IDAS

Ein Interaktiver Dialogassembler für KC 87



Der IDAS ist ein Dienstprogramm zur dialogorientierten Erarbeitung von Programmen in der Assemblersprache SYPS K 1520. Er vereinigt in sich Editier-, Übersetzer- und Interpreterfunktionen. Besonders geeignet ist er dann, wenn Befehlsfolgen und kurze Programme (z. B. Unterprogramme für den BASIC-Interpreter) rationell aufgestellt und getestet werden sollen.

Mit IDAS hat der Anwender die Möglichkeit, Programme auf Assemblerniveau zu schreiben. IDAS erzeugt daraus Maschinencode. Das Besondere am IDAS besteht in seiner einfachen Bedienung und der Möglichkeit, das Programm abschnittsweise abarbeiten zu lassen. Damit werden gute Voraussetzungen für einen effektiven Programmtest geschaffen. Der IDAS umfasst die Funktionen eines Editors (Programmaufbereitung), Assemblers (Übersetzung) und Laders (Installieren und Starten des Programms).

Speicherbelegung

Dateiname	IDAS.COM
Laden in	OS
Programmstandort	0400h-1BFFh (RAM-Version) → robotron Kassette R 0122 0C000h-0E7FFh (ROM-Version) → robotron Modul idas-modul
OS-Kommando	IDAS

Der Speicher von 300h-400h wird als Arbeitsspeicher genutzt.

Es gibt zwei ROM-Versionen:
Da der IDAS nur ~ 6KByte Speicher belegt, sind im Modul noch 4K ROM frei. Das robotron-IDAS-Modul meldet sich mit „IDAS :“ und enthält im zusätzlichen Speicher den ZM 2.0 (Z9001 MONITOR V2.0 (ROM) 1985) und die EPROM-Software für das Programmier-Modul. Der IDAS bei U. Zander meldet sich mit „INTERPRETING DIALOG-ASSEMBLER“ und enthält zusätzlich den ZM (Z9001 MONITOR CENT. 1986), RENEW und QUICK,NORMAL,LOAD. RENEW stellt ein versehentlich mit NEW gelöschtes Basic-Programm wieder her. Die Kassettenroutinen QUICK,NORMAL,LOAD erfordern einen speziellen Adapter am User-Port. Sie funktionieren nicht mit dem normalen Kassetteninterface! → quick_v_idas-modul_uzander

Auch von der RAM-Version gibt es verschiedene Fassungen (s.u.).

Downloads

- z9001_idas.zip
  reass. Quellcode zum Neuübersetzen, RAM-Version/ROM-Version/Version UZ
- http://www.sax.de/~zander/z9001/z9_idas.pdf → U. Zander, Beschreibung INTERPRETATIVER DIALOGASSEMBLER

Anleitung

Nach Start erscheint die Meldung 'IDAS:'; mit Eingabe des Buchstabens 'N' erfolgt ein Neustart, jede andere Taste bewirkt einen Restart. Eine Zuweisung der Speicherbereiche fuer Quelltext (TB) und Maschinencode (MC) ist mit dem Kommando **ASGN** möglich.

Standard: MC 1C00H - 1FFFH und TB 2000H - ...

Bei Bereichsüberschreitung erfolgt eine Fehlermeldung (+TB+ / +MC+).

IDAS wird durch das Kommando **CM** (Call Monitor) verlassen.

Im Kommando-Mode werden die eingegebenen Befehle sofort ausgeführt. Die Befehle DI, IM0, IM1 sind hier verboten, da sie die Interrupt-Bearbeitung der CPU beeinflussen und dadurch die Tastaturabfrage blockieren. Danach hilft nur noch RESET. Es ist zu beachten, dass IDAS eventuell vorhandenen Quelltext übersetzt, bevor er einen Befehl im Kommandomode ausführt. Dadurch kann auch im Kommandomode auf Marken des Quelltextes Bezug genommen werden.

Im Editmode erfolgt die Eingabe des Quelltextes in der Form:

```
Zeilennnummer Marke: Befehl
```

Eine Marke kann stehen, muss aber nicht. Kommentare sind nicht möglich.

Das Tabulatorzeichen ist nicht zulässig, es muss mit Leerzeichen gearbeitet werden!

Verarbeitet wird ROBOTRON-Code für U880.

Gegenüberstellung der Schreibweise einiger Beispielbefehle:

```
     ROBOTRON      Zilog

M1:  LD M,A   M1:  LD (HL),A
     JR M1-#       JR M1
     JMP M1        JP M1
     JRC M1-#      JR C,M1
     JPZ M1        JP Z,M1
     CANZ M1       CALL NZ,M1
     DJNZ M1-#     DJNZ M1
     OUT 80H       OUT (80H),A
```

```
     IN  3CH       IN A,(3CH)
     CMP 0         CP 0
     ADD B         ADD A,B
     JMP M         JP (HL)
     RNC           RET NC
```

Steueranweisungen für den Assembler

ORG adresse	Anfangsadresse ; muss im MC-Bereich liegen
konstante:EQU wert	Wertzuweisung zu einem Symbolnamen
BER anzahl	Bereich freihalten (wird nicht gelöscht)
DB byte	Byte definieren
DB 'text'	Text definieren
DA adr	Adresse (Doppelbyte) definieren
END	Ende des Quelltextes ; danach dürfen keine Befehle mehr stehen, sonst Absturz möglich

Zahlenwerte können dezimal (ohne Kennzeichnung) oder hexadezimal (mit nachgestelltem 'H') angegeben werden. Zahlen müssen mit einer Ziffer beginnen, d.h. Hexadezimalzahlen größer 9FH benötigen eine Vornull (z.B.: 0CBH). Die Verwendung von ASCII-Zeichen ist möglich, wenn diese in Hochkommas eingeschlossen werden.

Bsp.: `LD A,75` und `LD A,4BH` und ''LD A,'K' '' sind identisch.

Kommandos

ASGN	Speicherplatzuweisung (s.o.)
CM	Rückkehr ins OS (call monitor)
LIST marke	Anzeige Quelltext ab Marke
LIST zeile1,zeile2	Anzeige Quelltext ab Zeile1 bis Zeile2
STEP abstand	neuer Abstand der Zeilennummern
KILL	Löschen des gesamten Quelltextes
KILL zeile1,zeile2	Löschen Quelltext von Zeile1 bis Zeile2
RUN	ganzes Programm übersetzen und starten
RUN zeile1	Programm ab Zeile1 übersetzen und starten
RUN zeile1,zeile2	wie RUN, aber nur von Zeile1 bis Zeile2
ADDR markename	Ausgabe der Markenadresse
PUT konstante	Ausgabe des Wertes der Konstanten
DR	Anzeige Register (display register) Flags: S Z x H x P/V N C
DS	Speicher anzeigen / ändern (Ende mit 'STOP'-Taste)
READ	Eingabe von Quelltext (an den Anfang) durch 'DEVICE (T/C)' T=Tape oder C=Console zuweisen. READ von Console ermöglicht Eingabe ohne Zeilennummern. Diese werden automatisch erzeugt.
READ zeile1	Einfügen von Quelltext ab Zeile1
READ marke	Einfügen von Quelltext ab Marke

WRIT	Ausgabe von Quelltext durch 'DEVICE (T/C)' T=Tape oder C=Console zuweisen.
WRIT zeile1	
WRIT marke	
MCSV	MC-SAVE. Maschinencode entsprechend der bei ASGN zugewiesenen Adressen auf Kassette schreiben.
MCSV adr1,adr2	Speicherabzug von Adresse1 bis Adresse2 auf Kassette schreiben.

Fehlermeldungen

+ILO+	unzulässige Operation
+IO+	Ein-/Ausgabefehler
+LNR+	unzulässige Zeilennummer
+MDS+	mehrfach definiertes Symbol
+MNE+	falsche Syntax ; unbekannter Code
+NDS+	nicht definiertes Symbol
+OPD+	Operandenfehler bzw. Operand fehlt
+SYM+	Symbolfehler

Ein beliebter Fehler: in der Eingabezeile sind Control-Zeichen enthalten. Deshalb: keine Cursortasten benutzen!

Beispiel

```
IDAS : N
+       1 JR START-#
+       2 TXT: DB 'HALLO Z9001'
+       3 DB 0
+       10 START: LD DE,TXT
+       20 LD C,9
+       30 CALL 5
+       LIST
00001           JR      START-#
00002   TXT:    DB      'HALLO Z9001'
00003           DB      0
00010   START:  LD      DE,TXT
00020           LD      C,9
00030           CALL    5
+       RUN
HALLO Z9001+
+       ASGN
TB (2000H-206BH,BFFFH):
MC (1C00H-1C16H,1FFFH):
```

erzeugter Maschinencode:

```
MC:
1C00   18 0C 48 41 4C 4C 4F 20 5A 39 30 30 31 00 11 02   ..HALLO Z9001...
```

```
1C10   1C 0E 09 CD 05 00 C3 0A 05 00 FF 00 FF 00 FF 00   ................

1C00  18 0C            JR      1C0EH
1C02  48..             DB "HALLO Z9001",0
1C0D  31 00 11         LD      DE,1C02H
1C11  0E 09            LD      C,09H
1C13  CD 05 00         CALL    0005H
1C16  C3 0A 05         JP      050AH
```

JP 050Ah wird automatisch am Ende ergänzt und ist der Rücksprung in die Kommandoschleife.

Textspeicher:

```
TB:
2000   01 00 12 32 02 00 1C 00 4A 52 00 53 54 41 52 54   ...2....JR.START
2010   2D 23 02 00 1B 64 0B 02 1C 54 58 54 00 44 42 00   -#...d...TXT.DB.
2020   27 48 41 4C 4C 4F 20 5A 39 30 30 31 27 03 00 0C   'HALLO Z9001'...
2030   64 01 0D 1C 00 44 42 00 30 0A 00 16 62 03 0E 1C   d....DB.0...b...
2040   53 54 41 52 54 00 4C 44 00 44 45 00 54 58 54 14   START.LD.DE.TXT.
2050   00 0E 62 02 11 1C 00 4C 44 00 43 00 39 1E 00 0E   ..b....LD.C.9...
2060   3A 03 13 1C 00 43 41 4C 4C 00 35 00 FF 00 FF 00   :....CALL.5.....

2000   01 00   12   32 02   00 1C         00 "JR" 00 "START"
2012   02 00   1B   64 0B   02 1C "TXT"   00 "DB" 00 "'HALLO Z9001'"
202D   03 00   0C   64 01   0D 1C         00 "DB" 00 "0"
2039   0A 00   16   62 03   0E 1C "START" 00 "LD" 00 "DE" 00 "TXT"
204F   14 00   0E   62 02   11 1C         00 "LD" 00 "C" 00 "9"
205D   1E 00   0E   3A 03   13 1C         00 "CALL" 00 "5"
206B

2 Byte Zeilennummer (hex)
1 Byte Länge
2 Byte Token + Parameter (zB. Länge bei DB)
2 Byte Adr MC-Code
Marke (optional),Befehl,Parameter,immer durch 00 getrennt.
Token: (Liste  1648h-1837h)
    32 JR
    3A CALL
    62 LD
    64 DB
```

Auf Kassette gespeicherte Quelltexte haben ein Format analog zum EDAS:

- Dateiendung .TXT (wird automatisch angehängt)
- keine Zeilennummern
- Zeilenende 0Dh
- Dateiende 00h

```
Block 00
00000080h: 4A 52 20 53 54 41 52 54 2D 23 0D 54 58 54 3A 20 ; JR START-#.TXT:
00000090h: 44 42 20 27 48 41 4C 4C 4F 20 5A 39 30 30 31 27 ; DB 'HALLO
```

```
Z9001'
000000a0h: 0D 44 42 20 30 0D 53 54 41 52 54 3A 20 4C 44 20 ; .DB 0.START: LD
000000b0h: 44 45 2C 54 58 54 0D 4C 44 20 43 2C 39 0D 43 41 ; DE,TXT.LD
C,9.CA
000000c0h: 4C 4C 20 35 0D 00 00 00 00 00 00 00 00 00 00 00 ; LL
5.............
```

Versionen

Vom IDAS gibt es mehrere verschiedene Versionen, die sich im Detail unterscheiden (Startmeldung, Fehlermeldungen, ...). Die ROM-Versionen enthalten auch verschiedene Zusatzprogramme.

2025 habe ich IDAS reassemblieren müssen[1)] und dabei auch die Versionen miteinander verglichen. Vermutlich ist die Version idas_c0.851 (U. Zander) die ältere der beiden Versionen.

IDAS RAM-Version Kassette R0122
1. Block CRC = DF95, SUM = 0536
2. Block CRC = EF0A, SUM = 51A5
3. Block CRC = 0904, SUM = E84B
4. Block CRC = 6904, SUM = 3FC4

robotron Modul idas.rom
1. Block CRC = 7938, SUM = 8F16
2. Block CRC = 1809, SUM = F6E1
3. Block CRC = F7FA, SUM = 47AF
4. Block CRC = B8EF, SUM = C527
5. Block CRC = E250, SUM = D35F

- idas.rom = IDAS von Modul M002..M006
- Meldung mit „IDAS:“
- enthält ZM Z9001 MONITOR V2.0 (ROM) 1985
- EPROM (Software für robotron-EPROMMER)
- ZM hat Register C und E vertauscht wie in ZM30 der Kassette R0112
- IDAS entspricht der Version von Kassette R0122 (nur auf andere Adr. gelinkt)

ROM-Version U. Zander idas_c0.851
1. Block CRC = A02E, SUM = 8BC7
2. Block CRC = 628C, SUM = FBAD
3. Block CRC = A1C7, SUM = 4EA0
4. Block CRC = BBBA, SUM = C4D0
5. Block CRC = B139, SUM = EE30

- idas_c0.851 = IDAS von UZ,
- Meldung mit „INTERPRETING DIALOG-ASSEMBLER“
- enthält ZM Z9001 MONITOR CENT. 1986
- QUICK, NORMAL, QLOAD (fuer Turbolader mit Extra-Hardware)
- IDAS entspricht nicht der Version von Kassette R0122
- IDAS entspricht nicht der Beschreibung (Seite 3: Nach dem Start meldet sich der Assembler mit der Ausschrift 'IDAS :')

ROM von W. Hasche idas_whasche.rom
1. Block CRC = A02E, SUM = 8BC7
2. Block CRC = 628C, SUM = FBAD
3. Block CRC = A1C7, SUM = 4EA0
4. Block CRC = BBBA, SUM = C4D0
5. Block CRC = 4FDA, SUM = C4F0

- die letzten 2K sind vermutlich falsch EPROM5 = EPROM4 (bis auf ein Bit), damit entspricht idas_whasche.rom höchstwarscheinlich idas_c0.851

ROM-Version Bereich 8000h
diese Version war bei Ebay versteigert worden. Leider ist die Software nicht publik.
2025: Ich kann jetzt aber eine solche ROM-Version kompilieren.

1)

IDAS Laden/Speichern hat nicht im Disk-OS funktioniert, da Speicherzellen im FCB genutzt werden, die auch im Disk-OS verwendet werden. Ein Patch ist jetzt verfügbar.

From:
https://hc-ddr.hucki.net/wiki/ - **Homecomputer DDR**

Permanent link:
https://hc-ddr.hucki.net/wiki/doku.php/z9001/software/idas

Last update: **2026/07/14 05:59**