

Achtung: Vor der Programmierung  
einmal NEW eingeben!

## S-BASIC-COMPILER

---

Mit diesem Basic-Compiler können mit dem MZ-700 die Basic-Programme das beim Kauf des Computers miterworbenem Basic-Interpreters MZ-013B direkt ohne Diskettenlaufwerk kompiliert, ausgeführt und editiert werden. Diese Anleitung setzt natürlich voraus, daß der Leser bereits das MZ-700 Bedienerhandbuch durchgelesen und einigermaßen verstanden hat.

## Programmierbare Befehle

---

Im Programmtext sind alle Basicbefehle des Interpreters verwendbar außer denen der fehlerbehandelnden Befehlsgruppe, also

ON ERROR GOTO  
RESUME  
IF ERL  
IF ERN

Die Beschreibung der einzelnen Befehle entnehmen sie bitte dem MZ-700-Bedienerhandbuch. Außerdem fehlen natürlich wie in jedem Compiler die typischen Editor-Befehle SAVE VERIFY LIST AUTO DELETE MERGE RENUM CONT.

Zusätzlich existiert noch

BREAK ON und BREAK OFF schaltet die Breaktaste während der Programmausführung ein oder aus  
\$ML hh, hh, hh ... Hexadezimale Z-80 Befehle werden an dieser Stelle in das Programm eingefügt ( \$ML 21, #3000 lädt HL mit der Anfangsadresse der Zeilennummer 3000 )  
DOKE und DEEK : wie POKE und PEEK, aber 16bit werden in zwei Bytes gespeichert ( immer im 64k-Hauptspeicher )  
AND OR : wie in Maschinensprache z.B. A=B AND 127 aber auch IF (...) AND (...) THEN...

## Editor-Befehle

---

Der Editor des Basic-Compilers entspricht im wesentlichen dem des Interpreters. Programmzeilen werden bei Änderung einfach neu eingegeben oder (wenn sie schon am Bildschirm stehen) mit dem Cursor geändert und eingegeben ( = CR gedrückt ). Folgende Editor-Befehle können im Direct-Mode verwendet werden (jedoch nicht als Mehrfachanweisungen wie z.B. LOAD:LIST):

LOAD ; Laden eines MZ-700-Basic-Programms  
SAVE ; Schreiben des im Speicher befindlichen Basic-Programms auf Band  
VERIFY ; Vergleichen des gespeicherten Programms mit dem im Speicher befindlichen  
LIST , LIST x , LIST x-y , LIST x- , LIST -y ; listet die angegebenen Zeilen  
NEW ; Löschen des Basic-Programms

DELETE x , DELETE x-y , DELETE x- , DELETE -y ; löscht die angegebenen Zeilen  
FINDtext ; sucht den angegebenen Text (alle Zeichen, die direkt nach FIND stehen)  
BYE ; Sprung in den Systemmonitor ( zurück mit R )  
BOOT ; Sprung in den Monitor-ROM ( zurück mit # )  
KEYLIST ; listet Funktionstasten  
PLOT ON ; schaltet Ausgabe auf den Farbplotter ( PLOT ON S bedeutet kleine Schrift, PLOT OFF schaltet wieder aus )  
TEST ; Farbplottertest  
Bei LOAD, SAVE, VERIFY und EXEC (siehe unten) können auch Programmnamen in Anführungszeichen angegeben werden.  
Das SAVEN eines Programms dauert nur noch solange wie das Laden.  
Bei Cassettenlesebefehlen wie z.B. LOAD kann durch Drücken der Shift-Taste mitgehört werden ( erleichtert Programm-Suche ).

#### Compiler-Befehle

RUN startet das kompilierte Objectprogramm. Wenn es noch nicht kompiliert wurde oder seit dem letzten Kompilieren Zeilen geändert wurden, wird vor der Ausführung automatisch COMP ausgeführt.  
EXEC beinhaltet die Befehlsfolge LOAD:COMP:RUN, also Laden, Kompilieren und Ausführen eines Programms.

COMP Kompilieren eines Programms. Fehlerhafte Zeilen werden ausgedruckt. Treten LINE NOT FOUND ERRORS auf, so werden diese zwar angezeigt, das Objectprogramm wird jedoch trotzdem so erzeugt, daß diese Fehler erst bei der Ausführung den Programmablauf abbrechen.  
COMP,R bedeutet Kompilieren und Ausführen des Programms.

COMP,N erzeugt ein Objectprogramm, das bei Fehlermeldungen die gegenwärtig abgearbeitete Zeilennummer anzeigt. Verbraucht zusätzlichen Speicherplatz. Achtung bei Verwendung von \$ML und COMP,N zusammen: Keine Mehrbyte-Anweisungen in getrennte \$ML-Zeilen schreiben, also nicht 10 \$ML CD und 20 \$ML 3E,00 sondern  
10 \$ML CD,3E,00.

COMP,D wird verwendet, wenn bei COMP, RUN oder EXEC die Meldung MEMORY ERROR ausgegeben wird. Es sind dann Basic-Programme bis etwa 24 KByte kompilierbar, was jedoch in jedem Fall mit der Löschung des Basic-Quelltextes erkauft wird. Werden fehlerhafte Zeilen mit DATA- oder REM-Anweisungen ausgedruckt, dann fehlt der Text dahinter. Außerdem treten bei vielen DATA- und REM-Anweisungen längere Kompilierzeiten auf.

COMP,M (beinhaltet COMP,D) gibt dem Objektprogramm maximale Speicherkapazität für Variablenfelder und Strings, ist also für Programme, in denen während der Ausführung OUT OF MEMORY ERROR auftritt. Allerdings ist nach der Ausführung des Programms in der Regel der Compiler zerstört.

COMP"NAME" Kompilieren, danach Schreiben des im Monitor laufenden Objektprogramms auf Cassette. Wird das Maschinenprogramm dann vom Monitor geladen und ausgeführt, so erhält es automatisch den gesamten verfügbaren Speicherplatz des MZ-700.

Es ist auch möglich, die verschiedenen Modi mit Kommas aneinanderzuhängen, z.B. COMP,N,R,"NAME".

#### Bemerkungen

Es sind maximal 15 ineinandergeschachtelte FOR,... NEXT-Schleifen zulässig .

Die Schleifenvariable nach FOR kann kein Arrayelement sein.

Um den RESTORE n-Befehl zu verwenden muss in der Zeile n anstatt "DATA" "\*DATA" geschrieben werden.

# S - B A S I C - C O M P I L E R

=====

Mit diesem Basic-Compiler können mit dem MZ-700 die Basic-Programme des beim Kauf des Computers miterworbenen Basic-Interpreters IZ-013B direkt ohne Diskettenlaufwerk kompiliert, ausgeführt und editiert werden. Diese Anleitung setzt natürlich voraus, daß der Leser bereits das MZ-700 - Bedienerhandbuch durchgelesen und einigermaßen verstanden hat.

## ----- Programmierbare Befehle :

Im Programmtext sind alle Basicbefehle des Interpreters verwendbar außer denen der fehlerbehandelnden Befehlsgruppe, also

ON ERROR  
RESUME  
IF ERL  
IF ERN

Die Beschreibung der einzelnen Befehle entnehmen Sie bitte dem MZ-700 - Bedienerhandbuch. Außerdem fehlen wie in jedem Compiler die typischen Editor - Befehle SAVE, VERIFY, LIST, AUTO, DELETE, MERGE, RENUM und CONT.

Zusätzlich existieren noch : BREAK ON und BREAK OFF.

BREAK ON und BREAK OFF schaltet die BREAKTASTE während der Programmausführung ein oder aus.

\$ML hh, hh, hh, hh ... Hexadezimale Z-80 - Befehle werden an dieser Stelle in das Programm eingefügt ( \$ML 21, #3000 lädt HL mit der Anfangsadresse der Zeilennummer 3000 ).

DEEK und DOKE : wie POKe und PEEK, aber 16 Bit werden in zwei Bytes gespeichert ( immer im 64k-Hauptspeicher ).

AND, OR : wie in Maschinensprache z.B. A=B AND 127 aber auch : IF (.....) AND (.....) THEN ... .

## Editor - Befehle :

-----

Der Editor des Basic - Compilers entspricht im wesentlichen dem des Interpreters. Programmzeilen werden bei Änderung einfach neu eingegeben oder ( wenn sie schon am Bildschirm stehen ) mit dem neuen Text überschrieben und mit CR bestätigt.

Folgende Editor - Befehle können im Direct - Mode verwendet werden ( jedoch nicht als Mehrfach - Anweisung wie z.B. LOAD:LIST ) :

LOAD : Laden eines MZ-700 - Basic - Programmes .

SAVE : Schreiben eines im Speicher befindlichen Basic - Programmes auf Bandcassette .

VERIFY : Vergleichen des geSAVEten Programmes mit dem im Speicher befindlichen .

LIST, LIST x, LIST x-y, LIST x-, LIST -y : listet die angegebenen Zeilen auf .

NEW : Löscht das Basic - Programm im Speicher .

DELETE x , DELETE x-y , DELETE x- , DELETE -y : Löscht die angegebenen Zeilen .

FINDtext : Sucht den direkt nach FIND angegebenen Text .

BYE : Sprung in den System - Monitor ( zurück mit R ) .

BOOT : Sprung in den ROM - Monitor ( zurück mit # ) .

KEYLIST : Listet die Funktionstasten - Belegung auf .

PLOTT ON : Schaltet die Ausgabe auf den Farbplotter um ( PLOTT ON S : bedeutet kleine Schrift ) .

PLOTT OFF : Schaltet den Plotter wieder ab .

TEST : Farbplotter - Test .

Optionen :

Bei LOAD , SAVE , VERIFY und EXEC ( siehe unten ) können auch Programmnamen in Anführungszeichen angegeben werden .

Das SAVEN eines Programmes dauert nur noch so lange wie das Laden .

Bei Cassetten - Lesebefehlen wie z.B. LOAD kann durch drücken der SHIFT-TASTE mitgehört werden zur leichteren Programm - Suche .

Compiler - Befehle :

RUN : Startet das compilierte Object - Programm . Wenn es noch nicht compiliert wurde oder seit dem letzten compilieren Zeilen geändert wurden , wird vor der Ausführung automatisch COMP ausgeführt .

EXEC : Beinhaltet die Befehlsfolge LOAD:COMP:RUN , also laden , compilieren und ausführen eines Programmes .

COMP : Compilieren eines Programmes . Fehlerhafte Zeilen werden auf dem Bildschirm angezeigt . Treten LINE NOT FOUND ERRORS auf , so werden diese zwar angezeigt , das Object - Programm wird jedoch trotzdem so erzeugt , das diese Fehler erst bei der Ausführung den Programm - Ablauf abbrechen .

COMP,R : Bedeutet compilieren und ausführen eines Programmes .

COMP,N : Erzeugt ein Object - Programm , daß bei Fehlermeldungen die gegenwärtig abgearbeitete Zeilennummer anzeigt . ( Verbraucht zusätzlichen Speicherplatz ! ) .

Achtung bei Verwendung von COMP,N und \$ML zusammen : Keine Mehrbyte - Anweisungen in getrennte \$ML - Zeilen schreiben ( also nicht :  
10 \$ML CD            und            20 \$ML 3E,00            sondern            10 \$ML CD,3E,00 ) .

COMP,D : Benutzt man , wenn bei COMP , RUN oder EXEC die Meldung OUT OF MEMORY ausgegeben wird . Es sind dann Basic - Programme bis etwa 24k-Byte compilierbar , was jedoch in jedem Fall mit der Löschung des Basic - Quelltextes erkauft wird . Werden fehlerhafte Zeilen mit REM oder DATA ausgedruckt , dann fehlt der Text dahinter . außerdem treten bei vielen DATA - oder REM - Zeilen längere Compiler - Zeiten auf .

COMP,M : ( Beinhaltet COMP,D ) Gibt dem Object - Programm maximale Speicherkapazität für Variablenfelder oder Strings , ist also für Programme in denen während der Ausführung OUT OF MEMORY ERROR auftritt . Allerdings ist nach der Ausführung des Programmes in der Regel der Compiler zerstört .

COMP,"Name" : Compilieren , danach Schreiben des im Monitor lauffähigen Object - Programmes auf Bandcassette . Wird das Object - Programm dann vom Monitor geladen und ausgeführt , so erhält es automatisch den gesamten , verfügbaren Speicher des MZ-700 zur Programmausführung .

Es ist auch möglich , die verschiedenen Modi mit Komma aneinanderzuhängen , z.B. COMP,N,R,"Name" .