,LKO,L3L,LST, 1 AINT,NAN,ITXT,IP,POS,NUMLI,SPVLAB,UERCNT,LABEAV,POINT, 2 EIN,AUS,CLEAR,KUCCHR,ICHR COMMON /\$GSC0\$/ KOMMAN,INPUT,IANYT,ITIME,IANOUT COMMON /VERW/ FORM			
<u>Beschreibung:</u>					
Vergleicht Eingabetext "tx" mit Antworttext "a" und setzt entsprechend den log. Antwortindikator "aia". Vergleicht auch die gedrückte Kommandotaste mit der gewünschten, wenn K ≠ ∅					

HHI Berlin	L I F O R T Laufzeitsystem	Katalog-Nr. DG78915	
Name	VOCAL /log. Funktion/	UP	Nr. 27
Stichwort	Abfrage auf Vokal		
Verfasser	Herbert Gold	Version vom 15.9.1978	
Parameter	VOCAL (in)		
Unterprogramm- aufrufe	IBYTE		

Erklärung: E=Eingangs- A=Ausgangs- D=Durchgangsparameter &=Marke

Name	Art	Typ	Zweck
in	E	I 4	Eingabe zeichen
COMMON			COMMON /LIDIA/ A,S,Z,R,T,M,FRR,I,PROZ,LEX,LKO,LBL,LST, 1 AKT,NAN,TEXT,IP,POS,NUMLI,SPVLAB,UFRONT,LABELV,POINT, 2 EIN,AUS,CLEAR,KUNCHR,LCNR COMMON /SGSCOB/ KOMMAN,INPUT,IANYT,ITIME,IANOUT COMMON /VERW/ FORM

Beschreibung:

Diese log. Funktion stellt fest, ob es sich bei dem Eingabezeichen "in" um einen Vokal handelt.



6 Referenztable der Unterprogramme

Zu jedem Unterprogramm gehört eine Doppelzeile. In der oberen Reihe werden die FORTRAN-Unterprogrammaufrufe gekennzeichnet, in der unteren die Assembler-Unterprogrammaufrufe. Die Nummern der Spalten bezeichnen sich auf das jeweilige FORTRAN oder Assemblerprogramm.

Auf Seite 3 sind die Assemblerprogramme der Grundsoftware kurz erläutert.



6.1 Referenztabelle Unterprogrammaufrufe

Umsetzer (C:CALLS)	1										2																
	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5		
1 LIFORT	C			C	C			C	C	C	C	C	C	C	C	C	C	C	C	C	C	C	C	C	C	C	A
2 ERRMSG													C														B
3 EVAL	C		C										C						C								
4 FIND1C																											
5 FIRSTP	C				C	C			C						C				C								A
6 INCARD																											B
7 LABINX																											
8 OLABDF						C									C												A
9 PUCALL																			C								
10 PUCARD																											A
11 PUCCRD									C																		
12 PUCCHAR									C																		
13 PUGOTO									C	C	C		C	C				C									
14 PUINT	C											C															
15 PULAB	C								C	C																	
16 PWSTR									C	C	C		C					C									
17 PUTEXT	C								C	C	C		C					C									
18 PUTXT											C																B
19 SAVE	C																										A
20 SCNFOR				C																				C			
21 SCNEXP	C	C	C		C						C								C					C			
22 SCNLAB	C					C																		C	C		
23 SCANT				C																							A
24 SCANR				C																							B
25A COMPTX																											
25B EXPATX																											

Grundsoftware (Assembler)



6.2 Referenztabelle Unterprogrammaufrufe

Laufzeitsystem	1																												
	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7		
1 ANTW									C																				
2 BILD																C													
3 Blank,STW EXAKT,KONS		A	A	A	A	A				A	A	A																	
4 FSPLIT							C																						
5 INITAL				C																									
6 IZAHL																													
7 LADE					C																								
8 LADEF																													
9 LADM																													
10 LADR																													
11 LADZ																													
12 LIES																													
13 LIESR																													
14 LIESZ																													
15 LUECKE																													
16 PROZ																													
17 PUTNR																													
18 RUTRZ																													
19 REST																													
20 RTSPR																													
21 TEST																													
22 TIME																													
23 TSPR																													
24 NEND																													
25 UPRO																													
26 VERGL																													
27 VOCAL																													

C : FORTRAN UP
A : Assembler UP



6.3 Liste der verwendeten Assembler-Routinen /3/

- A) COMPTX : Komprimiert Eingabefeld E der Länge L mit n Zeichen pro Wort in das Ausgabefeld A.
CALL COMPTX (E, A, n, 1)
- B) EXPATX : Umkehrung der Routine COMPTX.
Zeichen in A stehen rechtsbündig.
CALL EXPATX (E, A, n, 1)
- 1) FILTXT : Füllt Zeichen in Übertragungspuffer für den Bildwiederholungspeicher des Sichtgerätes.
CALL FILTXT (E, p, a)
E: Eingabefeld, Zeichen komprimiert
p: Anfangsposition in E
a: Anzahl der Zeichen, wenn >∅
Wiederholung des p.Zeichens, wenn <∅
- 2) FILPOS : Setzt Cursor auf Position z,s
CALL FILPOS (z,s)
z: Zeile s: Spalte
- 3) FSCCR : Setzt Zeile auf den Anfang der nächsten Zeile
- 4) FSCHOM : "Cursor Home"
- 5) FSNORM : Löscht den Bildschirm, "Cursor Home"
- 6) GSREAD : Liest eine Zeile zu 48 Bytes in das Feld INPUT im Common/§GSCO§/
Dabei werden weitere Informationen in entsprechenden Common-Variablen gesetzt:
KOMMAN - benutze Kommandotaste
(1=C, 2=B, 3=A, 4=Beratung, 5=Ende,
6=Anfang, 7=Übersicht, 8=Zurück, 9=Weiter)
INPUT(12)- Zeichenkette der Eingabe (4 Zeichen/Wort)
IANBYT - Anzahl der Zeichen in INPUT
(keine "trailing blanks")
INTIME - Antwortzeit in Sekunden
- 7) IBYTE : Integerfunktion liefert das n-te Byte aus Feld F
(4 Zeichen/Wort)
Ergebnis ist INTEGER * 2
I2 = IBYTE (F, n)

- 8) INSERT : Speichert IZ an p-te Stelle im Feld F
 CALL INSERT (F, p, IZ)
 IZ ist INTEGER *2
- 9) MZART : Die Funktion liefert zu einem Zeichen in Feld
 IQUEL an der Stelle ip einen Integerwert iw
 und den Funktionswert mz, der eine Klassifikation
 des Zeichens ermöglicht:
 mz = MZART (IQUEL, ip, iw)

mz	iw	Zeichen
1	0- 9	0 - 9
2	10-15	A - F
3	16-35	G - Z
4	36-40	! : ; , \$
5	41-44	* / + -
6	45-47	' ()
7	48	sonstiges

- 10) NORMIR : Versetzt das Sichtgerät in Grundstellung
 (ohne Parameterangabe). Durch die Parameter-
 angabe (0,1) wird der Zeichenpuffer gelöscht.
- 11) SENDZ : Der Inhalt des Zeichenpuffers wird an den Bild-
 wiederholtspeicher übertragen.

6.4 Referenztabelle der COMMON-Blöcke

Umsetzer	CONSTS	IN	LABEL	LIDVRS	MOUT	OUT	SAVDAT
1 LIFORT	X	X	X	X	X	X	X
2 ERRMSG		X				X	
3 EVAL	X	X				X	
4 FIND1C	X	X					
5 FIRSTP	X	X	X	X	X	X	X
6 INCARD		X					X
7 LABINX			X				
8 OLABDF	X	X					
9 PUCALL	X						
10 PUCARD	X	X				X	
11 PUCCRD	X					X	
12 PUCRCH	X					X	
13 PUGOTO		X			X	X	
14 PUINT	X						
15 PULAB	X					X	
16 PWSTR	X	X				X	
17 PUTEXT	X	X				X	
18 PUTXT							
19 SAVE		X					X
20 SCNFOR	X	X					
21 SCNEXP	X	X					
22 SCNLAB	X	X		X			
23 SCANT		X					
24 SCANR	X	X					

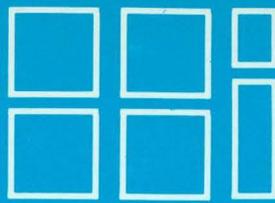
6.5 Referenztabelle der COMMON-Blöcke

Laufzeitsystem	/LIDIA/	/GSCO/	/VERW/
1 ANTW	X		
2 BILD	X		
3 BLANKS,STW EXAKT,KONS	X		
4 FSPLIT			
5 INITAL	X		
6 IZAHL			
7 LADE	X		
8 LADEF			X
9 LADM	X		
10 LADR	X		
11 LADZ	X		
12 LIES	X	X	X
13 LIESR			
14 LIESZ			
15 LUECKE	X		
16 PROZ	X		
17 PUTNR			
18 PUTRZ			X
19 REST	X		
20 RTSPR	X	X	
21 TEST	X		
22 TIME			
23 TSPR	X		X
24 UEND	X		
25 UPRO	X		
26 VERGL	X		
27 VOCAL			

7 Literatur

- /1/ Brüders, Tröller:
"Beschreibung des Displays M1008"
Interne Dokumentation des HHI, 1974
- /2/ Deutsche Bundespost:
"Bildschirmtext", 1977
- /3/ Hunger:
"Grundsoftware für das Display M1008"
Interne Dokumentation des HHI, 1974
- /4/ Kay, Gerensday:
"Social Services and Cable TV"
Report for National Science Foundation. Superintendent of
Documents. US Government Printing Office
Washington DC, 1976
- /5/ Keil:
"Das Projekt 'Computerunterstützter Unterricht, Augsburg'"
Abschlußbericht
Augsburg 1976
- /6/ Siemens:
"LIDIA- Autorensprache für die rechnerunterstützte Unterweisung"
Beschreibung
März 1973
- /7/ Siemens:
"TRANSDATE 8150 Datensichtstation"
Beschreibung
- /8/ Vöge:
"The Two-Way CATV Laboratory Project of the Heinrich-Hertz-Institut
Berlin"
Symposium Münchner Kreis
April 1977
- /9/ Siemens:
"LIDIA - Lernen im Dialog"
Anwendungs- und Sprachbeschreibung
Oktober 1975
- /10/ Mühlbach:
"Benutzersprache und Kommandotasten im Zweiweg-Kabelfernsehen"
Arbeitspapier, HHI, 9.2.78

- /11/ Hummel et al.:
"Beschreibung der Programmiersprache ELAN"
Forschungsprojekt Schulsprache,
TU Berlin, FB 20,
Oktober 1977
- /12/ Schmitt, Hummel:
"Konzeption und Anwendung des RGU-Systems LEKTOR"
Angewandte Informatik 5/76
- /13/ Mahnkopf, Gold:
"LIFORT - ein Programmsystem zur Umsetzung von LIDIA-Programmen"
Angewandte Informatik, 8/78



**Heinrich-Hertz-Institut
für Nachrichtentechnik
Berlin GmbH**

