# Zusatzmonitor ZM

Der Zusatzmonitor ZM stellt eine Erweiterung für das OS des Z9001 dar und bringt maschinennahe Funktionen wie Speichern von Programmen auf Kassette (ja, da gibt es nicht im OS!), Speichereditor (Move, Fill, Search, Edit), Portabfrage, Programmstart mit Breakpoints, Registeranzeige, Hex-Rechnen sowie zusätzliche Gerätetreiber für SIF1000, Drucker über den V24-Port sowie Ausgabe im Intel-HEX-Format oder ASCII auf allen unterstützten Geräten (auch TAPE!).

**Achtung! Der ZM hält sich nicht an die Vorschriften des Z9001 zur Installation von Gerätetreibern etc. Vorhandene Treiber werden einfach überschrieben, ebenso das I/O-Byte. Bei Nutzung der Vollgrafik und Treibern wie CRT80 ist deshalb der ZM in originaler Version nicht nutzbar! Im Mega-Flash sind modifizierte Versionen enthalten, die sich an die Vorgaben halten und mit anderen Programmen zusammenarbeiten**

2021 Zapple_Monitor ist der Urvater des Zusatzmonitors. Kommandos, Bedienung, Implementierung stimmen größtenteils mit dem dem Zapple Monitor 1.1 überein. Leider lässt sich ohne Kontakt zum Autor nicht mehr ermitteln, welche Version tatsächlich Pate gestanden hat. Denkbar ist, dass der RDK-Monitor des mc-Computers zur Verfügung stand. Dieser basiert ebenfalls auf dem Zapple-Monitor.

## Version 2.0

Entwickelt wurde der Zusatzmonitor von Dr. Frank Schwarzenberg, dem Entwickler des CP/A für den Z9001 und vieler anderer Systemsoftware. Die **Version 2.0** entstand 1985 und erschien in verschiedenen Varianten (die Ziffern im Programmnamen hinter ZM stehen dabei nicht für die Version, sondern für die Anfangsadresse des genutzten Speicherbereichs):

- ZM30 der Kassette R 0112 (Gerätegrundkassette),
  neue Variante des ZM 2.0
- ZM30, ZM70, ZMA8 von den Kassetten R 0121 und R 0122 (Assembler bzw. IDAS)
  alte Variante, s. Tabelle
- IDAS-Modul robotron mit ZM im ROM (IDAS ohne Meldung, ZM mit Meldung „Z9001 MONITOR V2.0 (ROM) 1985“ und mit EPROM-Software für das Programmiermodul
  (das ist die neue Variante des ZM 2.0)
- IDAS-Modul Variante U. Zander sowie KCEMU (IDAS mit Meldung „INTERPRETING DIALOG-ASSEMBLER“, ZM mit Meldung „Z9001 MONITOR CENT. 1986“ und mit RENEW und QUICK, NORMAL, QLOAD
  (das ist die alte Variante des ZM 2.0)

Bei den ersten Varianten des ZM und Version 1.2 des OS-KC85/1 kann es bei den Treibern SIFA und TD40 unter Umständen Probleme mit der CTRL/P-Funktion (Hardcopy) geben. Deshalb sei an dieser Stelle auf die erforderlichen Korrekturen für eine saubere Arbeitsweise der ZM-Treiber hingewiesen (aus mp 10/87 S. 311 ff.). In meinem Quellcode ist das entsprechend kommentiert.

| Treiber | relative Adresse | Befehl alt | HEX | → | Befehl neu | HEX |
|---|---|---|---|---|---|---|
| SIFA | 0062H | ld a,e | 7B | | ld a,c | 79H |
| TD40 | 00C3H | res 7,e | CB BB | | res 7,c | CB B9 |

| Treiber | relative Adresse | Befehl alt | HEX | → | Befehl neu | HEX |
|---|---|---|---|---|---|---|
| | 00E0H | sub e | 93 | | sub c | 91 |
| | 00E8H | cp e | BB | | cp c | B9 |
| | 00FCH | ld a,e | 7B | | ld a,c | 79 |
| | 0102H | ld c,154 | 0E 9A | | ld e,154 | 1E 9A |
| | 0117H | ld b,c | 41 | | ld b,e | 43 |

Die **Version 3.0** wurde gegenüber der Version 2.0 stark erweitert. Die wesentlichste Erweiterung stellt das I-Kommando dar, über das eine gewisse Kompatibilität des Betriebssystems des Z 9001 zum Betriebssystem CP/M hergestellt wird. (s.u.)

# Bedienung

Speicherbereiche

ZM30 : 3000-3FFF, 3000
ZM70 : 7000-7FFF, 7000
ZMA8 : A800-B7FF, A800

Hinweise zur Nutzung

- Komandos mit ENTER oder SPACE abschließen.
- Der Unterstrich _ steht in der Syntax für Leerzeichen. Es sind SPACE und Komma zulässig.
- Wird bei der Parametereingabe ein Fehler bemerkt, so kann durch weiterschreiben korrigiert werden, da nur die letzten 4 Zeichen gewertet werden.
- Fortlaufende Bildschirmausgaben können durch SPACE gestoppt werden. Erneute Betätigung setzt Ausgabe fort. STOP führt zum Abbruch des Kommandos. (Ausnahme Kommando S: Dort wird mit SPACE zur nächsten Zelle geschaltet und mit ENTER beendet).

**Kommandos**

| | | |
|---|---|---|
| **A (Assign)** | Gerätezuweisung (s. Treiber) | A<logisches Gerät>=<physisches Gerät> |
| **B (Bye)** | Rückgabe an das OS | B |
| **C (Convert)** | Umrechnung Dezimalzahl in Hexadezimalzahl und umgekehrt | CD(dez.z.) Dez. = > Hex.<br>CH(hex.z.) Hex. = > Dez. |
| **D (Display)** | Ausgabe von Speicherbereichen auf Konsole (Hexdump) | Danfadr_[endadr] |
| **E (Endeblock)** | Ausgabe eines Endeblockes auf das Punch-Gerät (nicht sinnvoll bei AP=T) | E[adr] |
| **F (Fill)** | Beschreiben eines Speicherbereiches mit konst. Wert | Fanfadr_endadr_wert |
| **G (Go)** | Start eines Anwenderprogramms mit max. zwei Haltepunkten. | Ganfadr[_break1][_break2] |
| **H (Hex)** | Bildung der Summe und Differenz zweier Hexzahlen | Hzahl1_zahl2 |
| **J** | Nichtzerstörender Speichertest | Janfadr_endadr |
| **M (Move)** | Transport eines Speicherbereiches auf einen anderen Speicherbereich | Manfadr_endadr_zieladr |

| | | |
|---|---|---|
| **N (Null)** | Ausgabe von binären Nullen auf den Punch-Kanal. (Lochbandvorschub, sinnlos bei AP=T) | N |
| **P (Punch)** | Ausgabe im INTEL-Hex-Format auf den Punch-Kanal | Panfadr_endadr (_stadr bei AP=T) |
| **Q (Query)** | Lesen und Schreiben auf I/O-Ports | QIport Lesen vom Port<br>QOport_byte Schreiben zum Port |
| **R (Read)** | Einlesen eines INTEL-Hex-Files *oder TDL-HEX* über den Reader-Kanal | R[base] |
| **S (Substitute)** | Modifizieren von Speicherzellen. Mit der Leertaste kann zur Adresse weitergeschaltet werden. Mit <ENTER> wird das Kommando beendet. | Sanfadr |
| **T (Type)** | Ausgabe in ASCII-Darstellung | Tanfadr_[endadr] |
| **V (Verify)** | Vergleich eines Speicherbereiches mit einem anderen | Vanfadr_endadr_zieladr |
| **W (Write)** | Schreiben von ASCII-Zeichen in den Speicher. Mit ← kann zurückgeschritten werden. Mit <STOP> wird die Eingabe beendet. Die Adresse des zuletzt eingegebenen Zeichens +1 wird vom Monitor ausgegeben. | Wanfadr |
| **X** | Ausgabe des Z80-Registersatzes | X 1.Registersatz<br>X' 2.Registersatz<br>X[']<reg> Ausgabe und Änderungsmöglichkeit für Register <reg> |
| **Y** | Suchen einer Zeichenfolge von max. 255 byte | Ybyte_byte u.s.w. |
| **Z** | Ausgabe der höchsten RAM-Adresse (RAM-TOP) | |

Für die Kommandos I, K, L, O und U stehen ab 220H fünf Sprungvektoren, über die durch Modifikation der Adressen zusätzliche Kommandoroutinen angebunden werden können.

## Kassettenarbeit

Zum Speichern auf Kassette und zum Lesen werden spezielle Treiber (UP1 und UR1) installiert. Diese umgehen die normale Funktion des P- und R-Kommandos.

### Speichern eines Speicherbereichs

```
AP=T
Paadr_eadr_sadr
```

Es wird der Filename.Typ abgefragt. Ohne Typ wird COM genommen.

Nach dem Speichern kann eine Überprüfung der Aufzeichnung erfolgen. Die Frage „Verify ( (Y)/N )?:“ ist dazu mit Y (oder Enter) zu beantworten.

### Lesen eines Speicherbereichs

```
AR=T
R[ofs]
```

Es wird der Filename.Typ abgefragt. Ohne Typ wird COM genommen. Als Parameter kann ein Offset angegeben werden, der zur originalen Anfangsadresse der Datei addiert wird. Die Datei wird auf die neue Anfangsadresse geladen.

Nach dem Laden werden Anfangs-, End- und Startadresse angezeigt.

# Treiber

Im Zusatzmonitor sind verschiedene Gerätetreiber enthalten. s. auch Artikel über mp 10/87 S. 311 ff.. Den logischen Geräten Console, Reader, Punch, List können verschiedene physische Treiber zugeordnet werden. Im ZM sind physische Treiber für Tastatur mit Beep, SIF-1000 Lochbandleser und -stanzer, V24-Drucker und Kassettenzugriff enthalten.

Das Kommando A<logisches Gerät>=<physisches Gerät> dient der Zuweisung eines vorhandenen Gerätes zu einem logischen Gerät. Folgende Zuordnungen sind angebbar:

| **CONSOLE** | **READER** | **PUNCH** | **LIST** |
|---|---|---|---|
| AC=C für Konsole (mit Tastatur-Beep) | AR=C für Konsole | AP=C für Konsole | AL=C für Konsole |
| AC=V für Konsole (ohne Tastatur-Beep) | AR=P für LB-Leser (SIF-1000) | AP=P für LB-Stanzer (SIF-1000) | AL=V für Konsole |
| AC=B für BATCH | AR=T für Kassette (Tape) | AP=T für Kassette | AL=L für V24-Drucker (TD40) |
| AC=U für User-Konsole | AR=U für User-Read-Gerät | AP=U für V24-Ausgabegerät (TD40) | AL=U für SIF-1000-Ausgabegerät |

Nach Anlauf des Monitors gilt die Zuordnung AC=V, AR=C, AP=C und AL=C (alles auf Konsole).
Soll z.B. ein über den freien PIO-Port angeschlossener V24-Drucker verwendet werden, so muss vorher AL=L gegeben werden. AC=C bewirkt, dass jeder Tastendruck mit einem Ton quittiert wird.

### SIF-1000

Für den BATCH-Modus muss ein SIF-1000-Lesegerät an den PIO-Port angeschlossen werden. Ferner sind die Zuweisungen AR=P und AL=C (oder V) vorher unbedingt erforderlich! Nach AC=B werden jetzt alle Eingaben nicht mehr von der Tastatur, sondern vom angeschlossenen Lesegerät angefordert. Protokolliert wird infolge der Zuweisung AL=C nach wie vor auf dem Bildschirm. Eine Zuweisung AL=L oder U ist in diesem Fall nicht möglich, da an den PIO-Port zusätzlich zum Lesegerät kein LIST-Gerät anschließbar ist.

User-Port-Belegung für SIF-1000-Anschluss:

```
    A    B    C
---------------------------
1   0     0    /END
2   /DAT1    /DAT2    /DAT3
3   /DAT4     /DAT5     /DAT6
4   /DAT7     /RUF    -
5   (+5V)--> +5V    -
```

### Drucker

Sollte der Druckermodul gesteckt sein, so wird bei AC=C auf dem Drucker protokolliert, bei AL=V auf dem Bildschirm. Bei AL=L erfolgt die Druckausgabe auf einen an den PIO-Port angeschlossen seriellen-Drucker TD40.

User-Port-Belegung V24-Ausgabe TD40

```
    A    B    C
---------------------------
1   0    0    -
2   TxD    -    -
3   -    -    -
4   -    DTR*    -          * DTR-Signal des Druckers
5   (+5V)    -    -
```

Bei der ROM-Version ZM2.0 CENT. 1986 und auch im ZM2.0A erfolgt mit AL=V die Druckausgabe auf einen an den PIO-Port angeschlossen Centronics-Drucker.

CENTRONICS-Druckerschnittstelle über User-Port

- 7 Datenbits verfuegbar
- PIO-RDY wird zur Bildung des CENTRONICS-/STROBE-Signals verwendet
- /ACKNLG vom Drucker wird ueber den CTC-Kanal erfasst.

User-Port-Belegung fur CENTRONICS-Anschluss

```
    A    B    C
---------------------------
1   0     0    /ACK
2   DAT1    DAT2    DAT3
3   DAT4    DAT5    DAT6
4   DAT7     -       /STROBE
5   (+5V)==> +5V    -
```

Details zum Hardware-Anschluss siehe mp-Artikel mp 10/87 S. 311 ff. und Quellcode.

Der ZM nutzt **nicht** die ASGN-Anweisung. Beim start des ZM wird die Zuordnungstabelle mit den entsprechenden Adressen geladen, da alle Treiber resident sind. Die Auswahl der ZM-Treiber muss deshalb mit dem ZM-Kommando A(ssign) erfolgen (Stellen des I/O-Bytes).

# Quellcode

* zm.zip

Im Quellcode-Paket ist mein reassemblierter Quellcode für die verschiedenen ZM 2.0-Varianten zu finden. Der Code enthält einige interessante Assemblerabschnitte. Allerdings hält sich der ZM nicht an die Konventionen des OS bezüglich des ASSIGN-Kommandos. Aktive eigene OS-Treiber werden beim Starten des ZM gnadenlos überschrieben, Außerdem werden keine Treibernamen hinterlegt.

30.4.15: Im Paket ist eine modifizierte Version des ZM 2.0: hier werden die Gerätetreiber etwas OS-freundlicher eingebunden: vorhandene Treiber werden nicht einfach überschrieben, und bei ASGN

werden Treibernamen angezeigt. Diese Version funktioniert recht gut z.B. mit den CRT40P und CRT80P-Treibern der KRT-Grafik. Das I/O-Byte wird beim Start nicht verändert, es bleiben die OS-Zuweisungen gültig. Bei Bedarf muss man PUNCH und READER die Konsole selbst zuweisen[1].

2020: die aktuelle modifizierte Version des ZM 2.0 initialisiert wie im Original AR=C, AP=C, eine Manipulation der Treibertabelle ist nicht mehr nötig!




# Allgemein verwendbare Monitor-Routinen

Der Monitor stellt über eine Sprungtabelle eine Reihe von Subroutinen von allgemeiner Bedeutung bereit. Leider war diese Sprungtabelle durch Robotron nicht dokumentiert, so dass kein Programm bekannt ist, das diese Schnittstelle nutzt.

Die Sprungtabelle steht direkt an Monitor-Anfang (= base).

| rel.Adr. (hex.) (base + ) | Subroutine | Register Ein | Register Aus | Register verändert |
|---|---|---|---|---|
| 0000 | Monitor-Neustart | | | |
| 0003 | Trap-Eingang (RST 38H) | Prozessorzustand wird für GO-Routine gerettet | | |
| 0006 | Error-Eingang | SP wird initialisiert | | |
| 0009 | I/O-Byte abfragen | - | A | |
| 000C | I/O-Byte setzen | C | - | AF |

| rel.Adr. (hex.) (base + ) | Subroutine | Register Ein | Register Aus | Register verändert |
|---|---|---|---|---|
| 000F | Test RAM-Größe | - | B | AF |
| 0012 | Eingabekonvertierung | (CI) | HL | AFCDE |
| 0015 | Ausgabekonvertierung | HL | (CO) | AFBCDEHL |
| 0018 | HEX-Eingabe (2 Byte)<br>C=Anzahl der Eingaben | (CI) | (SP) | AFBCHL |
| 001B | HEX-Ausgabe | A | (CO) | AF |
| 001E | BIT-Ausgabe | | | |
| 0021 | Kassetten-Ein/-Ausgabe Mit dieser Funktion sind Speicherbereiche als Files von/auf Kassette lad-/entladbar.<br>Eintrittsparameter:<br>(69H) : =0 Aus-, sonst Eingabe<br>(58H) : Verschiebung bei Eingabe<br>(6DH) : Anfangsadr. für Ausgabe<br>(6FH) : Endeadr. für Ausgabe<br>(71H) : Startadr. | | | |

Der ZM nutzt den **RST 38** (Trap), siehe auch GO-Kommando.

Neben dem ZM selbst ist im Code auch das OS-Kommando **RENEW** enthalten. Mit diesem Kommando kann ein versehentliches NEW im ROM-BASIC wieder rückgängig gemacht werden. BASIC löscht nicht den Programmspeicher, sondern setzt nur interne Variablen. Dies macht RENEW rückgängig. Dazu ist BASIC zu verlassen, auf OS-Ebene RENEW zu tippen und dann mit WBASIC zum BASIC zurückzukehren.

# Version 3.01

s.a. mp 02/89, S.57

Der **Zusatzmonitor ZM 3.0** ist eine stark erweiterte Form gegenüber der ursprünglichen von Robotron. Die Erweiterungen betreffen folgende Leistungen

- Herstellung einer Quasi-CP/M-Kompatibilität. In diesem Betriebsmode können eine Reihe von CP/M-Programmen ohne Änderung auf dem KC abgearbeitet werden (z. B. Turbo-Pascal, ZSID, MBasic etc.).
- Neue User-Port-Treiberroutinen für V.24-DTR-Protokoll (z. B. 1200 und 9600 Baud) und Centronics. Drucker wie Epson LX86 oder LX1000 (auch K6313) sind ohne Druckermodul betreibbar.
- Verbesserte Kassetten-E/A-Routinen
- Umschaltung 20/24-Zeilen-Bildschirmmode.

Neue Kommandos:

| | | |
|---|---|---|
| **I (Init CP/M)** | Dieses Kommando stellt CP/M-Kompatibilität für Anwenderprogramme her. Programme können jetzt ab der Anfangsadresse 100H geladen werden (nicht mehr 300H!). Die Verarbeitung der BDOS-Rufe erfolgt wie unter CP/M üblich. Diskettenrufe werden auf Kassette umverlegt! Zusätzlich wird der Standard-FCB (jetzt ab 5CH) mit filename.typ initialisiert. | I[Filename[.typ]] |
| **K (Kassette)** | Tapelook | |
| **L (Laden)** | Laden von Programmen, die auf Festwertspeicher vorliegen und der Konventionen für das symbolische Aufrufen von Anwenderprogrammen des 0S-Z 9001 entsprechen. | L<name> |
| **O (Off CP/M)** | wieder ^ im Z 9001-Mode | |

Die Sprungtabelle am Monitor-Anfang ist um eine Funktion erweitert:

| **rel.Adr. (hex.) (base + )** | **Subroutine** |
|---|---|
| 0024 | Call 5 : BDOS-Ruf im CP/M-Regime |

Bislang sind drei verschiedene Varianten des ZM 3. Version bekannt. Im Quellcode-Paket sind alle Versionen (reassembliert) enthalten.

Die Version 3.01 arbeitet auf einem Z9001 mit 2 RAM-Modulen und bietet durch die eingebaute BDOS-Emulation eine CP/M-Kompatibilität.

Die vorliegende Version 3.01 wurde offenbar nachträglich gepatcht und bietet veränderte Kommandos:

- **Kommando E** → Einzelschritt (Step), nach GO kann so ein Programm schrittweise ausgeführt werden, nach jedem Schritt werden alle Register, alle Flags und der Speicher angezeigt.
- **Kommando N** → CRC-Berechnung (CRC16, SDLC)
- Es wird ein PIO-Modul mit 2 PIO initialisiert (cmd_cold2) PIO1C (FA, FB), PIO2C (FE, FF) (Vermutlich das **PARALLEL-INTERFACE ZfK 9902)**

# Version 3.02

Die verschiedenen vorhandenen Versionen 3.02 unterscheiden sich i.W. nur in den implementierten Treibern für Drucker, und den unterschiedlichen Start-Möglichkeiten diverser anderer Programme. Der Hauptunterschied zur Version 3.01 ist die Benutzung eines 64K-RAM-Moduls **64 KB DRAM ZfK 9906**. Im Shadow-RAM 4000H-7FFFH werden Teile des ZM kopiert, außerdem wird die verdeckte Speicherbank des 64-KByte-DRAM-Moduls für die Abbildung direkter (random) Diskettenrufe des CP/M genutzt. Weiterhin ist die Sprungtabelle am ZM-Anfang gegenüber Version 3.01 erweitert und nicht mehr kompatibel. Die Unterstützung von SIF-1000 wurde nicht mehr implementiert. Deshalb sind auch die Batch-, Punch- u.a. Routinen nur noch rudimentär vorhanden.

- ZM302_BITEX
  - Diese Version steckt in einem ROM (C000-CFFF, ab D000 BITEX).
  - läuft mit 2 RAM-Modulen (das 64K-Modul wird nicht ausgenutzt)
  - Für LIST sind 2 Treiber vorhanden: V24 und CENTRONICS
  - OS-Kommandos: ZM, RZM,
- BITEXZM302_CPM
  - Diese Version wird von Kassette geladen (1B00-2AFF) und nach 0D800h in den 64K-RAM

kopiert.
    - Benutzung eines 64K-RAM-Moduls **64 KB DRAM ZfK 9906**
    - 1 LIST-Treiber
    - zusätzlicher Code wg. Shadow-RAM-Umschaltung
    - OS-Kommandos: ZMCPM, RZMCPM
- ZM302_BOOT
    - Das ist die umfangreichste Version. Sie steckt in einem schaltbaren ROM-Modul **10 KB SCHALTBARER EPROM ZtK 9905.**
    - Benutzung eines 64K-RAM-Moduls **64 KB DRAM ZfK 9906.**
    - Für LIST sind 2 Treiber vorhanden: V24 und CENTRONICS
    - OS-Kommandos: ZM, RZM, BASIC (da das integrierte BASIC durch 64K RAM und Schalt-ROM verdeckt ist), wbasic, BITEX, EDIT (für weitere schaltbare Module), CPM (Starten des CP/A-Urladers)

1)

Das ist leider nicht so einfach, da im OS ein ASGN PUNCH:=CRT nicht zulässig ist. So muss man die Treibertabelle manipulieren: auf EFD1 DA 0F8F1h (TTY-Treiber für READER) und EFD9 DA 0F8F1h (TTY-Treiber für PUNCH), d.h. SEDF1 F1 F8, SEDF9 F1 F8, und dann AR=C, AP=C

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

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

Last update: **2025/01/13 14:34**