

Bel

Janich & Klass

Computersysteme

JK82 <E>PROM-PROGRAMMER
JK82 PAL-PROGRAMMER

ANWENDER-HANDBUCH

Ihr autorisierter Händler:

```

*****
*
*
*
*
*
*
*
**
**
*****

```

<C> 1985 by Janich & Klass Wuppertal

26.03.85

Inhaltsverzeichnis:

Allgemeine Bedienungshinweise	Seite 3
Teil A: jk82 (E)PROM-PROGRAMMER	
A1. Programmbeschreibung:	Seite 3
A1.1 Aufgabe und Voraussetzungen:	Seite 3
A1.2 Erstinstallation:	Seite 3
A1.3 Menuestruktur:	Seite 4
A1.4 Interaktive Eingaben:	Seite 5
A1.5 Statuszeile:	Seite 5
A1.6 Overlaymodule:	Seite 5
A1.7 Virtueller Puffer:	Seite 6
A1.8 Memory Map:	Seite 6
A1.9 Mitgelieferte Software:	Seite 7
A1.10 Besonderheiten beim jk82 Gang Modul:	Seite 7
A2. Programmabi auf:	Seite 8
A2.1 Start und Initialisierung:	Seite 8
A2.2 Modulmenue:	Seite 9
A2.3 Installation eines neuen Overlaxmoduls:	Seite 9
A2.4 Bausteinmenue:	Seite 10
A2.5 Funktionenmenue:	Seite 10
A3. Funktionen:	Seite 11
A3.1 Verifizieren:	Seite 11
A3.2 Lesen:	Seite 11
A3.3 Testen:	Seite 12
A3.4 Programmieren:	Seite 12
A3.5 Einlesen:	Seite 13
A3.6 Ausgeben:	Seite 13
A3.7 Modifizieren:	Seite 13
A3.8 Bildschirmausgabe:	Seite 14
A3.9 Druckerausgabe:	Seite 14
A4. Datenformat:	Seite 14
A5. Ausfallrate bipolarer PROMs:	Seite 14
Teil B: jk82 PAL-PROGRAMMER	
B1. Installation:	Seite 15
B2. Aufruf:	Seite 15
B3. PALAuswahl:	Seite 16
B4. PAL Verifizieren:	Seite 17
B5. PAL Lesen:	Seite 17
B6. PAL Programmieren:	Seite 18
B7. Datei Einlesen:	Seite 18
B8. Datei Ausgeben:	Seite 19
B9. Bildschirmausgabe:	Seite 19
B10. Druckerausgabe:	Seite 20
B11. PAL-TypWechseln:	Seite 20
B12. Abbruch:	Seite 20
B13. Datenformat:	Seite 20
B14. Ausfallrate:	Seite 22
B5. Programm ADJUST:	Seite 22

Allgemeine Bedienungshinweise

Bei dem Umgang mit EPROMs und PROMs sind die allgemeinen Vorschriften für den Umgang mit CMOS- bzw. NMOS-Bausteinen zu beachten. Insbesondere sind statische Entladungen über die Bauteilanschlüsse zu verhindern.

Das Einsetzen von Bausteinen, die weniger Anschlüsse besitzen, als der entsprechende Sockel, sind grundsätzlich unten bündig einzusetzen. Als "Unten" wird hier die dem Pin 1 gegenüberliegende schmale Kante des Gehäuses bezeichnet. Daraus ergibt sich, das die GND-Leitung für alle Bausteine an demselben Pin des Sockels sitzt.

Achtung: Alle Bausteine dürfen erst nach der Abfrage "(E)Prom eingesetzt ? (J/N) in die Fassung eingesetzt werden. Nachdem die Busy-LED wieder verloschen ist, muß der Baustein aus der Fassung genommen werden. Die Karte muß bei Unterschreiten der Betriebsspannung ein PWCLR-Signal erhalten. Wird dieses Signal nicht korrekt generiert, können noch in der Fassung befindliche Bausteine zerstört werden.

jk82 (E) PROM-PROGRAMMERA1. Programmbeschreibung:A1.1 Aufgabe und Voraussetzungen:

Das Programm PROGRAMM ermöglicht zusammen mit entsprechenden Overlaymodulen die Programmierung von EPROMs und bipolaren PROMs bis zu einer maximalen Größe von 64KByte.

Das Programm läuft unter den Betriebssystemen CP/M 2.2, 2DOS und CP/M PLUS und benötigt mindestens 48KByte TPA bzw. freies RAM. Die zugehörige Platine jk82 Universal-Programmer setzt ein ECB-kompatibles System voraus.

Da das Programm nur die ASCII-Steuerzeichen CARRIAGE RETURN, LINE FEED und BELL benutzt, kann eine Terminal-Anpassung entfallen. Vorausgesetzt wird ein Format von mind. 24 Zeilen a 80 Zeichen.

A1.2 Erstinstallation:

Zuerst müssen Sie mindestens eine Kopie der Diskette herstellen. Die Original diskette ist schreibgeschützt. Sie dürfen auf die Originaldiskette keinesfalls schreiben. Eine modifizierte Originaldiskette wird von uns wie eine kostenpflichtige Reparatur behandelt.

Auf der Diskette befindet sich das Programm **PROGRAMM.COM** mit der zugehörigen **Menuedatei MENUE.PGM**. Diese **zwei** Dateien sind unbedingt **erforderlich**.

Außerdem befinden sich noch eine Anzahl Dateien mit der Extension **".PGM"** auf der Diskette (im Auslieferungszustand **INTEPRO.PGM, GANGMOD.PGM, TEXASPRO.PGM**). Diese Dateien enthalten die hardware-abhängigen Module.

Alle Dateien mit der Extension **".PGM"** müssen sich auf einer Diskette befinden. Falls Sie ein neues **Overlaymodul** (**VALVOPRO.PGM, HARRIPRO.PGM etc.**) erworben haben, kopieren Sie die Datei auf Ihre **Arbeitsdiskette** und tragen Sie den Namen und den Kennbuchstaben in die Datei **MENUE.PGM** ein (**siehe** unter: **Installation neuer Overlay-module**).

A1.3 Menuestruktur:

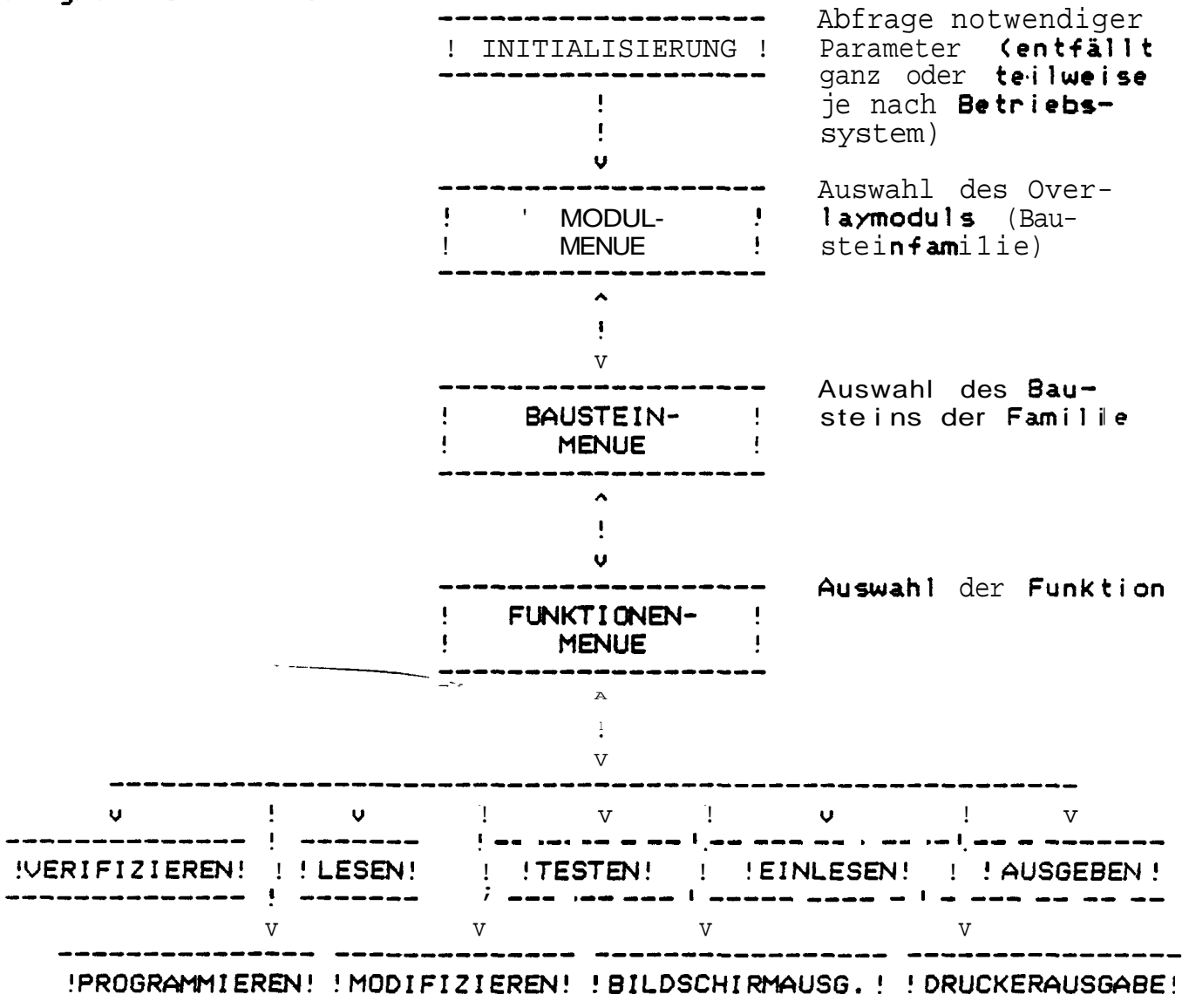
Um die **Bedienung** des Programmes **möglichst einfach** zu **gestalten**, **erfolgt die Auswahl** von Modulen, Bausteinen und Funktionen **menuegesteuert**.

Immer wenn Sie eine Auswahl **treffen** müssen, erscheint die Meldung: (<=, x, =>) Bitte waehlen Sie:

Zusätzlich zu den im Menue **ausgewiesenen** Kennbuchstaben können Sie die Zeichen <, X und > angeben. Diese Zeichen haben unterschiedliche, wenn auch ähnliche Bedeutung, je nachdem ob sie auf Menueebene oder innerhalb einer Funktion (**nur** Modifizieren und **Bildschirmausgabe**) eingegeben werden.

Die Eingabe von < bringt Sie in das vorherige Menue zurück bzw. auf die vorherige Adresse (**Modifizieren**) oder **Seite** (**Bildschirmausgabe**), die Eingabe von > bringt Sie in **sinnvollen Fällen** in **das nächste** Menue bzw. auf die nächste Adresse oder **Seite**. Die Eingabe von X beendet das Programm oder die Funktion.

Programmablaufstruktur:



A1.4 Interaktive Eingaben:

Um die Bedienung an den Stellen zu vereinfachen, an denen interaktive Eingaben gemacht werden müssen (z.B. bei Dateinamen), werden Defaultwerte vorgegeben, wenn dies sinnvoll möglich ist.

Durch einfache Eingabe von CARRIAGE RETURN kann der Defaultwert akzeptiert werden. Unter CP/M Plus wird der Defaultwert direkt in den Inputpuffer übertragen, so daß er nicht nur akzeptiert, sondern über die CP/M Plus Editierfunktionen auch modifiziert werden kann.

A1.5 Statuszeile:

Die oberste Zeile jedes Menues ist die Statuszeile. Sie dient in erster Linie als Gedächtnisstütze und setzt sich wie folgt zusammen:

Modulname, Baustein, Dateiname, (M), Adreßraum, 4/8 Bit-Typ, Systemfrequenz, Datum

Der Dateiname erscheint nur, falls eine Datei eingelesen wurde. Wird ein (E)Prom eingelesen ist der Dateiname "(E)Prom".

Haben Sie mit der Funktion Modifizieren den Pufferinhalt verändert, wird ein M eingetragen.

Beispiel:

TEXASPRO TBP24S(A)10 C:PROG.COM M 00256-BYTE 4-BIT 6-MHZ 28.12.84

A1.6 Overlaymodule:

Das Programm arbeitet mit Overlays, die die Programmiervorschriften und Anschlußbelegungen jeweils einer Bausteinfamilie beinhalten. Neben den bereits im Lieferumfang enthaltenen Overlaymodulen können weitere Module erworben werden. Das aktuelle Angebot entnehmen Sie bitte der jeweils gültigen Preisliste.

Jedes Overlay bildet eine eigene Datei, wobei alle Overlaydateien die Extension .PGM haben.

Die Namen der implementierten Overlaymodule sind in der Menuedatei MENUE.PGM festgehalten. Haben Sie ein neues Modul erworben, so muß es in die Menuedatei aufgenommen werden (siehe unter: Installation neuer Overlaymodule).

Alle Overlaydateien müssen zusammen mit der Programmdatei PROGRAMM.COM und der Menuedatei unter derselben Useummer auf demselben Laufwerk liegen.

A1.7 Virtueller Puffer:

Um Bausteine mit einer Kapazität von bis zu 64KByte programmieren zu können, wird der dafür benötigte Datenpuffer virtuell verwaltet. D.h. nur ein Teil des Puffers ist speicherresident, der Rest wird auf einen externen Datenträger ausgelagert.

Die Verwaltung dieses Puffers geschieht vollautomatisch und ist für den Anwender transparent. Es ist lediglich dafür zu sorgen, daß auf einem der angeschlossenen Laufwerke eine freie Speicherkapazität von mindestens 64KByte vorhanden ist.

Zur Realisierung des virtuellen Puffers wird vom Programm eine Scratch-Datei unter dem Namen EPROMMER.*** angelegt. Sollte auf dem dafür vorgesehenen Laufwerk (wird während der Initialisierung festgelegt) nicht genügend Platz sein, so fragt das Programm nach einem anderen Laufwerk. Die Datei wird erst dann angelegt, wenn die Kapazität des angesprochenen Bausteins tatsächlich die Größe des speicherresidenten Teils des Puffers überschreitet, so daß kleinere Bausteine voll speicherresident verarbeitet werden können. Bei ordnungsgemäßem Abschluß des Programms wird die Datei automatisch gelöscht.

Wie groß der speicherresidente Teil des Puffers ist, ist abhängig von der Größe der TPA (Transient Program Area) sowie der Größe des verwendeten Overlaymoduls. Dieser Wert wird dynamisch ermittelt und kann 4, 8 oder 16KByte betragen.

A1.8 Memory Map:

0000	-----
	! Zero Page !
0100	-----
	! Programmcode und Daten !

	! Lokale Variablen und Stack !
8000	-----
	! Overlaycode und Daten !

	! residenter Teil des Puffers !

	! Betriebssystem !
FFFF	-----

A1.9 Mitgelieferte Software:

INTEPRO 1.01 programmiert EPROMs sowie INTEL-Typen im "Fast Mode"

2516	2564	27128(INTEL)
2716	2764(21V)	27128A(INTEL)
2532	2764(INTEL)	27C128
2732	2764A(INTEL)	27256(21V)
2732A(21V)	27C64	27256(INTEL)
27C32	27128(21V)	27C256

TEXASPRO 1.01 programmiert PROMs von TEXAS INSTRUMENTS

TBP24S(A)10	TBP24S(A)81	TBP28(S/L/P)85
TBP28L(A)22	TBP28(S/L/P)45	TBP28S(A)166
TBP28S(A)42	TBP28S(A)46	TBP28(L/P)166
TBP28(L/P)42	TBP28S(A)86	
TBP24S(A)41	TBP28L86	

GANGMOD 1.01 programmiert bis zu 8 EPROMs mit dem Jk82 Gang Modul

2516	2764(21V)	27C128
2716	2764A(12V)	27256(21V)
2732	27C64	27256(12V)
2732A(21V)	27128(21V)	27C256
27C32	27128A(12V)	

(alle mit einer Programmierzeit von 25ms oder 50ms)

A1.10 Besonderheiten beim jk82 Gang Modul:

Das jk82 Gang Modul programmiert bis zu 8 EPROMs gleichzeitig. Dazu wird das **Overlay GANGMOD** ausgewählt. Einige Fehlermeldungen sind bei Benutzung des Gang Modul etwas anders zu interpretieren.

Das Lesen eines EPROMs kann nur im Sockel 1 erfolgen. In den Sockeln 2 - 8 müssen entweder **unprogrammierte EPROMs sitzen** oder sie müssen **leer sein**. Ansonsten erscheint eine **Fehlermeldung** und alle roten LEDs leuchten **auf**.

Beim **Verifizieren** und **Programmieren** werden alle korrekt **verifizierten** EPROMs durch **Aufleuchten** der grünen LED, die **fehlerhaften** durch **Aufleuchten** der roten LED angezeigt. Das Programm meldet nur dann **einen Fehler**, wenn **alle** EPROMs mindestens einen Fehler **aufweisen**. Der **fehlerhafte** Wert wird immer mit 0 ausgegeben, unabhängig von dem **tatsächlichen** Fehler.

Die Funktion **Test** liefert nur ein korrektes Ergebnis, wenn alle EPROMs **leer** sind. Ist nur ein EPROM nicht **leer**, leuchten **alle** roten LEDs **auf**.

A2. Programmablauf:

A2.1 Start und Initialisierung:

Das Programm ist durch Eingabe des Namens ohne Extension oder Parameter aufzurufen.

Für eine ordnungsgemäße Funktion benötigt das Programm das aktuelle Tagesdatum, die Namen der Laufwerke, auf denen die Overlays zu finden sind und der virtuelle Puffer angelegt werden soll, sowie den tatsächlichen Systemtakt. Diese Werte werden in einer Initialisierungsphase unmittelbar nach dem Programmstart ermittelt.

Wie sich diese Initialisierungsphase für den Benutzer darstellt, ist abhängig von dem eingesetzten Betriebssystem:

Unter ZDOS und CP/M 2.2 werden alle Angaben interaktiv abgefragt; unter CP/M Plus muß lediglich der Systemtakt angegeben werden und falls Sie über die CP/M Plus Implementation der Firma Janich & Klass mit einer CBIOS Version 1.5 oder größer verfügen, so wird auch dieser Wert automatisch ermittelt.

Datum: ('00.00.84') _

Geben Sie das Datum in der Form (TT.MM.JJ) an.

Unter CP/M Plus wird das Datum über die Datumsfunktion des Betriebssystems ermittelt.

Laufwerk fuer Overlays: ('') _

Geben Sie hier den Namen des Laufwerks an, auf dem die Overlaydateien liegen. Unter CP/M Plus wird als Laufwerk für die Overlays das Laufwerk angenommen, von dem das Programm PROGRAMM geladen wurde.

Laufwerk fuer virt. Puffer: ('') _

Geben Sie hier den Namen des Laufwerks an, auf dem der virtuelle Puffer angelegt werden soll. Falls das Anlegen des Puffers scheitert, weil die Directory dieses Laufwerks voll oder nicht wenigstens 64KByte freier Speicherplatz vorhanden ist, so fragt das Programm nach einem anderen Laufwerk. Unter CP/M Plus wird als Laufwerk für den virtuellen Puffer das "temporary drive" angenommen. Das "temporary drive" ist in der CP/M PLUS Implementation von Janich & Klass entweder die "RAM-disk" oder das "default drive".

Maschinentakt 4, 3, 6 MHz: ('') _

Die Eingabe ist zur Einhaltung der Programmiersvorschrift unbedingt erforderlich. Unter dem CP/M PLUS von Janich & Klass wird der Systemtakt automatisch über das CBIOS ermittelt.

Die hier beschriebenen Programme sind nur bei Systemtaktfrequenzen von 4MHz, 5MHz oder 6MHz lauffähig.

A2.2 Modulmenue:

Nach der Initialisierungsphase wird die Datei **MENUE.PGM** vom Overlaylaufwerk **gelesen** und das **Modulmenue** auf den **Bildschirm** ausgegeben. In der **Auslieferversion** erscheint folgendes Menue:

```

      <A>      INTEPRO      :      <B>      TEXASPRO
      <C>      GANGMOD     :      < >

```

Waehlen Sie < zum **Implementieren** eines neuen Moduls
(<=, x, =) Bitte **waehlen** Sie:

Nun kann durch Eingabe des **Kennbuchstabens** eine **Bausteinfamilie** **ausgewaehlt** werden.

A2.3 Installation eines neuen Overlaymoduls:

Falls Sie ein neues **Modul** erworben haben, kopieren Sie die **Overlaydatei** zu den anderen **Dateien** und waehlen Sie im **Modulmenue** "<". Es erscheint die Meldung:

Name der neuen **Moduldatei** <ohne '.PGM'>: (<'>)

Geben Sie den Namen (z.B. **HARRIPRO**, **VALVOPRO**) ohne Extension ein. Danach wird **gefragt**:

Kennbuchstabe dieses Moduls:

Dann erscheint die Frage:

Eintrag speichern (J/N)

Falls Sie hier mit **N** antworten, koennen Sie das eingetragene **Overlaymodul** zwar starten, jedoch ist Ihr neuer Eintrag bereits beim naechsten Anzeigen des Modulmenues nicht mehr vorhanden.

Achten Sie **darauf**, daB Sie keinen **Kennbuchstaben** doppelt vergeben. Eintraege in der Datei **MENUE.PGM** koennen weder **geloescht** noch **veraendert** werden. Es kann lediglich die gesamte Datei **MENUE.PGM** **geloescht** werden.

Wenn das Programm beim naechsten Aufruf die Datei nicht auf dem **Overlaylaufwerk** findet, legt es eine leere Datei **MENUE.PGM** an und **meldet**:

```

Datei MENUE.PGM nicht gefunden
Fehler! System nicht initialisiert

```

Sie muessen nun nach dem oben **beschriebenen Verfahren** alle **Overlaynamen** und **Kennbuchstaben**, auch die **bereits** im Lieferumfang enthaltenen, neu eintragen.

A2.4 Bausteinmenue:

Haben Sie beispielsweise das Modul TEXASPRO ausgewählt, erscheint folgendes Menue:

TEXASPRO 00000-BYTE 0-BIT 6MHZ 22.01.85

Texas Instruments PROMS : Version 1.01 : 14.01.85 : <C>85 T.Ingendoh

	Typ	Organisation	Ausgaenge
(A)	TBP24S(A)10	256 X 4	T.S./O.C.
(B)	TBP28L(A)22	256 X 8	T.S./O.C.
(C)	TBP28S(A)42	512 X 8	T.S./O.C.
(D)	TBP28(L/P)42	512 X 8	T.S./O.C.
(E)	TBP24S(A)41	1024 X 4	T.S./O.C.
(F)	TBP24S(A)81	2048 X 4	T.S./O.C.
(G)	TBP28(S/L/P)45	512 X 8	T.S.
(H)	TBP28S(A)46	512 X 8	T.S./O.C.
(I)	TBP28S(A)86	1024 X 8	T.S./O.C.
(J)	TBP28L86	1024 X 8	T.S.
(K)	TBP28(S/L/P)85	1024 X 8	T.S.
(L)	TBP28S(A)166	2048 X 8	T.S./O.C.
(M)	TBP28(L/P)166	2048 X 8	T.S.

(<=, x, =>) Bitte waehlen Sie:

Waehlen Sie nun den entsprechenden Baustein durch Eingabe des Kennbuchstabens aus. Anschließend gelangen Sie in das Haupt- oder Funktionenmenue.

A2.5 Funktionenmenue:

TEXASPRO TBP24S(A)10 00256-BYTE 4-BIT 6-MHZ 28.12.84

(V)	Verifizieren	(E)Prom = Puffer
(L)	Lesen	(E)Prom --> Puffer
(T)	Testen	(E)Prom <-- Puffer
(P)	Programmieren	(E)Prom <-- Puffer
(E)	Einlesen	Datei --> Puffer
(A)	Ausgeben	Datei <-- Puffer
(M)	Modifizieren	Tastatur --> Puffer
(B)	Bildschirmausgabe	Bildsch. <-- Puffer
(D)	Druckerausgabe	Drucker <-- Puffer

Die gewünschte Funktion wird durch Eingabe des entsprechenden Kennbuchstaben ausgewählt. Die Pfeile geben die Richtung der Operation an, z.B. programmiert die Funktion <P> das (E)Prom (<E)Prom <-- Puffer).

A3. Funktionen:**A3.1 Verifizieren:**

Die Funktion **vergleicht** den Pufferinhalt mit dem **(E)Prom** bei über- und Unterspannung. Besitzen **beide** denselben Inhalt, erscheint die **Meldung:**

Kein Fehler beim **Verifizieren**.
Druecken Sie irgendeine Taste

Tritt ein Fehler auf, wird dies durch Ausgabe eines **"Bell-Zeichens"**, der **Fehleradresse**, des **Pufferinhalts** und des **(E)Prom-Inhalts** ausgewiesen:

Fehler beim **Verifizieren!** Adresse: 0000 Puffer : 00 (E)Prom: FF
Abbrechen **(J/N)**

Die Eingabe von **J** führt zurück ins **Hauptmenue**, die Eingabe von **N** führt die Funktion bis zum nächsten Fehler oder bis zum **Ende weiter**.

Werden **4-Bit Typen verifiziert**, wird das obere **Digit** des Puffers ignoriert. Es kann beliebige **Werte** annehmen.

A3.2 Lesen:

Die Funktion kopiert den **Inhalt** eines **(E)Proms** in den Puffer. Die korrekte Ausführung wird mit folgender Meldung quittiert: •

(E)Prom fehlerfrei gelesen.
Druecken Sie irgendeine Taste

Werden 4Bit-Typen eingelesen, so wird das obere Digit mit 0 oder F angenommen, je nachdem welcher Wert den **unprogrammierten** Zustand darstellt.

Nur beim **Einlesen eines EPROMs** mit dem **jk82 Gang Modul** kann die folgende **Fehlermeldung** auftreten:

Fehler beim Lesen!
Druecken Sie irgendeine Taste

Diese **Fehlermeldung besagt**, daß entweder mehrere EPROMs in dem Gang Modul stecken oder das zu lesende **EPROM** nicht im Sockel **1** sitzt.

A3.3 Testen:

Die Funktion vergleicht den (E)Prom-Inhalt mit dem Pufferinhalt unter dem Gesichtspunkt, ob abweichende Bits nachprogrammierbar sind oder nicht. Da einige Bausteine eine unprogrammierte Sicherung durch eine 0, andere durch eine 1 anzeigen, ist das Resultat von dem jeweiligen Baustein abhängig. Die Funktion berücksichtigt dies. Folgende Meldungen können auftreten:

(E)Prom ist leer.

Druecken Sie irgendeine Taste

(E)Prom ist nachprogrammierbar.

Druecken Sie irgendeine Taste

Diese Meldung besagt, daß das (E)Prom zwar teilweise programmiert ist, aber nachprogrammiert werden kann.

Test Fehler! Adresse: 0000 Puffer: 00 (E)Prom: FF

Fehler! (E)Prom nicht nachprogrammierbar!!!

Druecken Sie irgendeine Taste

Fehlermeldung mit Angabe der Adresse, des Puffer- und des (E)Prom-Inhalts.

Bei Benutzung des jk82 Gang Modul liefert diese Funktion nur ein korrektes Ergebnis, wenn alle EPROMs leer sind. Ist nur ein EPROM nicht leer, leuchten alle roten LEDs auf.

A3.4 Programmieren:

Die Funktion programmiert den Pufferinhalt in das (E)Prom. Folgende Meldungen können auftreten:

Kein Fehler beim Programmieren.

Kein Fehler beim Verifizieren.

Druecken Sie irgendeine Taste

Der Baustein ist korrekt programmiert und bei über- und Unter- spannung funktionsfähig.

Kein Fehler beim Programmieren.

Fehler beim Verifizieren! Adresse: 0000 Puffer: 00 (E)Prom: FF

Abbrechen (J/N)

Der Programmiervorgang ist korrekt abgelaufen, jedoch der nachfolgende Verifizierdurchgang hat einen Fehler gefunden.

Schreib Fehler! Adresse: 0000 Puffer: 00 (E)Prom: FF

Fehler! beim Programmieren

Druecken Sie irgendeine Taste

Durch Aufruf der Funktion "Test" können Sie überprüfen, ob der Baustein nachprogrammierbar ist oder nicht. Das Nachprogrammieren eines einmal als defekt erkannten Bausteins entspricht allerdings nicht den Herstellerspezifikationen.

A3.5 Einlesen:

Die Funktion **lädt** eine Datei in den **virtuellen** Puffer und **fragt** daher zuerst nach dem Dateinamen.

Name der zu **lesenden Datei:**

Der Name kann **mit** oder ohne Laufwerk angegeben werden. **Fehlt** die **Laufwerksbezeichnung**, so wird auf dem "default drive" gearbeitet. Ist die Datei auf dem Laufwerk nicht vorhanden erscheint die **Fehlermeldung:**

Fehler! Datei nicht gefunden.
Druecken Sie irgendeine Taste

Haben Sie in dem laufenden **Programm** **bereits** einmal eine Datei gelesen oder **geschrieben**, so wird deren Dateiname als **Default** vorgegeben (**siehe** unter: **Interaktive Eingaben**).

A3.6 Ausgeben:

Die Funktion schreibt den Inhalt des virtuellen Puffer auf **eine** Datei und fragt daher zuerst nach dem Dateinamen

Name der zu schreibenden **Datei:**

Es gelten dieselben Bedingungen wie **bei** der Funktion Einlesen. Ist die Datei auf dem Laufwerk bereits vorhanden, erscheint die **Fehlermeldung:**

Datei existiert! **Ueberschreiben (J/N)**

A3.7 Modifizieren:

Die Funktion modifiziert den Puffer. Zuerst wird nach **der** Startadresse im HEX-Format gefragt. Durch Eingabe von CR wird die Startadresse zu 0000 angenommen. Die Ausgabe hat folgendes Format:

0000: FF '.'

Adresse: **Pufferinhalt** 'ASCII-Wert'

Der ~~neue Wert~~ wird im HEX-Format eingegeben. **Dabei** steht das Zeichen '.' für **einen HEX-Wert**, der sich nicht **als ASCII-Wert** darstellen **läßt**.

Die Zeichen > und < können benutzt werden, um zu der nächsten bzw. vorherigen Adresse zu gelangen. Die Eingabe von X beendet die **Funktion**.

A3.8 Bildschirmausgabe:

Nach der **Abfrage** der Startadresse erfolgt die **Bildschirmausgabe** des Inhalts von jeweils **256** Adressen. Der **ASCII-Wert** wird hinter jeder **Zeile** ausgegeben. Dabei steht das Zeichen **'.'** für einen **HEX-Wert**, der sich nicht als **ASCII-Wert** darstellen läßt.

Die Zeichen **>** und **<** können benutzt werden, um zu der nächsten bzw. **vorherigen** Seite zu gelangen. Die Eingabe von **X** beendet die Funktion.

A3.9 Druckerausgabe:

Nach **Abfrage** der Startadresse beginnt die Ausgabe von je **32 Zeilen mit Programmversion**, Datum, Seitennummer und Statuszeile. Die Ausgabe des **Pufferinhalts** auf den Drucker kann durch Eingabe eines beliebigen Zeichens unterbrochen werden.

A4. Datenformat:

Alle Dateien werden nur im **Binär-Format** gelesen und geschrieben. Eine mit **einem** Assembler erzeugte **COM-Datei** kann so direkt in ein **(E)PROM** programmiert werden.

A5. Ausfallrate bipolarer PROMs:

Ein **unprogrammiertes** PROM ist kaum durch den Hersteller zu testen. Durch das **Programmieren** wird praktisch ein neuer Baustein **gefertigt**. Daher ist es **außerordentlich** wichtig, das PROM nach dem **Programmieren bei über- und Unterspannung** zu **verifizieren**. Das **vorliegende** Programm zusammen mit der Platine **TIJ-2-1292** hält sich an die **Programmiersvorschrift** des Herstellers. Trotzdem müssen Sie mit einer **Ausfallrate** von **durchschnittlich 2 - 3%** rechnen. Der Wert streut von **0%** bis **5X** pro Charge und ist abhängig von der Anzahl der zu **programmierenden** Sicherungen.

über **Ausfallraten** sollten Sie buchführen. **Falls** Ihre **Programmiersstation** **plötzlich** stark erhöhte Ausfallraten produziert, schicken Sie die Platine unbedingt sofort zur Reparatur ein. Für fehlerhaft **programmierte** Bausteine können wir keine Haftung übernehmen.

JK82 PAL-PROGRAMMER

B1. Installation:

Das Programm MMIPAL (TIPAL etc.) läuft unter den Betriebssystemen CP/M 2.2, ZDOS und CP/M PLUS und benötigt mindestens 48K8xte TPA bzw. freies RAM. Die zugehörige Platine jk82 Universal-Programmer setzt ein ECB-kompatibles System voraus.

Die Platine muß mit einer externen Spannung von 30V bei mind. 500mA versorgt werden. Dazu wird das von uns mitgelieferte Festspannungsnetzteil Jk82 V30 oder ein anderes Netzteil benötigt. Die Spannung wird vor jedem Aufruf eines Me'nues überprüft.

Zuerst müssen Sie mindestens eine Kopie der Diskette herstellen. Die Originaldiskette ist schreibgeschützt. Sie dürfen auf die Originaldiskette keinesfalls schreiben. Eine modifizierte Originaldiskette wird von uns wie eine kostenpflichtige Reparatur behandelt.

Auf der Diskette befindet sich das Programm MMIPAL mit den zugehörigen Overlays PAL20.OVR und PAL24.OVR. Diese drei Programme sind unbedingt erforderlich.

Außerdem befinden sich noch eine Anzahl Dateien mit der Extension "PLO" auf der Diskette. Diese Dateien entsprechen dem Inhalt eines unprogrammierten PALs. Falls nicht mit einem PAL-Assembler gearbeitet wird, können diese Dateien direkt mit einem Texteditor bearbeitet und anschließend programmiert werden.

B2. Aufruf:

Das Programm ist durch Eingabe des Namens ohne Extension oder Parameter aufzurufen. Da das Programm mit Overlays arbeitet und diese auf dem Default-Laufwerk sucht, müssen Sie vorher durch Eingabe von z.B. B: oder C: auf das Laufwerk umschalten, auf dem sich das Programm und die Overlays befinden.

B3. PAL Auswahl:

Nach Aufruf des Programms erscheint die unten abgebildete Liste unter der Voraussetzung, dass die 30V-Versorgung anliegt. Falls nicht, so erscheint die Fehlermeldung:

Die 30Volt-Versorgung fehlt!

solange, bis der Fehler behoben ist. Die Fehlermeldung kann auch auf eine durchgebrannte Sicherung auf der Platine hinweisen.

Es erscheint das folgende Menue:

***** PALs der Firma NS sind kompatibel zu MMI-PALs *****

Auswahl des PAL-Typs: (1), (2) oder (3)

(1)	(2)	(3)
MMI PAL10H8	MMI PAL10L8	MMI PAL12L10
MMI PAL12H6	MMI PAL12L6	MMI PAL14L8
MMI PAL14H4	MMI PAL14L4	MMI PAL16L6
MMI PAL16H2	MMI PAL16L2	MMI PAL18L4
	MMI PAL16R4	MMI PAL20L2
	MMI PAL16X4	MMI PAL20X4
	MMI PAL16A4	MMI PAL20X8
	MMI PAL16R6	MMI PAL20L10
	MMI PAL16L8	MMI PAL20X10
	MMI PAL16R8	MMI PAL20R4
		MMI PAL20R6
		MMI PAL20R8
		MMI PAL20L8

Auswahl:

Dieses Menue enthält alle mit dem entsprechenden Programm (MMIPAL, TIPAL etc.) programmierbaren Typen.

Durch Eingabe der Ziffer 1, 2 oder 3 wird die entsprechende PAL-Familie ausgewählt.

Jetzt erscheint das folgende Menue:

MMIPAL	Version 1.01	(MMI-PALS)	21.01.85	(C)85T.Ingendoh
(V)	Verifizieren		PAL =	Puffer
(L)	Lesen		PAL -->	Puffer
(P)	Programmieren		PAL <--	Puffer
(E)	Einlesen		Datei -->	Puffer
(A)	Ausgeben		Datei <--	Puffer
(B)	Bildschirmausgabe		Bildsch. <--	Puffer
(D)	Druckerausgabe		Drucker <--	Puffer
(W)	Wechsel des PAL-Typ			
(X)	Zurueck zum System			

Bitte waehlen Sie:

Die **gewuenschte** Funktion wird **ausgewaehlt**. Nach Beendigung der Funktion erscheint das **Haupt-Menue** wieder (Ausnahme W und X). **Vorher** wird die **30V-Versorgung** noch **ueberprueft**. Einige Funktionen benoetigen **relativ viel** Zeit (**besonders P, L und V**). In diesen Faellen erscheint die Meldung **### Bitte Warten *****.

B4. PAL Verifizieren:

(V) Verifizieren PAL = Puffer

Die Datei **muß vorher** in den **Puffer** geladen werden (L oder E). Das PAL wird **vollstaendig bei** ueber- und **Unterspannung** mit der Datei verglichen. Folgende Meldungen koennen **auftreten**:

Korrekt **verifiziert** ! (CR)
oder
Fehler beim **Verifizieren** gefunden ! (CR)

Nach **beiden** Meldungen kommen Sie mit CR zurueck ins **Hauptmenue**.

B5. PAL Lesen:

(L) Lesen PAL --> Puffer

Das PAL wird in den Puffer kopiert. Dazu **laedt** das Programm die Datei **PAL24.OVR** oder **PAL20.OVR**. In diese Datei wird der **Inhalt** des PAL eingetragen. Der Puffer kann nun **mit** neuem Namen auf eine Diskette geschrieben werden. Der Kopf, ausgedruckt in der obersten **Zeile** auf dem **Bildschirm**, kann nachher durch ein **Textverarbeitungsprogramm** aktualisiert werden.

Folgende **Fehlermeldungen** koennen auftreten:

Dateiname existiert nicht ! (CR)
Datei **"PAL24.OVR"** nicht gefunden ! (CR)
oder
Datei **"PAL20.OVR"** nicht gefunden ! (CR)

Ein PAL kann nur in die **entsprechende Overlay-Datei** kopiert werden. Aendern Sie diese **zwei** Dateien nie.

B6. PAL Programmieren:

<P) Programmieren PAL <-- Puffer

Da< PAL wird mit der Datei programmiert. Es gelten dieselben Voraussetzungen wie beim Verifizieren. Folgende Meldungen treten auf:

Korrekt programmiert !
oder
Fehler beim Programmieren gefunden !

Die Meldung: Korrekt programmiert ! besagt lediglich, daß alle zu programmierenden Sicherungen erfolgreich programmiert wurden. Nicht zu programmierende Positionen werden erst bei anschließendem Aufruf der Verify-Funktion überprüft. Daraus folgt, daß ein fehlerfrei programmiertes PAL, das trotzdem einen Fehler beim Verifizieren zeigt, eine zuviel programmierte *Fuse" besitzt. Dieser Zustand ist irreversibel.

Erscheint die Meldung: Fehler beim Programmieren gefunden ! besteht noch die Möglichkeit durch einen erneuten Versuch die bisher nicht programmierte "Fuse" zu programmieren. Diese Prozedur entspricht allerdings nicht den Herstellerspezifikationen, nach denen ein Bauteil als defekt auszusondern ist, wenn der erste Programmierungsversuch erfolglos war.

Die Funktion ruft automatisch die Verify-Funktion auf. Ist das PAL korrekt programmiert erscheint die Meldung:

PAL sichern ? (J/N)

Das PAL wird nur durch die Eingabe von "J" gesichert, jede andere Eingabe führt zurück ins Hauptmenue. Ein gesichertes PAL ist nicht mehr les- oder programmierbar. Dies wird durch erneuten automatischen Aufruf der Verify-Funktion überprüft. Ein Verify-Fehler produziert die Meldung:

PAL ist Kopiergeschützt ! <CR)

Die Programmierung ist beendet. Erscheint die Meldung:

PAL sichern war erfolglos ! <CR)

so war der Versuch erfolglos. Das PAL ist zwar voll funktionsfähig aber nicht Kopiergeschützt. Dieser Fehler tritt erfahrungsgemäß sehr selten auf, so daß Sie in diesem Fall das Bauteil aussondern sollten.

B7. Datei Einlesen:

<E> Einlesen Datei --> Puffer

Eine Datei wird von der Diskette eingelesen. Zuerst wird nach dem Laufwerk gefragt. Das Laufwerk <A - P) ist explizit anzugeben. Der Dateiname ist ebenfalls vollständig im Format NAME.PLO anzugeben. Eine Eingabe von CR als erstes Zeichen führt zurück ins Hauptmenue.

Es können nur Dateien eingelesen werden, die dem später beschriebenen Format entsprechen.

Folgende **Fehlermeldungen** können auftreten:

Falsches Laufwerk ! <CR)
Dateiname existiert nicht ! (CR)

Nach beiden Fehlern kommen Sie **mit** CR zurück ins Hauptmenue.

Falls die einzulesende Datei nicht der **Definition** (s. Datenformat) entspricht, tritt eine weitere **Fehlermeldung** auf:

Dateiname **korrespondiert** nicht mit dem **gewählten** PAL-Typ ! <CR)

Der Dateiname bzw. der PAL-Typ ist in der **ausgewählten** Gruppe (1, 2 oder 3) nicht enthalten. Rufen Sie die Funktion U **auf**.

Nach diesem Fehler kommen Sie **mit** CR zurück ins Hauptmenue.

B8. Datei Ausgeben:

(A) Ausgeben Datei <-- Puffer

Es **gelten** die gleichen **Voraussetzungen** und **Fehlermeldungen** wie beim Einlesen einer **Datei**. Sie **sollten** das Format **NAME.PLO** benutzen. **Zwei zusätzliche Fehlermeldungen** können auftreten:

Inhaltsverzeichnis voll ! (CR)
 Diskette **voll ! (CR)**

Nach beiden Fehlern kommen Sie **mit** CR **zurück** ins Hauptmenue.

Falls die Datei **auf** dem **gewählten** Laufwerk **bereits** existiert, **erscheint die Meldung:**

Dateiname existiert, **ueberschreiben ? (J/N)**

Nur die Eingabe von **"J"** schreibt die Datei **auf** die Diskette. Jede andere Eingabe **führt** zurück ins Hauptmenue.

B9. Bildschirmausgabe:

(B) **Bildschirmausgabe** **Bildsch.** <-- Puffer

Die Datei aus dem RAM-Puffer wird **auf** dem **Bildschirm** **ausgegeben**. Dazu **muß** die Datei entweder von der Diskette eingelesen oder durch die Funktion (L) **erstellt** worden **sein**.

Der Ausdruck beginnt **mit** der Meldung:

*** Jedes eingegebene Zeichen stoppt den Ausdruck ***

d.h. Sie können den Ausdruck durch Eingabe eines beliebigen Zeichens anhalten. Ein zweites Zeichen setzt den Ausdruck fort. Ein Abbruch der Funktion **ist** nicht **möglich**. Nach dem Ende des Ausdrucks kommen Sie mit der Eingabe von CR wieder ins Hauptmenue. Ist keine Datei im Puffer wird auch nichts ausgegeben. In diesem **Fall** kommen Sie ebenfalls **mit** CR ins Hauptmenue zurück.

B10. Druckerausgabe:

(D) Druckerausgabe Drucker <-- Puffer

Die Datei wird auf dem Drucker ausgegeben. Es gelten **dieselben Voraussetzungen** wie bei der **BildschirmAusgabe**. Der Ausdruck ist jedoch nicht **unterbrechbar**.

Die Funktion setzt einen Drucker **mit 80 Zeichen** und **mind. 60 Zeilen pro Seite** voraus. Nach **55 Zeilen** wird ein **Seitenwechsel** ausgegeben.

B11. PAL-Typ Wechseln:

(W) Wechsel des PAL-Typ

Diese Funktion bringt Sie zurück in das **Auswahlmenue**. Sie **muß** nur dann aufgerufen werden, wenn das neue PAL aus einer anderen **Gruppe** (1, 2 oder 3) stammt. Ein **Wechsel** innerhalb einer Gruppe ändert die Programmier**ervorschrift nicht**.

B12. Abbruch:

(X) Zurueck zum System

Die Eingabe von "X" übergibt die Kontrolle an das **Betriebssystem**.

B13. Datenformat:

Der Inhalt eines PALs wird **als Textdatei** erstellt und **verarbeitet**. Sie hat **folgendes** Format:

PAL10L8	INTERNE	BEZEICHNUNG	DATUM	NAME				
					Zeile 1			
					Zeile 2			
					Zeile 3			
					Zeile 4			
					Zeile 5			
					Zeile 6			
					Zeile 7			
0	XXXX	XX00	XX00	XX00	XX00	XX00	XXXX	
1	XXXX	XX00	XX00	XX00	XX00	XX00	XXXX	
2	0000	0000	0000	0000	0000	0000	0000	
3	0000	0000	0000	0000	0000	0000	0000	
4	0000	0000	0000	0000	0000	0000	0000	
5	0000	0000	0000	0000	0000	0000	0000	
6	0000	0000	0000	0000	0000	0000	0000	
7	0000	0000	0000	0000	0000	0000	0000	
8	XXXX	XX00	XX00	XX00	XX00	XX00	XXXX	
9	XXXX	XX00	XX00	XX00	XX00	XX00	XXXX	
10	0000	0000	0000	0000	0000	0000	0000	
11	0000	0000	0000	0000	0000	0000	0000	
12	0000	0000	0000	0000	0000	0000	0000	
13	0000	0000	0000	0000	0000	0000	0000	
14	0000	0000	0000	0000	0000	0000	0000	
15	0000	0000	0000	0000	0000	0000	0000	
16	XXXX	XX00	XX00	XX00	XX00	XX00	XXXX	
17	XXXX	XX00	XX00	XX00	XX00	XX00	XXXX	
18	0000	0000	0000	0000	0000	0000	0000	

```

19 0000 0000 0000 0000 0000 0000 0000 0000
20 0000 0000 0000 0000 0000 0000 0000 0000
21 0000 0000 0000 0000 0000 0000 0000 0000
22 0000 0000 0000 0000 0000 0000 0000 0000
23 0000 0000 0000 0000 0000 0000 0000 0000

24 XXXX XX00 XX00 XX00 XX00 XX00 XX00 XXXX
25 XXXX XX00 XX00 XX00 XX00 XX00 XX00 XXXX
26 0000 0000 0000 0000 0000 0000 0000 0000
27 0000 0000 0000 0000 0000 0000 0000 0000
28 0000 0000 0000 0000 0000 0000 0000 0000
29 0000 0000 0000 0000 0000 0000 0000 0000
30 0000 0000 0000 0000 0000 0000 0000 0000
31 0000 0000 0000 0000 0000 0000 0000 0000

32 XXXX XX00 XX00 XX00 XX00 XX00 XX00 XXXX
33 XXXX XX00 XX00 XX00 XX00 XX00 XX00 XXXX
34 0000 0000 0000 0000 0000 0000 0000 0000
35 0000 0000 0000 0000 0000 0000 0000 0000
36 0000 0000 0000 0000 0000 0000 0000 0000
37 0000 0000 0000 0000 0000 0000 0000 0000
38 0000 0000 0000 0000 0000 0000 0000 0000
39 0000 0000 0000 0000 0000 0000 0000 0000

40 XXXX XX00 XX00 XX00 XX00 XX00 XX00 XXXX
41 XXXX XX00 XX00 XX00 XX00 XX00 XX00 XXXX
42 0000 0000 0000 0000 0000 0000 0000 0000
43 0000 0000 0000 0000 0000 0000 0000 0000
44 0000 0000 0000 0000 0000 0000 0000 0000
45 0000 0000 0000 0000 0000 0000 0000 0000
46 0000 0000 0000 0000 0000 0000 0000 0000
47 0000 0000 0000 0000 0000 0000 0000 0000

48 XXXX XX00 XX00 XX00 XX00 XX00 XX00 XXXX
49 XXXX XX00 XX00 XX00 XX00 XX00 XX00 XXXX
50 0000 0000 0000 0000 0000 0000 0000 0000
51 0000 0000 0000 0000 0000 0000 0000 0000
52 0000 0000 0000 0000 0000 0000 0000 0000
53 0000 0000 0000 0000 0000 0000 0000 0000
54 0000 0000 0000 0000 0000 0000 0000 0000
55 0000 0000 0000 0000 0000 0000 0000 0000

56 XXXX XX00 XX00 XX00 XX00 XX00 XX00 XXXX
57 XXXX XX00 XX00 XX00 XX00 XX00 XX00 XXXX
58 0000 0000 0000 0000 0000 0000 0000 0000
59 0000 0000 0000 0000 0000 0000 0000 0000
60 0000 0000 0000 0000 0000 0000 0000 0000
61 0000 0000 0000 0000 0000 0000 0000 0000
62 0000 0000 0000 0000 0000 0000 0000 0000
63 0000 0000 0000 0000 0000 0000 0000 0000

```

OUTPUT POLARITY WORD XXXXXXXXX

LEGEND: X : FUSE NOT BLOWN (L,N,0) - : FUSE BLOWN (H,P,1)

0 : PHANTOM FUSE (L,N,0) 0 : PHANTOM FUSE (H,P,1)

NUMBER OF FUSES BLOWN - 0

SECURITY FUSE XX

Eine solche Datei wird von dem Programm PALASM 1.7D der Firma MMI mit der Funktion <P> = PRINTS THE ENTIRE FUSE PLOT erzeugt. Benutzen Sie dazu nicht die Funktion (B) - FUSE PLOT OF THE USED PRODUCT LINES.

Die erste Zeile ist die Status-Zeile. Sie wird vor dem Haupt-Menue auf dem Bildschirm ausgegeben. In der Statuszeile muß als erstes der PAL-Typ stehen. Steht dort etwas anderes wird die Datei beim Einlesen abgelehnt. Danach können Sie beliebige Texte in dieser Zeile unterbringen.

In der Zeile 7 muß dann die Liste der "Fuses" beginnen. Es dürfen keine Zeilen oder Spalten ausgelassen werden.

Ein "X" bedeutet eine nicht programmierte Sicherung (logische Verbindung) an der entsprechenden Position, ein "-" bedeutet eine programmierte oder zu programmierende Sicherung (keine logische Verbindung).

Phantomsicherungen (0 oder O) werden nicht verifiziert.

Soll eine neue Datei mit einem Texteditor erstellt werden, so kopieren Sie einfach die Datei mit dem Namen des PALS und der Extension PLO (z.B. 10L8.PLO) ein.

Mit dem Texteditor können Sie nun in dieser Datei die zu programmierenden Positionen editieren. Phantomsicherungen dürfen nicht verändert werden. Achten Sie darauf, daß Sie die Positionen der "Fuses" in der Datei nicht ändern.

B14. Ausfallrate:

Ein unprogrammiertes PAL ist kaum durch den Hersteller zu testen. Durch das Programmieren wird praktisch ein neuer Baustein gefertigt. Daher ist es außerordentlich wichtig, das PAL nach dem Programmieren bei über- und Unterspannung zu verifizieren. Das vorliegende Programm zusammen mit der Platine TIJ-2-1292 hält sich an die Programmiervorschrift des Herstellers. Trotzdem müssen Sie mit einer Ausfallrate von durchschnittlich 2 - 3% rechnen. Der Wert streut von 0% bis 5% pro Charge und ist abhängig von der Anzahl der zu programmierenden Sicherungen.

Über Ausfallraten sollten Sie buchführen. Falls Ihre Programmierstation plötzlich stark erhöhte Ausfallraten produziert, schicken Sie die Platine unbedingt sofort zur Reparatur ein. Für fehlerhaft programmierte Bausteine können wir keine Haftung übernehmen.

B15. Programm ADJUST:

Das Gerät sollte mindestens jährlich neu abgeglichen werden. Den Abgleich können Sie mit dem Programm ADJUST selbst durchführen. Die Beschreibung des Programms und die Abgleichanweisung ist dem Technischen Handbuch, Kapitel 4 zu entnehmen.