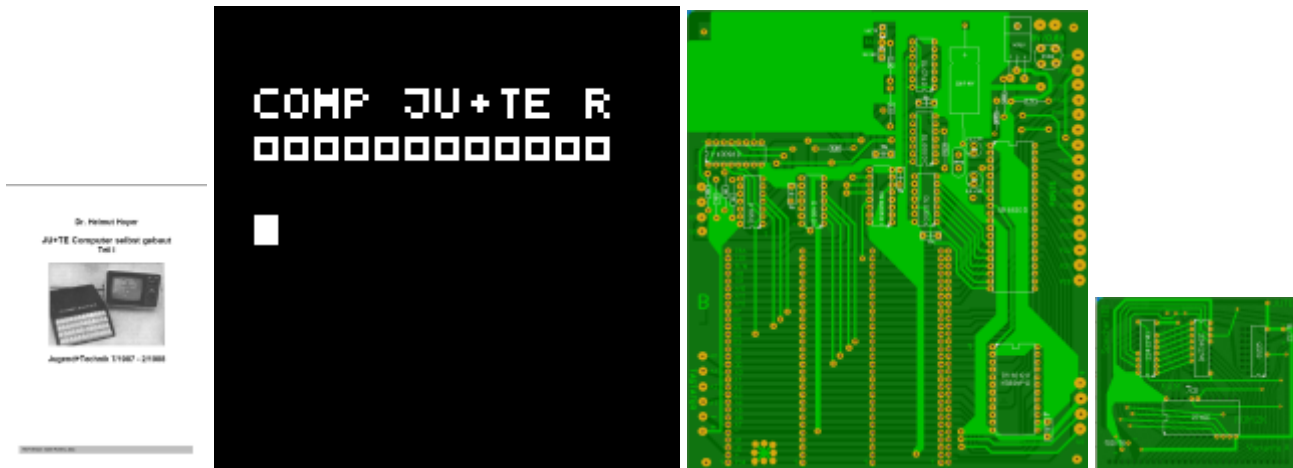


# 2K-System

## Ausbaustufe I



Die Ausbaustufe I des TINY ist die ursprüngliche Version, wie sie in der Zeitschrift Jugend+Technik 7/1987 - 2/1988 beschrieben wurde.

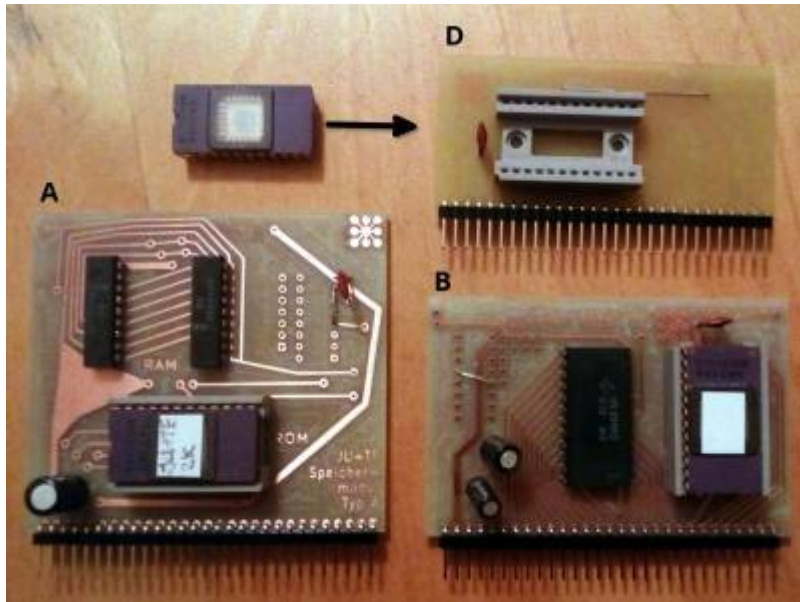
Benötigt werden

- Prozessorplatine
- Speichermodul A mit 1K RAM und 2K EPROM, auf Steckplatz 1 (ganz rechts)
- TV-Anschluss
- Tastatur-Anschluss
- Magnetbandanschluss-Platine
- EPROM mit 2K-Betriebssystem

Im Speichermodul A muss bei Minimalbestückung (1K RAM, kein Huckepack) der Dekoder DS 8205 durch eine Brücke ersetzt werden (Pin 4 und 7 brücken). /CSA geht dadurch direkt an den RAM, damit dieser im Bereich E000..E3FF (BASIC-Programmspeicher) und auch parallel in FC00..FFFF (Systemspeicher und Bildwiederholtspeicher) angesprochen wird.

Speichermodul B mit U 6516 D: Die minimale Bestückung erfordert einen U 6516 D. Der Dekoder DS 8205 D darf nicht eingesetzt werden, wenn der minimal bestückte Modul auf Steckplatz 1 zum Einsatz kommt. Statt des Dekoders DS 8205 D sind ein Durchkontakt anstelle des Pins 16 und eine Drahtbrücke zwischen den Anschlußbohrungen für die Pins 5 und 12 einzusetzen.

Zusätzlich zur Tastatur wird noch eine RESET-Taste gebraucht, die den Anschluß 6 des UB 8830 D mit Masse (00) verbindet. Ein entsprechender Anschlußpunkt befindet sich auf der Prozessorplatine direkt neben dem Einchiprechner-Schaltkreis.



Speichermodule für Tiny Ausbaustufe I, A (1k+1k) und B (2k+2k) funktionieren einwandfrei.  
D ist das Erweiterungsmodul für Ausbaustufe II

## Downloads

- Beschreibung
- ROM-Inhalte
- Quellcode System
- Anwendungen

## Bedienung

*J.Müller, Hilfe zu JTCEMU*

## Das originale 2K-System

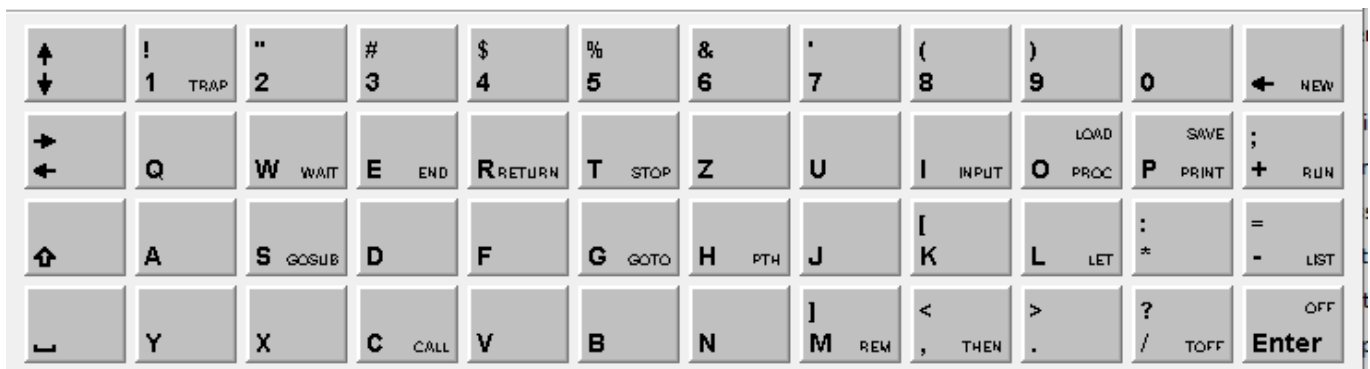
(Ausbaustufe I)

Das 2 KByte große Betriebssystem bietet die wichtigsten Funktionen zur Eingabe und Verwaltung von BASIC-Programmen. Des weiteren verwendet das Betriebssystem auch den im ROM des Einchipmikrorechners U883 enthaltenen BASIC-Interpreter.



## Tastatur

Beim 2K-Betriebssystem und beim EMR-ES 1988 besteht die originale Tastatur aus zwei Ebenen, die mit einer Shift-Taste umgeschaltet werden. Etwas ungewöhnlich ist jedoch, dass bei der Eingabe eines Zeichens aus der Shift-Ebene die Shift-Taste und die eigentliche Zeichentaste nicht zusammen sondern nacheinander zu drücken sind. Nach dem Drücken der Shift-Taste merkt sich das Betriebssystem den Shift-Modus und signalisiert diesen mit einem horizontalen Strich am unteren rechten Bildschirmrand. Nun drückt man die eigentliche Zeichentaste, woraufhin das Zeichen eingegeben und der Shift-Modus wieder verlassen wird.



oben rechts: CLR-Taste (clear, backspace, letztes Zeichen löschen)

## Kommandos

Nach dem Start bzw. nach einem RESET befindet sich das Betriebssystem im Kommandomodus. Das erkennt man an dem flimmernden Rechteck, unter dem sich der Buchstabe K verbirgt (K-Cursor). Ist der K-Cursor zu sehen, können Sie durch Drücken einer Taste (evtl. in Verbindung mit der Shift-Taste) folgende Kommandos eingeben, die auch sofort ausgeführt werden:

| Eingabe   | Kommando |  |
|-----------|----------|--|
| Backspace | NEW      | Löschen des BASIC-Programms, Achtung! Wenn Sie ein BASIC-Programm eingegeben haben und drücken die Backspace-Taste, wird das Programm ohne Nachfrage gelöscht. |
| - (Minus) | LIST     | Anzeige einer Zeile des BASIC-Programms, Durch nochmaliges Drücken der Minus-Taste wird die nächste Programmzeile angezeigt.                                   |

| Eingabe              | Kommando |   |
|----------------------|----------|---|
| + (Plus)             | RUN      | BASIC-Programm starten  |
| *                    | CONT     | BASIC-Programm nach der STOP-Anweisung fortsetzen   |
| Shift - O            | LOAD     | BASIC-Programm von Kassette laden. Achtung! Sie sollten das Kommando erst dann starten, wenn der Kennton zu hören ist.  |
| Shift - P            | SAVE     | BASIC-Programm auf Kassette speichern, Diese Kommando gibt das im Speicher befindliche BASIC-Programm auf dem Kassettenrekorderanschluss aus und führt danach, d.h. entsprechend der Länge des zu speichernden Programms nach einigen Sekunden bis einigen Minuten, eine Systeminitialisierung durch. |
| Shift - ENTER        | OFF      | Bisherige Eingabe löschen   |
| A...Z<br>(Buchstabe) |          | Anzeige des Inhalts der gleichnamigen BASIC-Variable  |
| 0...9 (Ziffer)       |          | Eingabe bzw. Löschen einer BASIC-Programmzeile  |

## Eingabe von BASIC-Programmzeilen

Möchten Sie eine BASIC-Programmzeile eingeben, müssen Sie folgendermaßen vorgehen:

1. Eingabe der Zeilennummer beim K-Cursor
2. ENTER drücken. Es erscheint der sogenannte A-Cursor (flimmerndes Rechteck über einem A)
3. Eingabe einer BASIC-Anweisung durch Drücken des entsprechenden Zeichens (siehe Tabelle)
4. Eingabe der evtl. notwendigen Argumente der BASIC-Anweisung
5. Eingabe evtl. weiterer BASIC-Anweisungen, die mit einem Semikolon zu trennen sind.
6. ENTER

Beim A-Cursor können Sie folgende BASIC-Anweisungen eingeben:

| BASIC-Anweisung | Einzugebendes Zeichen            |
|-----------------|----------------------------------|
| CALL            | C                                |
| ELSE            | > (größer gefolgt von Semikolon) |
| END             | E                                |
| GOSUB           | S                                |
| GOTO            | G                                |
| IF...THEN...    | F...,... (THEN: Komma)           |
| INPUT           | I                                |
| LET             | L                                |
| PRINT           | P                                |
| PTH (PRINTHEX)  | H                                |
| PROC            | O                                |
| RETURN          | R                                |
| REM             | M                                |
| STOP            | T                                |
| TOFF            | /                                |
| TRAP...TO...    | !...,...(TO: Komma)              |
| WAIT            | W                                |

Bei der Eingabe prüft der Editor die Syntax. Ist diese falsch, wird die Eingabe mit ERROR 0 abgebrochen. Achtung! Wenn der A-Cursor zu sehen ist, hat der Editor die Zeilennummer bereits in das BASIC-Programm eingefügt. Sie sollten nun auf keinen Fall RESET auslösen, denn dann bleibt die Zeilennummer ohne Programmzeile im BASIC-Programm stehen, was einen Syntaxfehler bedeutet. Das BASIC-Programm ist somit an dieser Stelle defekt. Sie können aber die störende Zeile einfach löschen und so das Programm wieder reparieren.

## Einschränkungen bei Zeilennummern

Das Betriebssystem verwendet ein Null-Byte als Endekennung für das BASIC-Programm. Aus diesem Grund darf kein Null-Byte im BASIC-Programm vorkommen, weder in Zeichenketten noch in Zeilennummern. Die Zeilennummern 0, 256, 512, 768, 1024 usw. enthalten jedoch in ihrem niederwertigen Byte eine Null. Aus diesem Grund wird bei der Eingabe einer solchen Nummer ihr Wert einfach um eins erhöht, d.h., aus 256 macht das Betriebssystem 257. Im höherwertigen Byte der Zeilennummer kann ein Null-Byte nicht vorkommen, da dort das Bit 7 immer gesetzt ist.

## Löschen von BASIC-Programmzeilen

Möchten Sie eine BASIC-Programmzeile löschen, drücken Sie nach Eingabe der Zeilennummer zweimal die ENTER-Taste, d.h., Sie lassen einfach die Anweisung weg. Zwar meldet der Editor ERROR 0, jedoch wird die Zeile trotzdem gelöscht.

From:

<https://hc-ddr.hucki.net/wiki/> - Homecomputer DDR

Permanent link:

<https://hc-ddr.hucki.net/wiki/doku.php/tiny/2k-system?rev=1659594651>

Last update: **2022/08/04 06:30**

