

```

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

```

Was steckt in meiner CPS 99. Eine Hardwarebestandsaufnahme mit ROM-Listings. Möglichkeiten der Erweiterung des Systems. Eine Hilfe bei eventueller Fehlersuche

- von Heinrich Cronjaeger

Liebe TI-Freaks !

Die CPS 99 - Station hat in der Zwischenzeit eine recht große Verbreitung gefunden, allerdings gibt es zu dem recht ordentlichen Handbuch von ATRONIC ( das aber auch kleine Fehler aufweist ) keinerlei Hardwareunterlagen, sodaß man über den Aufbau der Compactanlage nicht all zu viel weiß und damit der Aussage von ATRONIC, daß diese Anlage nicht erweiterbar ist, erst einmal Glauben schenken muß.

Der Autor hat sich die Mühe gemacht, die Hardware der CPS 99 aufzunehmen und zu analysieren, um einerseits in der Lage zu sein, an der CPS 99 selbständig Reparaturen vorzunehmen, andererseits die Verträglichkeit mit einer Systemerweiterung zu untersuchen, wie sie im Beitrag " TI 99/4A - Anatomie, Evolution, Revolution " vorgenommen wurde. Die Ergebnisse sollen dem interessierten Leser hiermit vorgestellt werden.

1. Die Schaltungsanalyse

=====

Die Gesamtschaltung der CPS 99 ist in Figur 1, der Schaltkreisbelegungsplan in Figur 2 dargestellt.

Die CPS 99 vereinigt in sich eine 32 KByte RAM-Erweiterung, eine Centronics-Schnittstelle, zwei V.24-Schnittstellen und einen Floppy-Disk-Controller mit zwei Slim-Line-Lafwerken.

1.1 Die 32 KByte-RAM-Erweiterung  
 \*\*\*\*\*

Die RAM-Erweiterung ist mit 4 8-KByte-CMOS-RAM's 6264 realisiert ( IS 15 - IS 18 ). Aus Platzgründen sind jeweils 2 Schaltkreise übereinander montiert, die Pins der 6264 sind bis auf ein Pin alle parallelgeschaltet. Die Selektion der einzelnen 8 Kbyte-Bänke nimmt die IS 26 aus den Signalen IA15 ... IA13 sowie /MEMEN vor. Belegt werden die Speicherbereiche >2000 ... >3FFF sowie >A000 ... >FFFF. Obwohl über das Gatter IS 21.2 das invertierte Signal /DBIN zur Verfügung steht, wird dieses nicht benutzt und die /OE-Eingänge der RAM's liegen auf GND.

1.2 Der DSR-ROM  
 \*\*\*\*\*

Im Bereich >4000 ... >5FFF selektiert die IS 26 den DSR-Bereich durch das Signal /MBE. In der CPS 99 sind zwei Geraetetreiber-routinen untergebracht :

- die DSR der V.24-Schnittstellen einschließlich CENTRONICS
- die DSR des Floppy-Controllers für zwei Laufwerke

Die Programme sind in einem 16-Kbyte-EPROM 27128 ( CPS V2.1 ) untergebracht. Selektiert wird der EPROM pauschal über das Signal /MBE, welche der beiden DSR benutzt wird, wird ueber A13 mit dem Signal R3.1 selektiert. Dieses Signal wird im ROM 3 nicht mit anderen Signalen verknüpft ( siehe ROM-Listings ) und ist identisch dem Signal M1.8 .

Die Bildung des Signals /OE für das DSR-ROM's ist schon schwieriger zu durchschauen. Dieses Signal wird mit dem D-Trigger IS 34.1 synchron zu /CLK3 aus dem NAND-Gatter IS 32.1 getastet. /OE wird genau dann gebildet, wenn :

$$/OE (DSR) = MBE-RS \& /R2.7 \vee MBE-DSK \& /R3.0$$

Aus beiden DSR-Bereichen, die normalerweise 8 KByte groß sind, werden also bestimmte Speicherbereiche ausgeklammert :

$$R2.7 = IA12 \vee /IA12 \& IA11 \& (/IA10 \& IA9 \vee IA10)$$

Aus MBE-RS ( normalerweise >4000 ... >5FFF ) wird mit R2.7 der Bereich >4A00 ... >5FFF eliminiert, er umfaßt also noch den Bereich >4000 ... >49FF.

Aus MBE-DSK ( ebenfalls >4000 ... >5FFF ) wird mit R3.0 eine sehr komplexe Zerlegung des DSR-Speicherbereiches vorgenommen, wie folgende Gleichung belegt :

$$/R3.0 = R2.0 \& /R1.0 \vee /R2.0 \& ( R2.3 \& /M1.8 \vee /R2.3 )$$

Die exakte Beschreibung findet sich im ROM-Listing in Abschnitt 2. Interpretiert besagt die Gleichung folgendes :

Der DSR-ROM wird selektiert,

\* wenn der Adressbereich >5F00 ... >5FFF angesprochen wird, ausschließlich der Bereiche "Lesen >5FF0 ... >5FF7" sowie "Schreiben >5FF8 ... >5FFF

oder

\* wenn der Adressbereich >4000 ... >5EFF angesprochen wird UND entweder der RAM 2114 durch M1.8 abgeschaltet ist und der Bereich >4A00 ... >4AFF angesprochen wird

oder der Bereich >4A00 ... >4AFF nicht angesprochen ist.

### 1.3 Das CRU-Interface \*\*\*\*\*

Über das CRU-Interface werden zum einen die ACC's IS 13 und IS 14, andererseits die bitweise adressierbaren D-Latches IS 12 und IS 19 bedient.

Die ACC's werden mit den NAND-Gattern IS 29.1 und IS 29.4 selektiert :

```

RS232/1 : /CERS1 = R2.2 & R1.5
RS232/2 : /CERS2 = R2.2 & R1.4

mit      : R2.2 = IA12 & /IA11 & /IA10 & IA9 & IAB
          ---> >1300 ... >13FF
          R1.4 = IA7 & /IA6
          R1.5 = /IA7 & IA6

--->     /CERS1 : >1340 ... >137F
          /CERS2 : >1380 ... >13BF

```

Die Adressierung der jeweils 20 Byte großen Ein-Ausgabefelder in den ACC's wird über die Adressen IA1 ... IA5 vorgenommen.

Die bit-adressierbaren D-Latches M1 (IS 12) und M2 (IS 19) erhalten ihren Takt CM1 und CM2 von den Gattern IS 28.1 und IS 28.3 :

```

CM1 = CRUCLK & R2.1 & R1.6
     = CRUCLK & IA12 & /IA11 & /IA10 & /IA9 & IAB &
       /IA7 & /IA6 & (IA4 v IA3)

```

Das entspricht einem aktiven CRU-Adreßbereich von >1108 ... >111F ( und nochmal gespiegelt >1128 ... >113F ). Zur Adressierung der 8 D-Latches werden die Adressen IA1, IA2 und IA4 verwendet, sodaß sich die Ausgänge M1.1 ... M1.8 auf folgenden Adressen ansprechen lassen :

```

M1.1 : >1108, >1128
M1.2 : >110A, >112A
M1.4 : >110E, >112E
M1.7 : >1114, >111C, >1134, >113C
M1.8 : >1116, >111E, >1136, >113E

```

```

CM2 = CRUCLK & R2.4 & R1.1
     = CRUCLK & IA12 & /IA11 & /IA10 & IAB &
       /IA7 & /IA6 & /IA4 & /IA3

```

Das entspricht einem aktiven CRU-Adreßbereich von >1100 ... >1107 ( und nochmal gespiegelt >1120 ... >1127 ) sowie >1300 ... >1307 ( und nochmal gespiegelt >1320 ... >1327 ). Zur Adressierung der 8 D-Latches werden die Adressen IA1, IA2 und IA9 verwendet, sodaß sich die Ausgänge M2.1 ... M2.8 auf folgenden Adressen ansprechen lassen :

```

M2.1 : 0 >1100, >1120      M2.5 : >1300, >1320
M2.2 : 1 >1102, >1122      M2.6 : >1302, >1322
M2.3 : 2 >1104, >1124      M2.7 : >1304, >1324

```

Die Signale M2.1 ... M2.4 sind also der DSR Floppydisc-Controller, die Signale M 2.5 ... M2.8 der DSR V.24/Centronics zugeordnet.

M2.1 wird mit Gatter IS 30.5 negiert und über eine LED mit der Bezeichnung "DSK" angezeigt, über Gatter 29.2 bildet dieses Signal mit dem zuvor invertierten /MBE das Signal /MBE-DSK.

M2.2 erzeugt durch Triggerung des monostabilen Multivibrators IS 10.1 das Freigabesignal DSK.EN.

M2.3 gibt die READY-/HOLD-Steuerung an IS 3 frei.

M2.5 wird mit Gatter IS 30.2 negiert und über eine LED mit der Bezeichnung "Schnittstelle" angezeigt, und durch Verknüpfung mit MBE an IS 29.3 wird das Signal /MBE-RS erzeugt.

M2.6 gibt die Centronics-Schnittstelle durch Freigabe des Rücksetzeinganges des J-K-Triggers IS 20.1 frei.

M2.7 erzeugt über den Negator IS 11.3 das Strobe-Signal für die Centronics-Schnittstelle.

#### 1.4 Bustreiber \*\*\*\*\*

Alle Adressleitungen werden mit den Treibern IS 36 und IS 37 ( 74LS244 ) zur CPS 99 getrieben. Der Datenbus wird mit IS 35 ( 74LS245 ) getrieben, mit DBIN in der Richtung gesteuert und mit dem Signal /GEN.EN aktiviert. GEN.EN wird an IS38 gebildet und durch Gatter IS11.4 zu /GEN.EN umgeformt.

/GEN.EN = /( EXPML v EXPMH1 v EXPMH2 v EXPMH3 v MBE-DSK v MBE-RS )

Gleichzeitig wird aus /GEN.EN durch Verknüpfung mit dem Gatter IS 32.4 an IS 24.2 und Negieren mit IS 30.4 das Signal /RAM und über eine LED die Meldung "RAM" erzeugt :

/RAM = /(GEN.EN & /( MBE-DSK v MBE-RS ))

#### 1.5 Die RS232-Schnittstelle \*\*\*\*\*

Die Adressierung der beiden ACC's wurde schon unter 1.3 besprochen. Der Datentransfer zwischen CPU und ACC erfolgt über den Eingang CRUOUT(ACC), den Ausgang CRUIN(ACC) sowie das Synchronsignal CRUCLK in gewohnter Weise. Eventuell anhängige Interrupts werden mit den Leitungen /IRS1 und /IRS2 an dem J-K-Trigger IS 20.2 mühevoll geodert und mit dem Open-Kollektor-Gatter IS 2.4 auf die Leitung /EXTINT gegeben. ( Hier wäre eine Verknüpfung von /IRS1 und /IRS2 an dem freien Gatter IS23.3 sicher einfacher gewesen.

Die V.24-Schnittstelle ist als DÜE ( Datenübetragungseinrichtung ) eingerichtet.

##### a) Senden : \*\*\*\*\*

Die Funktion des Antwortsignals "Data Set Raedy" wurde mit der Funktion "Clear To Send" gekoppelt. Die Synchronisation mit der DEE ( Datenendeinrichtung, z.B. Drucker ) erfolgt über die V.24-Leitungen Request-To-Send RTS und Data-Terminal-Raedy DTR.

##### b) Empfangen : \*\*\*\*\*

Die Bereitschaftssignale für die DEE CTS und TSR werden permanent durch Pull-Up-Widerstände generiert ( was aber nicht bedeutet, daß die CPS 99 ständig zum Empfangen von Daten bereit sein muß!). Die Modembereitschaft DCD wird durch das ACC-Signal /RTS generiert.

Die TTL- auf V.24-Pegelumsetzung erfolgt mit den beiden Schaltkreisen IS6 und IS7.

#### 1.6 Die CENTRONICS-Schnittstelle \*\*\*\*\*

Das 8-Bit-Datenlatch 74LS374 ( IS4 ) bildet die Datenschnittstelle der CENTRONICS. Die Ausgänge sind immer aktiv. Die Daten werden mit der High-Flanke des Signals IN übernommen.:

IN = MBE-RS & WE & R2.6  
= MBE-RS & WE & IA12 & /IA11

Das entspricht einem Schreibzyklus auf eine MBE - Adresse zwischen >5000 ... >57FF .

Das "Daten-Gültig"-Signal /Strobe wird über M2.7 und Invertierung mit Gatter IS 11.3 erzeugt.

Die Datenübernahme vom Drucker wird mit /ACKNOWLEDGE oder BUSY Low gemeldet. Mit der High-Low-Flanke wird, wenn die Centronics-Schnittstelle mit M2.6 aktiviert wurde, der J-K-Trigger IS 20.1 gesetzt. Das wird letzten Endes mit einem High-Low-Übergang auf der Datenleitung IDO durch den Open-Kollektor-Negator IS 30.6 signalisiert.

/IDO = MBE-RS & ( /BUSY v ACKNOWLEDGE ) & R2.5  
= MBE-RS & ( /BUSY v ACKNOWLEDGE ) & IA12 & IA11

Das entspricht einem Lesezyklus auf einer MBE-Adresse zwischen >5800 ... >5FFF . Eine Sicherung gegen Schreibzyklen auf diese Adressen ( IDO !!! ) ist nicht vorgesehen.

#### 1.7 Der Floppy-Disk-Kontrolller \*\*\*\*\*

Als FDC wird der FD 1773 von Western Digital eingesetzt. Dieser Controller ist sehr komfortabel und kommt mit äußerst geringer Außenbeschaltung aus. Zum Zeitpunkt der Verfassung des Artikels standen dem Autor noch keine Systemunterlagen für diesen Controller zur Verfügung, sodaß die Schaltung aus ihrer Logik heraus interpretiert wird. Wenn die Funktion eines Pins nicht eindeutig ist, wird es als FDC(n) mit n = 1 ... 28 bezeichnet.

Das Kommunikationsprotokoll beim Sektor-Lesen oder -Schreiben erfolgt nicht interruptgesteuert, sondern über eine WAIT-Steuerung des Prozessors in einem Blocktransferzyklus genau dann, wenn das aktuelle Byte gelesen ( Sektor-READ ) oder geschrieben ( Sektor-WRITE ) werden soll. Diese WAIT-Steuerung wird über das Gatter IS3 mit Open-Kollektor-Auskopplung auf die Leitung READY-/HOLD erreicht :

/HOLD = /FDC(27) & /FDC(28) & IA0/CRUOUT & DSK.EN &  
MBE-DSK & M2.3 & R3.2

mit : IA0/CRUOUT muß high sein !!! Das bedeutet, daß die  
WAIT-Steuerung im Vorfeld des eigentlichen Daten-  
transfers stattfindet, wenn der Byte-Zugriff vom Low-  
Teil des Datenbusses her erfolgt und ins Leere geht !  
Das bedeutet aber auch, daß vor dem ersten Zugriff die  
WAIT-Steuerung noch abgeschaltet sein muß ( ~~vermutlich~~  
über M2.3 ). hoch

Zusammenhang?

M2.3 = Freigabe der Ready-/Hold-Steuerung

R3.2 = R2.0 & R1.0 & IA5  
= IA12 & IA11 & IA10 & IA9 & IA8 & IA7  
IA6 & IA5 & IA4 &  
(IA3 & /DBIN v /IA3 & DBIN)

Damit liegen diese Zugriffe im Bereich :

Lesen : >5FF0 ... >5FF7  
Schreiben : >5FF8 ... >5FFF

Der Floppy-Disc-Controller wird genau dann selektiert, wenn :

/FDC(1) = /( MBE-DSK & /IA0/CRUOUT & R3.2 )

mit : R3.2 = siehe oben

Der Datentransfer fällt also in den Bytezyklus mit  
AO=Low, also über den High-Teil des Datenbusses, wie  
das bei Bedienung byte-orientierter Peripherie beim  
TMS9900 üblich ist.

Die DSR des FDC benutzt weiterhin einen 1Kbyte mal 4 Bit RAM.  
Dieser wird erreicht unter folgenden Adressen :

/RAM-CS = /R3.3  
= /R2.0 & R2.3 & M1.8  
= /(IA12 & IA11 & IA10 & IA9 & IA8) &  
/IA12 & IA11 & /IA10 & IA9 & /IA8 & M1.8

Davon ist der erste Term redundant und kann ebenso weggelassen  
werden. Obwohl an den RAM die Leitungen IA0/CRUOUT ... IA9  
herangeführt werden, kann nur ein 256 Byte langer Block  
selektiert werden, genau dann, wenn IA9 = High und IA8 = Low  
führt. Der Bereich liegt zwischen >0A00 und >0AFF in einem 8-  
KByte-Segment, sinnvollerweise in einem MBE-Segment, also auf den  
Adressen >4A00 ... >4AFF, wo auch die Verriegelung zum DSR-ROM  
vorgesehen ist, und er wird mit M1.8 ein- und ausgeschaltet.

Die Floppy-Disc-Controller-Ausgangssignale DIRECTION-SELECT,  
STEP, WRITE-DATA und WRITE-GATE werden mit den Open-Kollektor-  
Negatoren IS 2.1 ... 2.3 und IS 2.6 getrieben, die Latch-Signale  
M1.1 ... M1.4 werden mit DSK.EN an den Open-Kollektor-Gattern der  
IS 9 verknüpft und bilden die Signale SELECT1, SELECT2, HEAD-  
SELECT und MOTOR-ON. Die Eingänge READ-DATA, WRITE-PROTECT,  
TRACK0 und INDEX werden direkt an den FDC geführt und sind mit  
Pull-Up-Widerständen versehen.

Die Kopplung zwischen Floppy-Disc-Controller und Floppy-Disk-Device erfolgt grundsätzlich über Open-Kollektor-Gatter mit Pull-Up-Widerständen an der Empfängerseite.

Die Gatter IS1.2 ... 1.4 bilden den 8 MHz-Quarzgenerator für die Synchronisation beim ~~Beschreiben~~ der Diskette.

1.8 Die Stromversorgung  
 \*\*\*\*\*

Die Stromversorgung ist konventionell aufgebaut und bedarf keiner weiteren Erläuterung. Sie liefert 3 Spannungen für folgende Aufgaben :

- 12 Volt        \* V.24-Schnittstelle
- + 5 Volt       \* TTL-Logik der CPS 99 sowie der Floppy-Disc-Devices
- +12 Volt       \* V.24-Schnittstelle und Floppy-Disc-Devices

Die Reihenfolge des Anschlusses der Kabel vom Trafo ist nicht zu verwechseln. Man beachte aber bei einer Demontage unbedingt den Anschluß des Schutzleiters !

*2091  
 - V.24-Stecker  
 10 Pins*

2. ROM-Listings der TTL-ROM's  
 \*\*\*\*\*

2.1 ROM 1 - IS 27  
 =====  
                   A0    :    IA3  
                   A1    :    IA4  
                   A2    :    DBIN  
                   A3    :    IA6  
                   A4    :    IA7

```
*****
*                               A1/A0                               *
* A4/A3/A2 *    00    *    01    *    10    *    11    *
*****
*    000    * >02 00000010* >40 01000000* >40 01000000* >40 01000000*
*    001    * >02 00000010* >40 01000000* >40 01000000* >40 01000000*
*    010    * >20 00100000* >20 00100000* >20 00100000* >20 00100000*
*    011    * >20 00100000* >20 00100000* >20 00100000* >20 00100000*
*    100    * >10 00010000* >10 00010000* >10 00010000* >10 00010000*
*    101    * >10 00010000* >10 00010000* >10 00010000* >10 00010000*
*    110    * >00 00000000* >00 00000000* >00 00000000* >01 00000001*
*    111    * >00 00000000* >00 00000000* >01 00000001* >00 00000000*
*****
```

- R1.0 = IA7 & IA6 & IA4 & ( IA3 & /DBIN v /IA3 & DBIN )
- R1.1 = /IA7 & /IA6 & /IA4 & /IA3
- R1.4 = IA7 & /IA6
- R1.5 = /IA7 & IA6
- R1.6 = /IA7 & /IA6 & ( IA4 v IA3 )

2.2 ROM 2 - IS 22

=====

A0 = IAB  
 A1 = IA9  
 A2 = IA10  
 A3 = IA11  
 A4 = IA12

```

*****
*           *           A1/A0           *
* A4/A3/A2 *           00           *           01           *           10           *           11           *
*****
* 000 *>00 00000000*>00 00000000*>00 00000000*>00 00000000*>00
* 001 *>00 00000000*>00 00000000*>00 00000000*>00 00000000*>00
* 010 *>00 00000000*>00 00000000*>8B 10001000*>80 10000000*>00
* 011 *>80 10000000*>80 10000000*>80 10000000*>80 10000000*>80
* 100 *>C0 11000000*>D2 11010010*>C0 11000000*>D4 11010100*>00
* 101 *>C0 11000000*>C0 11000000*>C0 11000000*>C0 11000000*>C0
* 110 *>A0 10100000*>A0 10100000*>A0 10100000*>A0 10100000*>A0
* 111 *>A0 10100000*>A0 10100000*>A0 10100000*>A1 10100001*>00
*****
    
```

in irgendeinem  
 8 KByte-Segment

```

R2.0 = IA12 & IA11 & IA10 & IA9 & IAB ----> >1F00 ... >1FFF
R2.1 = IA12 & /IA11 & /IA10 & /IA9 & IAB ----> >1100 ... >11FF
R2.2 = IA12 & /IA11 & /IA10 & IA9 & IAB ----> >1300 ... >13FF
R2.3 = /IA12 & IA11 & /IA10 & IA9 & /IAB ----> >0A00 ... >0AFF
R2.4 = IA12 & /IA11 & /IA10 & IA9 & IAB ----> >1100 ... >11FF
      o. >1300 ... >13FF
R2.5 = IA12 & IA11 ----> >1800 ... >1FFF
R2.6 = IA12 & /IA11 ----> >1000 ... >17FF
R2.7 = IA12 v /IA12 & IA11 & (/IA10 & IA9 v IA10) ----> >0A00 ... >1FFF
    
```

2.3 ROM 3 - IS 33

=====

A0 = M1.8  
 A1 = IA5  
 A2 = R1.0  
 A3 = R2.3  
 A4 = R2.0

```

*****
*           *           A1/A0           *
* A4/A3/A2 *           00           *           01           *           10           *           11           *
*****
* 000 *>0B 00001000*>0A 00001010*>0B 00001000*>0A 00001010*>0B
* 001 *>0B 00001000*>0A 00001010*>0B 00001000*>0A 00001010*>0B
* 010 *>0B 00001000*>03 00000011*>0B 00001000*>03 00000011*>0B
* 011 *>0B 00001000*>03 00000011*>0B 00001000*>03 00000011*>0B
* 100 *>0B 00001000*>0A 00001010*>0B 00001000*>0A 00001010*>0B
* 101 *>0B 00001000*>0A 00001010*>0D 00001101*>0F 00001111*>0D
* 110 *>0B 00001000*>0A 00001010*>0B 00001000*>0A 00001010*>0B
* 111 *>0B 00001000*>0A 00001010*>0D 00001101*>0F 00001111*>0D
*****
    
```



```

/R3.0 = R2.0 & /R1.0 v /R2.0 & ( R2.3 & /M1.8 v /R2.3)
        IA12 & IA11 & IA10 & IA9 & IAB &
        /((IA7 & IA6 & IA4 & (IA3 & /DBIN v /IA3 & DBIN)) v
        /((IA12 & IA11 & IA10 & IA9 & IAB) &
        (/IA12 & IA11 & /IA10 & IA9 & /IAB & /M1.8 v
        /((/IA12 & IA11 & /IA10 & IA9 & /IAB) )

```

R3.1 = M1.8

```

R3.2 = IA12 & IA11 & IA10 & IA9 & IAB & IA7 & IA6 &
        IA5 & IA4 & (IA3 & /DBIN v /IA3 & DBIN)

```

```

/R3.3 = M1.8 & /R2.0 & R2.3
        = M1.8 &
        /((IA12 & IA11 & IA10 & IA9 & IAB) &          redundant
        /IA12 & IA11 & /IA10 & IA9 & /IAB

```

### 3. Erweiterungsmöglichkeiten

\*\*\*\*\*

In diesem Abschnitt soll kurz diskutiert werden, wie die CPS 99 erweitert werden kann, z.B. durch Kopplung der CPS 99 über ein Kabel und Stecken desselben in die Erweiterungsbox aus dem Beitrag "TI 99/4A-Anatomie, Evolution, Revolution". Grundsätzlich muß an der Schaltung nichts geändert werden, aber es fehlt ein wichtiges Signal : "RDBENA". Dieses Signal aktiviert die Datentreiber des Kabels auf der Konsolen- und der Boxseite. Generiert wird dieses Signal immer durch die Funktion "General Enable", in der CPS 99 heißt das entsprechende Signal "GEN.EN". Dieses Signal ist lowaktiv und wird über Open-Kollektor-Gatter erzeugt, sodaß die einzelnen Baugruppen "Wired-Or"-verknüpft werden können.

Erzeugt werden kann dieses Signal einfach aus Verknüpfung eines nicht benutzten Open-Kollektor-Negators ( z.B. IS 30.1 oder IS 30.2 ) mit dem Ausgang der IS 31. Das so gewonnene Signal /RDBENA muß nun an der entsprechenden Stelle auf den Bus gegeben werden ( im Ausführungsbeispiel auf KB A23 ). Nützlich wäre auch noch ein Treiben der Ausgänge CRUIN der ACC's. Dies ist möglich, wenn man die Ausgänge von der Leitung CRUIN trennt, sie mit 4,7 kOhm pull-Up't und parallel an ein freies Negator-Gatter legt und dessen Ausgang an ein freies Open-Kollektor-Negator-Gatter. Dessen Ausgang wird dann wieder mit der Leitung CRUIN verbunden.

Die im Ausführungsbeispiel beschriebene interne 16-Bit-RAM-Erweiterung kann dann natürlich nicht benutzt werden, es sei denn, man legt die CPS-RAM-Erweiterung durch Abtrennen der Signale /EXPML, /EXPMH1, /EXPMH2 und /EXPMH3 von der IS 31 tot. Verzichtet man auf die 16 Bit-Erweiterung, will aber trotzdem die Hardwareergänzungen nutzen, so stellt man die Verbindungen /EXL, /EXH1, /EXH2 und /EXH3 nicht her, legt diese Eingänge statt dessen auf +5 Volt und bestückt die RAM's nicht.

Diese Möglichkeiten seien als Anregung für diejenigen gedacht, die den TI 99/A über sein geplantes Anwendungsgebiet hinaus verwenden wollen.

# Schematischer Bestückungsplan der CPS 99

