

KDT-Lehrgang "Erweiterung MC 80 System"  
=====

Ø Einleitung

- 1. MC 80 C SOS (Ein CP/M-kompatibles Betriebssystem für MC 80.3x)  
*Koll. Damme 20.3.*
- 2. Slave - Einheit (Hardware)
  - Slave-Einheit/Baugruppen mit Stützspannungsüberwachung
  - Digitale E/A-Baugruppen (DEG, DAO, DAG) *Koll. Ruscher*
- 3. Analog - Digital Wandler (Hardware) *Koll. Axmann 27.3.*
  - Umsetzverfahren (Vor- und Nachteile)
  - C 500/ C 504 - Wandlersysteme
  - Meßstellenumschlag
  - Rechnerschnittstelle
  - Anwendungsmöglichkeiten
- 4. Master-Slave-System (Software) u.a. Software
  - SLOS *Koll. Ott*
  - MAKS 30
  - ADU Software
  - EGOS 3.0 E
- 5. Periphere und Ergänzungsbaugruppen, Interface *Dr. Kochendorfer*
  - Display - Ansteuerungen (V/S/QEA)
  - Speichererweiterungen
  - Interfaceschnittstellen Seriell/parallel
  - Prozeßankopplung

Damme 395 2851  
 Axmann 584 186 (4)  
 Ruscher "  
 Ott 584 120  
 Dr. Kochendorfer 395 2713  
 (2735)

Nachnutzung:  
 BfN Erhardt 584167  
 KDT Kolln. Röhrer  
 23338-9

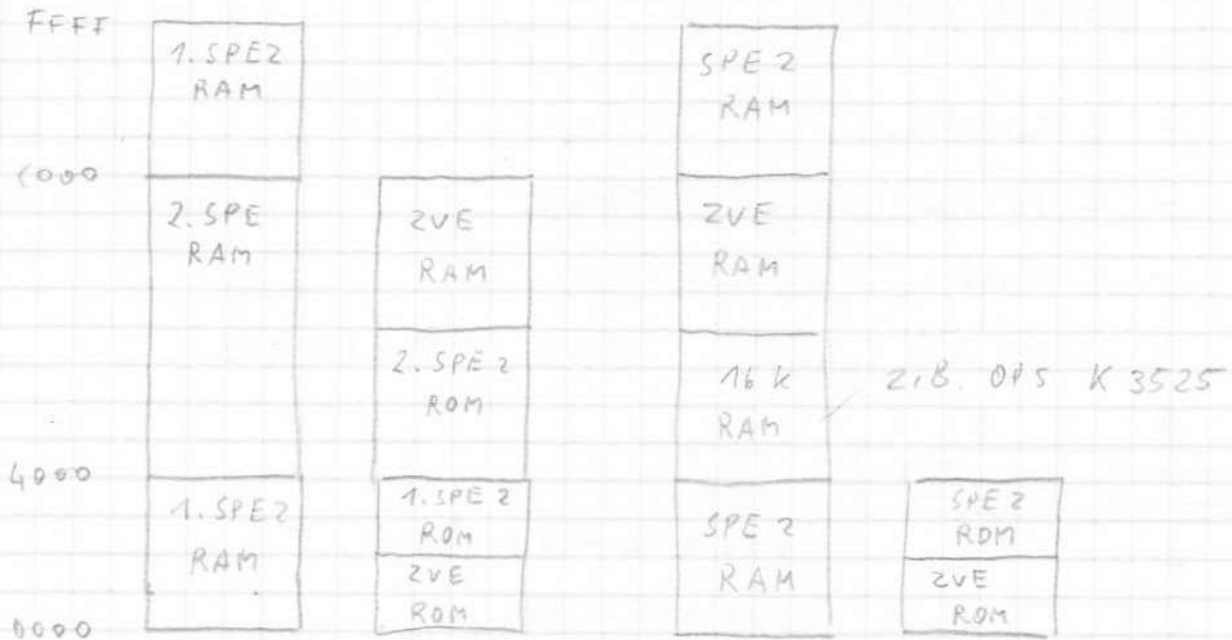
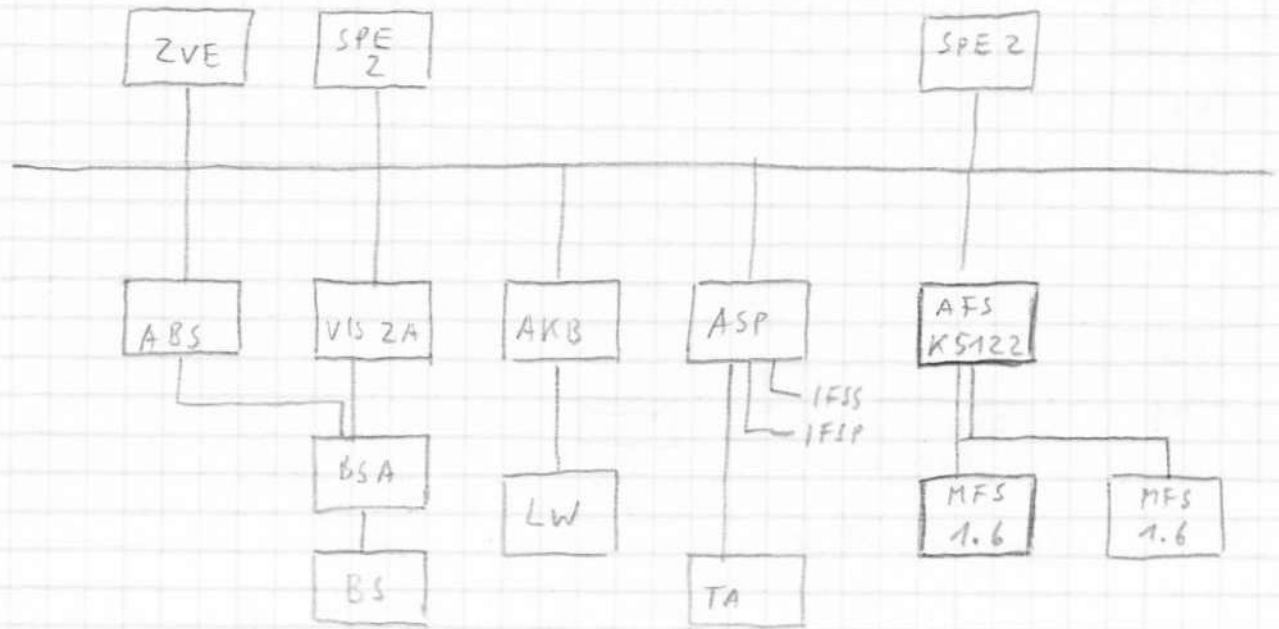
Applikation  
 Gebhard 58 22 12

Kundendienst  
 Koll. Schulze 22661-3

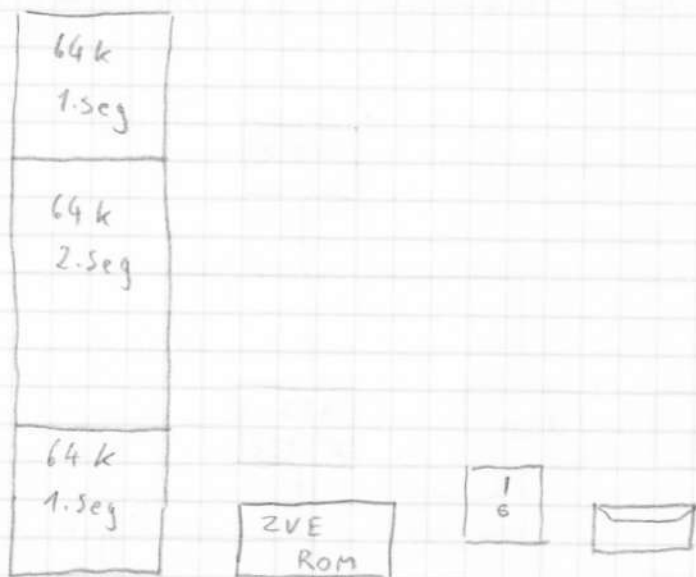
KIB Absatz  
 Kolln. Linke 3950

(Kolln. Wolf)  
 ← Ersatzteillager

# Betriebssystem (SOS für MC8030 (CP/M komp.))



Version 1.6 mit 2 SPE 2      Version 1.5



Version 1.x

• CSOS - Varianten

mit 2 SPE 2 oder  
1 SPE 2 / 1 OPS } mit VIS 2A

- 2 Kassetten
- 2 FL-LW (physol), 1 Kass.
- 2 Kass., 1 FL-LW
- 2 FL-LW (phys.) , 1 Kass., 1 RAM-FL.



A	B	$256 \times 16 \times 80$ !
C	D	$1024 \times 5 \times 80$

### 3 Kommandosyst. CSOS / SCP

- Kommando eing. ist mögl. wenn LW-Meldung A > oder B > auf BS erscheint

- man untersch. zw.:

1. Steuer u. Editor komm.
2. resid. Komm. (DIR, ERA...)
3. transiente Komm. (v. Progr. laden)  
(FORMAT, CINIT...)

### Bestandteile d. Nachnutzung

- CSOS
- 10 x 2716
- 1 Dig. Kass. } vom Nachnutzer
- Hardware dok.
- Anwender dok.

NN-Preis 1. M 80.30 CSOS STM Kass., RTM Disk  
2. EPR-Softw. 1TM

VEB Elektr. Gera

Abt. ENQ

Parkstr. 3

Gera

6500

MC 80.30 CSOS / 1 (2) Version - - -

CPM-komp.

Single User  
Operating  
System

gem. Speicher.

mit DSE (BWS ab F800)

mit VIS 2H

### Bestandteile v. CSOS

CCP Console Command Prozessor

BDOS Basic Disk Operating System

BIOS Basic Input/Output System

- Bei CSOS werden die Funktionstasten vom Betriebssystem benutzt

- Taste 6 (nur bei Kass.) A ~~6~~ liest bzw. schreibt  
Directory (ein/aus)

- Taste 5 (nur bei Kass.) B

- Taste 4, Escape

- Taste 3, Umsch. Klein-Groß in d. ersten Ebene

- Taste 2, Control-P

- Taste 0, Control-C bzw. auf OFF-Taste, WARM-START

- Taste 7, RECALL, Sprung zur Adr. 100

- Taste 1, WARM-RESET, Soft-Reset

6	5
4	3
2	1
0	7

Bestandteile : STM +5V / 100W  
 (Grundbest.) STZ -5V / +12V / -5V / +15V / -15V -je 20W  
 AVS Anschlusssteuerung V10 / IFSS  
 ZVE  
 AKKUS - Nied - Akkus 3 x KR3  
 STU - Stützespannungsumschaltung

Maße: 480 mm x 236 mm x 240 mm

Lieferumf.: Grundgerät, Softw. SLOS, Netzkabel, IFSS-Kabel

Erweiterungen:

DEG - 12V Dig. Eing. Baugr.  
 (5V)  
 (24V) 32 Kanäle über Optokoppler galv. getr.

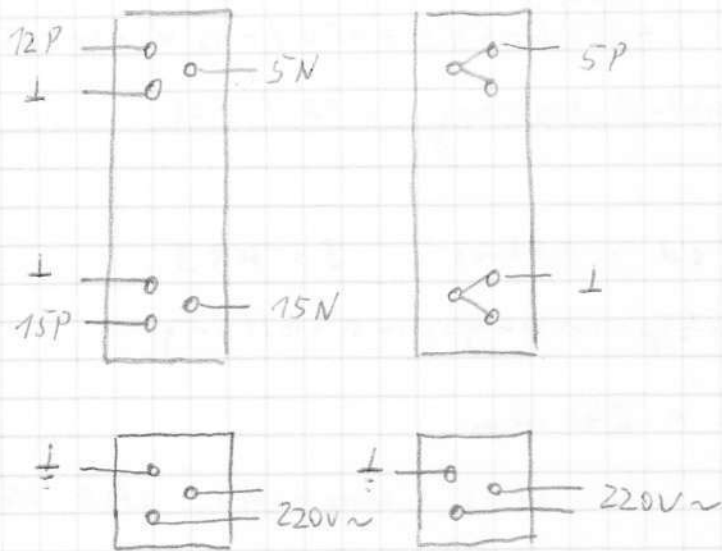
DAG Dig. Ausg. Baugr. (Relais) R6K  
 32 Kanäle

DAO Dig. Ausg. über Optokoppler  
 32 Kanäle - Ausg. über E 412

ADU Analog - Digital - Umsetzer  
 8 Kanäle, 14 bit, 150 ms, keine galv.  
 Trenn.

SPE 1 16 kByte CMOS - RAM (U 224)

# Spannungen an STM / STZ (MC 80.30)



## • AVS: (zur Kopplung von Rechnern geeignet.)

1 IFSS-Kanal, 1 V24-Kanal

• 5P	350 mA
15P	25 mA
15N	30 mA
12P	80 mA

## • DEG:

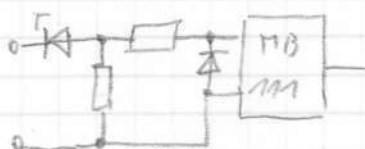
- interruptfähig
- 32 Eing. (5V o. 12V o. 24V)
- kann als PIO-E/A-Karte Verwendung finden, da PIO-Signale auch auf Steckv. geführt sind
- bis 40 kHz im Interruptbetrie.
- Peripheriesteckv. BL 304/58 TGL 29331/03 auf Karte
- 5V - 350 mA / Stromaufn.

DEG 5V



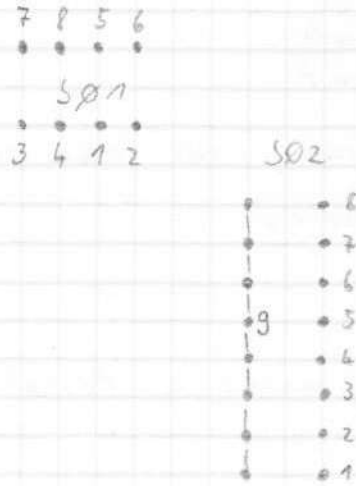
$I_E$  ca. 15-30 mA

DEG 12V  
24V



5

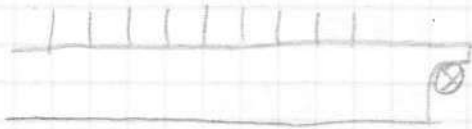
- Leitung zw. Geber und DEG verdrillt
- Adressierung: von xx00 ... xxFF (einstellbar)



Adr.	S01	S01
00	2-6	4-8
40	2-6	3-7
80	1-5	4-8
C0	1-5	3-7

	S2
80	1-9
88	2-9
10	3-9
18	4-9
20	5-9
28	6-9
30	7-9
38	8-9



Stw.	PIO 1	A	-	S
		B	-	7

Det.w.	A	-	S
	B	-	6

Stw.	PIO 2	A	-	S
		B	-	3

Det.w.	A	-	S
	B	-	2

DAG:

- 32 Ausg. (Relais), galv. getr.
- Adr. ber. xx00 ... xxFF
- D126 - Ausg. auch auf Periph. Stecker geführt



S02	Adr.	AB 6,7
	00	6-7 10-11
	40	4-5 10-11
	80	6-7 8-9
	C0	4-5 8-9

+ add S01 AB 3,4,5

80	1,2	28 11-12
88	3,4	30 13-14
10	5,6	32 15-16
18	7-8	
20	9-10	





S02 AB 2

00 1-2  
04 3-2

stromvers : 5P - 430 mA  
12P - 420 mA

\* 24P ext. 30 mA alle Relais aus

- Peripherie st. SL 324/58 (Anwender)
- Relais belast. max. 0,2 A  
max. 42 V<sub>~</sub>

- max. Spannung zw. Rechnermasse u. Periph. Masse  $\leq 42V$

### DAO :

- dig. Ausg. über Optokoppler u. E 412
- Peripher. st. SL 322/39 TGL 29 331/24
- 24P ext. f. E 412
- Stromaufnahme : 5P - 800 mA  
\* 24P ext. - 600 mA

- max Freq. 4 kHz

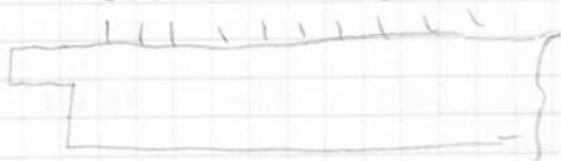
S01 AB 2,3,4

S02 AB 5,6,7



00	16 - 24
04	17 - 24
08	18 - 24
0C	19 - 24
10	20 - 24
14	21 - 24
18	22 - 24
1C	23 - 24

00	7 - 15
20	8 - 15
40	9 - 15
60	10 - 15
80	11 - 15
A0	12 - 15
C0	13 - 15
E0	14 - 15



# 6 Analog - Digital - Umsetzer

(Axmann)

## 1. Auflösung $\neq$ Genauigkeit

Fehler: - Nichtlinearität der Quantisierungskennlinie (Wandlerbeant.)

- Nullpunktfehler
- Umpolfehler
- Temp. driftfehler, Klimafehler
- Langzeitdriftfehler
- Eich- u. Kalibrierfehler
- Netzspannungsschwankungen

(ADU > 15 Bit), (-2 ... +2V)

## 2. Umsetzrate

## 3. Eingangsspannungsbereich

## 4. Eingangswiderstand

## 5. Eingangsmuxer

## 6. Eingangsschutzbeschaltung

## Wandlerprinzipien

### 1. Parallel ADU (sehr schnell, sehr aufwendig)

### 2. Stufen - ADU

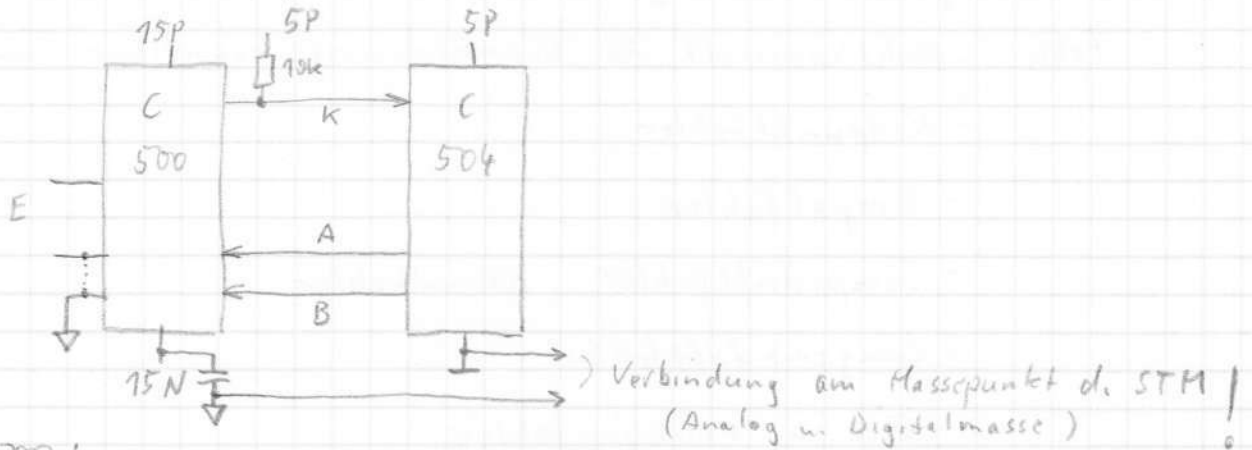
- Sample and hold - Schaltung erforderlich MAC 398
- sukzessive Approx.
- Beispiel: DAC 12, CS71C, CS74C

### 3. Serien - ADU

- Nachteil bei Spannung - Freq. - ADU, Benutzem. tel. geht auf Ums. ein
- Dual - Stop - ADU (zwei - Flanken - Sägezahn)
- Charge - Balancing

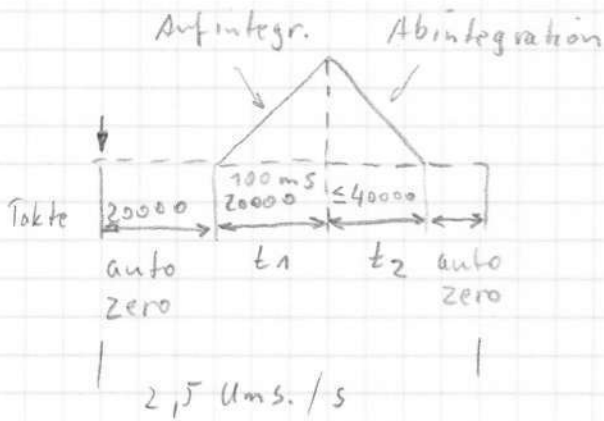
# Wandler system C500 / C504

Lit: RFE 3/4 86



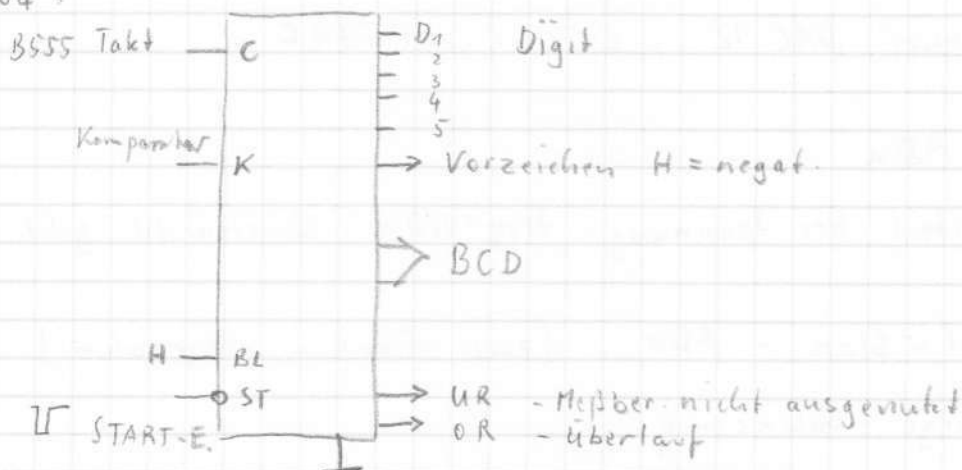
• C500:

Achtung: Ausfall d.  $\pm 15V$  oder Ausfall d. nur  $+5V$  führt zur Zerstörung d. C500

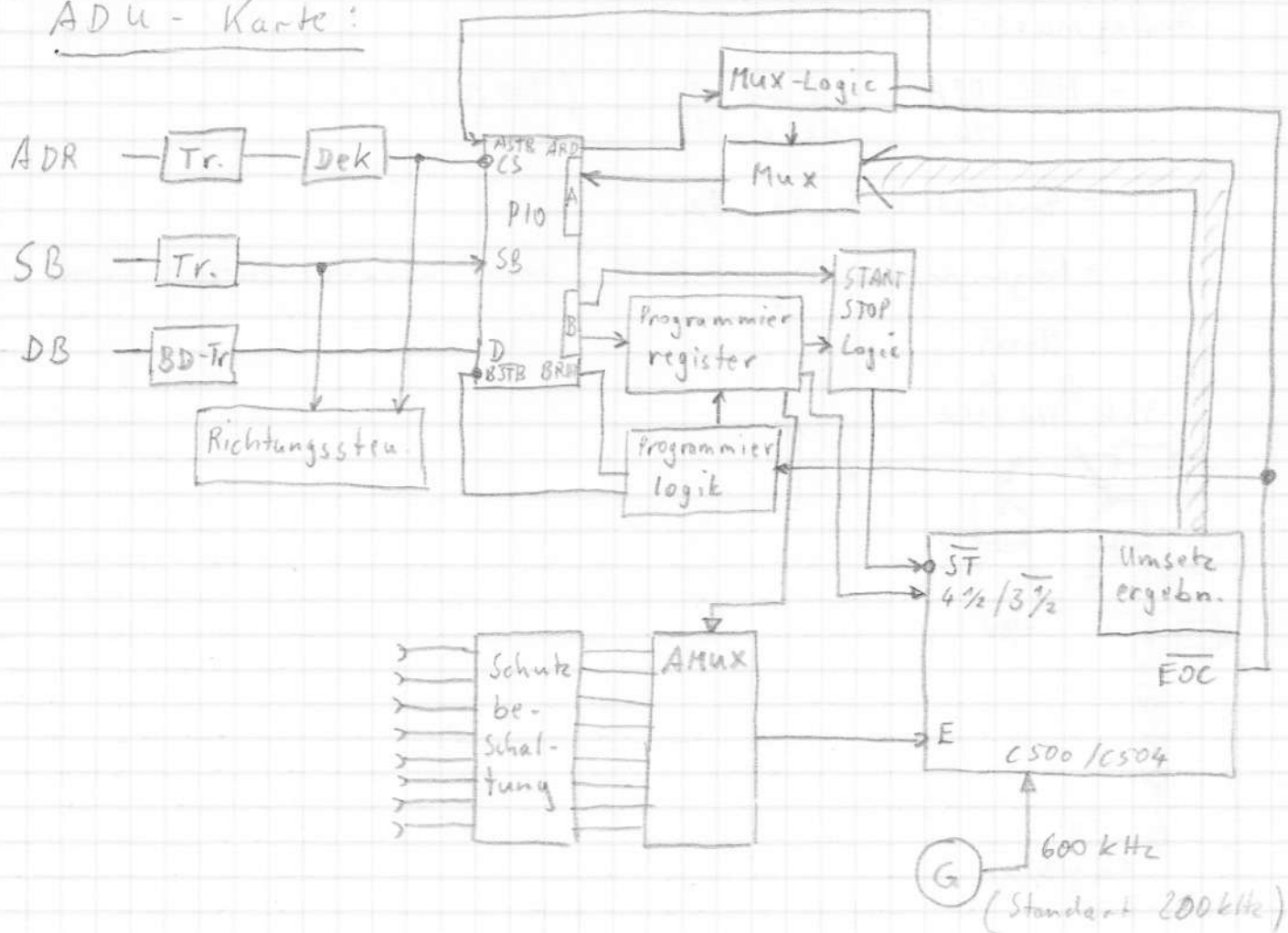


- Ref. mögl. ext. 1,000... V
- Refsp. erz. mit B 589 Ng (1,235V)
- C500 - 4,5 Digit  $\pm 1,9999$  (> 15 Bit Auflösung)
- C501 - 3,5 "  $\pm 1,999$

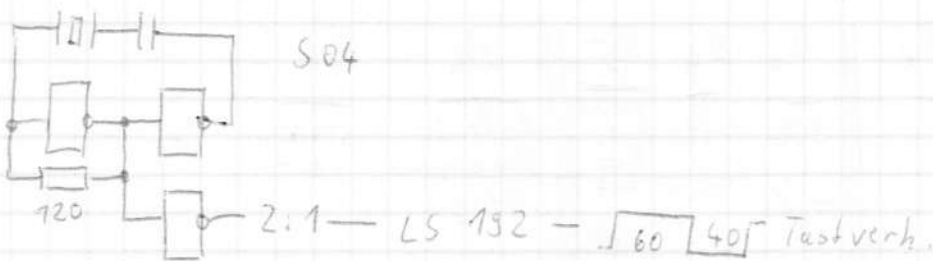
• C504:



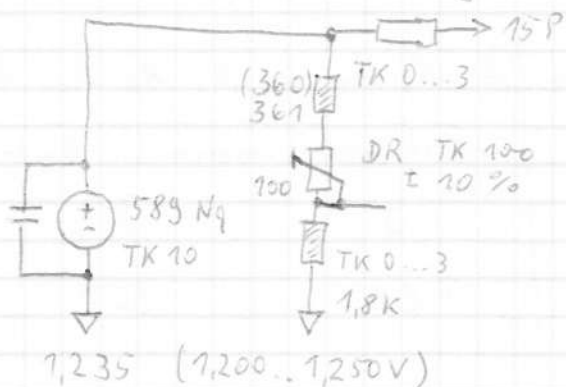
7

ADU - Karte:

## Taktgenerator:



## Ref. spannungserzeugung:

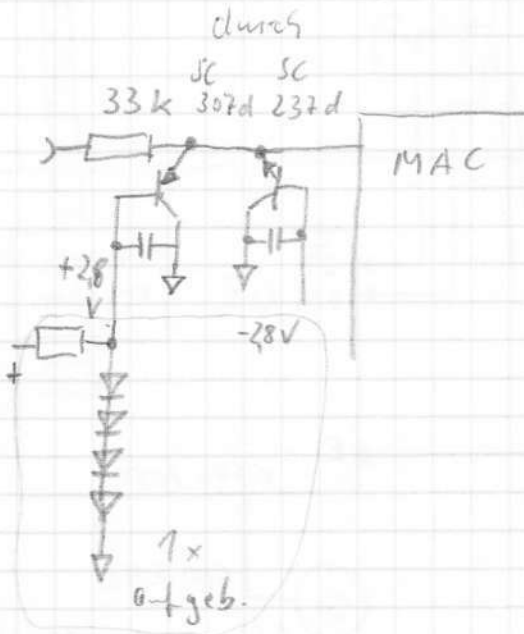


# Analog muxer :

- MAC 08A 8 auf 1 (300  $\Omega$ )  
 (16) 16 " 1

- Einschaltzeit ca. 1  $\mu$ s

- Eingangsspannung < 11V, sonst schaltet entspr. Kanal ein/aus

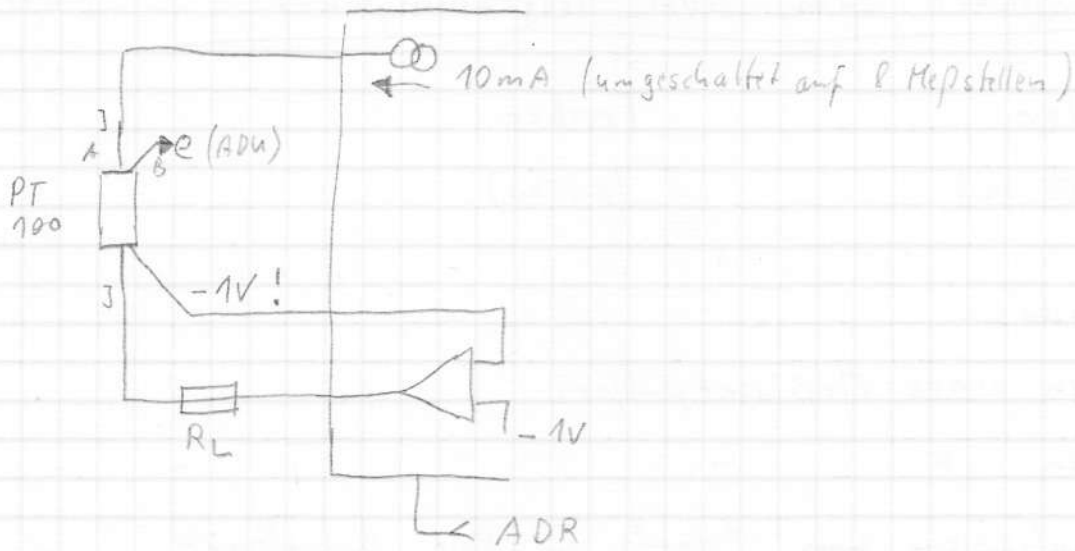


## • Programmierung d. ADU

$B_0$	$B_1$	$B_2$	$B_3$	$B_4$	$B_5$	$B_6$	$B_7$
Adr. AMUX			$3\frac{1}{2}$	lfd.	start d. 2. Out Bef.	Ende d. Dauerum- setz.	START BIT
			$4\frac{1}{2}$	einzel	selbst start nach $\sim$		

$A_7$ ...	$A_4$	$A_3$	$A_2$	$A_1$	$A_0$	$A_7$ ...	...	$A_0$	$A_7$	$A_6$	$A_5$	$A_4$	$A_3$	$A_2$	$A_1$	$A_0$
						zuerst übernommenes Byte										
F	$\overline{UR} \equiv L$	$OR \equiv L$	$+L$	$1_H$		g		g		g						g
			$-H$	$0_L$		BCD		BCD		BCD						BCD

8



# Programmpaket zum ADU-Programmpaket

(011)

Handler

(MC 80.3x)

Treiber

(K1520)

Kommandos:

- Anlegen eines Meßwertpuffers
- Löschen " "
- Initialisierung eines Kanals in best. Betriebsart
- Anhalten der laufenden Messung
- Rufen Meßwerte aus Puffer
- Abgleich

ADU DEF Länge, name

- legt Meßwertpuffer als @b-Daten in RAM an
- Länge - 4-stellig hexa

~~Errors~~ Errors : C7 nicht genügend freier RAM  
48 Kettenfehler  
44 Syntax

ADU DEL name

- angegebener Puffer wird gelöscht und freigegeben für neue Aufgaben

Errors : C7 Dat. nicht verfügbar  
48

ADU INIT betriebsart, kanalnummer, puffername

0-7, 0-7, vorh. Puffer

- ADU wird wie angegeben initialisiert
- Puffer wird gelöscht
- bei Autostart → Starten der Messung

sonst → " " " mit FT1  
(rechts, 2. v. u.)

9

- bei Einzelumschaltung: stop nach 1. Umschaltung
- bei Dauerumschaltung - unterbrechen mit FTZ (Links u.)
- keine Anzeige der Werte, nur d. Puffer wird gefüllt

Betriebsarten:

Ø	Dauerumschaltung	$3\frac{1}{2}$ Stellung	Taste
1	"	$4\frac{1}{2}$ "	"
2	Einzelumschaltung	$3\frac{1}{2}$ "	"
3	"	$4\frac{1}{2}$ "	"
4	Dauerumschaltung	$3\frac{1}{2}$ "	Auto
5	"	$4\frac{1}{2}$ "	"
6	Einzelumschaltung	$3\frac{1}{2}$ "	"
7	"	$4\frac{1}{2}$ "	"

Errors: C7 - Daten fehlt  
44 - Syntaxe

ADULIST name

- Ergebnisse werden auf dem BS gelistet
- Anhalten d. Anz.: bel. Taste
- Fortsetz.: ENTER
- Abbrechen: OFF

ABGLEICH

- ständige Messung in Betriebsart 7 Kanal Ø
- Messung wird auf US angezeigt
- Verlassen RESET



## ADU - Treiber

- auch in anderen K1520 Syst. nutzbar
- Ansteuerung über Schnittstellen
- Parameter in Registern

INIT Anfang + 0 ; Initialisierung d. ADU

INPUT : A Kanalnummer

E Betriebsart

HL Meßwertpufferanfang

BC Meßwertpufferlänge

OUTPUT : /

Errors : Cy = 1 , A = Fehlercode

Fehlercode : 44H ( BC + HL > 0FFFFH )  
BC = 0

STAR Anf. + 3 ; Starten d. Umsetzung

INPUT : Cy = 0 Ablegen d. Daten ab akt. Pufferzeiger

Cy = 1 " " " " Pufferanf.

OUTPUT : /

Errors : Cy = 1 , A = 49H - kein Start möglich

STOP Anf. + 6 ; Stoppen d. Dauerumsetzung

INPUT : /

OUTPUT : /

Errors : Cy = 1 , A = 49H - kein Stop möglich

MTAB Anf. + 9 ; Listen d. Umsetzungsergebn. im Textpuffer

INPUT : IX - Zeiger auf 1. Byte des Meßw. im Puffer

BC - Meßwertnummer für Nummerierung

( Anf. + 0CH ) - Stellenzahl 4 o. 5

10 OUTPUT : HL - Zeiger auf Textpuffer mit ~~der~~ Meßwertzeile  
Abgleich mit 0FFH

Errors : Cy = 1 kein Meßwert

Beispiel : Darstellung von MW auf BS (grafisch als  
Pixelzeilen)

CLB : EQU 0BD9H

PAR0 : EQU 685BH

CALL CLB ; BS löschen

LD B, 0 ; y-Koordinate

MESS PUSH BC

LD BC, 4 ; Meßwertpufferlänge

LD A, 0 ; Kanal

LD E, 7 ; Betriebsart

LD HL, PUF ; Meßwertpuffer

CALL PAR0

RC

LD BC, 180H ; Zeit

TIM : DJNZ TIM ; warten

DEC C

JRNZ TIM

LD HL, PUF

INC HL ; letzte Stelle uninteressant

LD A, M ; vorletzte Stelle

LD D, 0FFH

HEX : INC D

SUB 1

JRC HEX0

DAA

M

JRNZ HEX

HEX0 : LD A, 0 ; Vorgabe

INC HL

BIT 0, M

JRZ HEX1

LD A, 64H ; = 100 dez.

HEX1 : ADD D

LD D, A

POP BC

LD C, 0C1 ; VIS : Bildp. hell, links

INC D ; 1. Punkt freilassen

VIS : OUT D

DEC D

JRNZ VIS

INC B ; neue Y-Koord.

JRNZ MESS

XOR A ; Cy = 0

RET

PUF : DB 0

DB 0

DB 0

DB 0

DB 0FFH

3Byte  
an  
VIS  
BC, D

## 12 Slave - Kopplungs - Softw.

- Austausch v. Daten, Progr. u. Steuercode
- Aktivität geht grundsätzl. von Master aus
- Übertragungsrate: 1FSS : 19200 Bdt > 2KByte/sec

2 Programmteile - Masterprogr. (MC80.3x)  
Slaveprogr.

### Masterprogr.:

Bedienteil : (Kommandoeing., Parameterüberg., Datenabgabe)

Kopplungsteil (Initialisierung d. Kopplung)

(Empfangen, Senden, Starten von Progr.)

- (Datenbyte senden, Steuerbyte senden)

Interruptroutine - Datenbyte empfangen, abspeichern

- Steuerbyte " , ausführen

Parameter aktualisieren

### Softw. f. Datenverkehr

MAKS 30

SL05 (auf ROM)

(als Arbeitsbereich auf Kern für

OCED 30)

- Lauffähig machen von MAKS

1. INIT

2. READ OCED.RAM

3. RKET 01, 6400, BAF8

4. READ MAKS.30

5. OEDIT MAKS

6. Progr. soll auf Ladeadr. laufen

^ B ENTER

Progr. soll auf EPROM gelad. w.

^ B Adresse

Kettung im Zielbereich ändern!

7. RKET 01, 6405, 640D

8. RKET 01, (72EF), A000

Kennb. adr. v. "F"

} freien  
RAM schaffen

9. RKET 01, A000, BAF8

### Kommandos

- Segmentnummern können entfallen, dann = 1
- Segmentnummer immer hinter jew. Adresse z.B. 4000102
- Startadresse kann entfallen, dann = Ladeadresse
- Name muß für Parameterdatei angeg. werden KB 05

Fehler: Error 44 Syntax  
45 Slave antwortet nicht  
48 RAM-Kennung def.  
C2 Slave nicht bereit  
C7 Datei nicht gefunden, kein freier RAM

### Schnittstellen

Nutzung d. Koppelung aus Anwenderprogrammen

- SINI - Initialisierung
- SENS - Senden v. Daten
- EMPF - Empf. v. "
- STAR - Starten v. Progr. im Slave

dazu MODE - Merkwörter mit Parametern füllen

MODE + 0 ... 7, 16, 17

MODE + 15 Bit 7 - Besetztbit

phys. Ebene:

OUT - Datenbyte senden

INP: Byte in A

OUT 1 - Steuerbyte senden

INP = Byte in A

13

Prinzip d. Datenübertragung

- Byteweise, bitseriell
- Datenbytes, Steuerbytes
- bei Steuerbytes ist 10H vorangestellt
- Daten 10H: doppelt senden / empfangen
- Steuercode s: 80... 1FH (außer 10H!)
- verwendet 0, 1, 2, 3, 5, 6, 11, 15, 17

Interrupt - Empfangsinterr.

- Zeichen empfangen
- Entschieden: Steuer- / Datenbyte
  - UP entsprechend
  - abspeichern, Param. aktualisieren
  - Tabelle rufen
- Daten werden unblockiert übertragen, Test über CRC-Summe

Slave - Programm

- nach RESET:
  - Initialisierung, Stack, Interruptsystem, Segmentschaltung, Speicher freigabe (wie MC 80.3x)
  - Warten auf Master-Kommando oder NMI  
Sprung auf 800H
- Kopplungsteil  $\hat{=}$  Masterprogramm
  - nur Steuerzeichen reaktionen verändert, dadurch Master immer aktiv
  - Programmstart im Master durch Slave möglich
  - Slave voll segmentfähig, Segmentprogr. wie MC 80.3x aber ab Adresse 600H
  - wichtige Adressen im Slave
- RST 1...7 wie MC 80.3x (0FEEB...)
- RST 1 belegt mit Steuerzeichen ausgabe
- RST 2 " " Programmstart
- NMI führt auf 800H (Himmeler-Rom)

## Schnittstellen Slave

- MODE + 15, bit 7 = 1 Slave besetzt
- Steuerzeichen zum Master senden : 308 H o. RST 1  
INP : A = Steuerzeichen
- Masterprogr. starten : 302 H o. RST 2  
INP : IHL-Adresse, A-Segm. (im Master)

## Beispiel :

- Tastendruck am Master soll zur Ausg. d. Tastencodes auf DAG im Slave führen
- Slaveprogramm schreiben, binden auf Slaveadresse, senden

```
LD HL, MERK
LD A, M
OUT DAG
RET
```

```
DAG EQU 04
```

```
MERK EQU 0D000H
```

- Masterprogramm schreiben, binden, starten

```
SINI :
SENS :
STAR :
MODE : EQU 0FC70H
TST : EQU 0BCD0H
```

```
TAST : CALL SINI ; init
CALL TST ;
JRZ TAST ; Werten auf Taste
LD (PUF), A ; Tastencode
LD IHL, MERK
LD DE, MODE
LD BC, 18
LDIR ; Modezellen laden
CALL SENS
CALL STAR
JR TAST
```

```
PUF : DB 0
MERK : DB 1
DA 0D000H
DB 1
DA 0C000H
```

DA 1 Länge  
; Segment in Slave  
; Adr.  
; Segm. Master  
; Startadresse

DA Ø  
 DA Ø  
 DA Ø  
 DA Ø  
 DA PUF ; Adr. Master

EGOS 3.0 E

- Syntax geändert

- neue Kom. CALC - Hex-Taschenrechner +, -, \*, /
- CAT - Auslisten aller Progr.
- CAT x - " " " mit Kennb. x
- CHAIN name - Stapelbearbeitung
- COMP Anf. adr., Endadr., Vergl. adr. - Vergleich
- DAKT Treibername - Treiber wird im RAM gelöscht
- DEFINE Nr. Treiber, name - Treiber def.
- DEZ hex-zahl - Umw. hex -> dez.
- DIR -
- FIND Anf. Adr., Endadr., Byte(s) (Text) - Suchen
- FILL - Anf. = Ende, 1 Byte
- HEX dez. Zahl - Dez. - Hex
- KILL name - löscht angeg. Daten
- RAM - verbessert (Go Adr.  
 L löschen  
 B Byte  
 A ASCII)
- RELAD - Verschiebeadr. Tab wird nach Aktion gelöscht
- RKET Quelladr., Zieladr. -
- TINI Treiber nr. - Initialis.



# Speichererweiterung:

SPE 1	CMOS / SRAM	16 KB	
SPE 2	DRAM	32 KB	EPROM 8/16 KB
SPE 4	EPROM	48 KB	

Speichern u. Freigaben v. Speicherbereichen (hard-/softwaremäßig)

- Lesesperre - Datenschutz, Zugriffsrechte
- Schreibsperrung - Testen v. Prog., überschreiberschutz, "Virien"
- Zugriff - mehrere Module auf gleichem Adr.

Segmentierung - Erhöhung des Adressraumes, mehrfache Anordnung des gleichen Speicherbereiches

- Umschaltung: nur 1 Segm. aktiviert (PIOB / ZVE)  
(4000 - DFFF)
- SPE direkt programmieren

ZVE. → Koppelbus → X2: A21 - /MEMDI 1  
PIOB → X2: A21 - /MEMDI 2

X1: C9 - /MEMDI alle Sp. aus

## SPE 1:

U 224 (U 214)

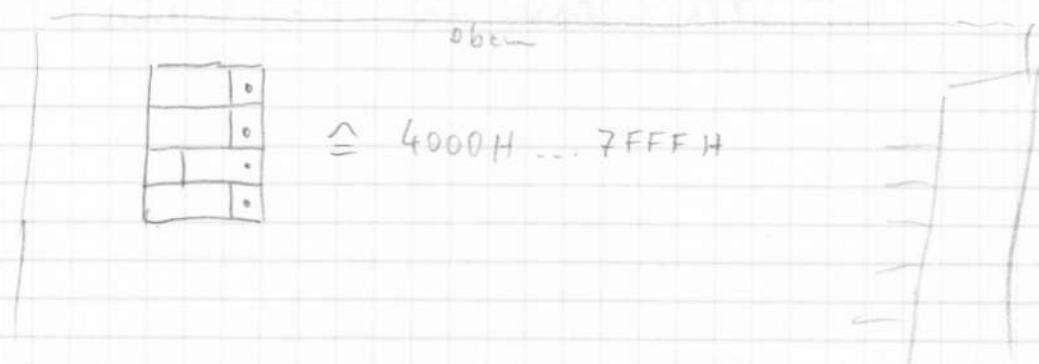
Stromversorgung normal: SP, SPG (AC 3) = SP  
(0,3 A)

Schlafzust.: SPG = 2...SP  
(40 mA)

Sperrsignal für Speicher bei SP < 4,75V Lakt.

Adressbereich: 16 KB zusammenhängend

- Anf. adr. in Stufen v. 4 KB einstellbar



15

Sperrn durch MEMDI : Zugriff zum ges. Speicher, Auswahl durch Wickelverb.

Sperrn programmierbar : über E/A - Adresse  
 - Zugriff- / Schreibschutz  
 - Bereichsgröße 1, 2, 4, 8, 16k } über Wickel

$A_{10} \dots A_{15}$  : zu beeinflussender Bereich

$A_7 \dots A_1$  : E/A - Adr.

$A_0$  :  $\begin{cases} 1 & \text{Freig.} \\ 0 & \text{Sperr} \end{cases}$

E/A - Ausg. mit Beeinfl. des oberen Adressraumes

OUT n  $n \rightarrow A_0 \dots A_7$

A  $\rightarrow D_0 \dots D_7$  nicht benötigt

$\rightarrow A_8 \dots A_{15}$

OUT r C  $\rightarrow A_0 \dots A_7$

B  $\rightarrow A_8 \dots A_{15}$

r  $\rightarrow D_0 \dots D_7$  n. b.

Reset : SPE 1 inaktiv

Bereich : 4000 ... 7FFF / E/A - Adr. B8 / B1

Schreibschutz f. 4000 ... 5FFF

SPE 1 bereits aktiv

Bereichsgr. 8KB

LD B, 40H (40... 5F)

LD C, 0B8H ~~mem~~ E/A - Adr.

OUT r r = beliebig

Bereich 1kB

LD B, 1FH (0FH)

LD A, 40H

MAR : OUT 0B8H E/A - Adr.

INC A (ADD 4)

DJNZ MAR

## Einstellung

4014 - Schalter	:	Anf. adr.	Speicherbereich
Wickel 1...4	(1)	Schlafsp. f. Speicher -15	(5P o. 5P6) 4214 4224
5...10	(3(2))	Sperrart	ohne Schreibschutz, Zugriffsschutz
11...22	(4)	E/A -Adr.	A <sub>4</sub> ...A <sub>7</sub>
23...38	(1)	E/A -Adr.	A <sub>1</sub> ...A <sub>3</sub>
39...50	(4)	Bereichsgröße	

## SPE 2

sperrten	:	/MEMDI	ges. Bereich
		/MEMDI 1	RAM - Ber.
		/MEMDI 2	ROM - Ber

sp.-programmierbar : - Zugriffs / Schreibschutz } durch Wickel  
1KBweise! - E/A - Adresse

A <sub>10</sub> ...A <sub>15</sub>	Festlegung	1k-Bereich
A <sub>1</sub> ...A <sub>7</sub>	E/A -Adr.	
A <sub>0</sub>	0 sperren	
	1 Freigabe	

- gesperrter Bereich wird mit aufgefriescht

## Modifizierung

RAM - Ber.	Anf. Adr. in 4k -Schritten	(4000...BFFF)
ROM - Ber.	" " " "	(2000...3FFF)
EPRom Auswahl	555 / 2716	(555)
E/A -Adr.	ROM : (B0/B1) Sp. Frei.	RAM : (B2/B3) Sp. Frei.
Anfangsinit.	(Freigabe beider Bereiche)	

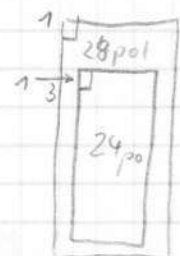
## SPE 4

- 24 Fassungen (28 pol.)

Typen	2716	2732	(2764 ...)
Kapazit.	48k	96k	192k

- Verwendung :

Nur -Lese -Speicher segmentierbar  
(MC 80 LA -LABS)



16 - Stromverbr. 5P 1,9 A (voll best.)  
 0,5 A (unbest.)

- Aufteilung / Bestückung:

	I	II	III	Bank
8				
7				
6				
5				
4				
3				
2				
1000 - 17FF	1			
	1000	5000	9000	
	4FFF	8FFF	CFFF	

- MEMDI - Sperren

- X1: C9 / MEMDI - Sperrt ges. Bereich
- X2: A21 / MEMDI 1 " Bank 1
- C21 / MEMDI 2 " Bank 2
- C20 / MEMDI 3 " Bank 3

E/A - Programmierung

- Zugriffssperren zur Segmentierung
- 3 / 6 Segmente 1 Wickel
- E/A Adressen A<sub>4</sub> ... A<sub>7</sub> 4 Wickel
- Banksteuerung 3 Wickel

Auswahl 3 Segmente 35 - 36

- X1 Segment 1 Bank I
- X2 " 2 " II
- X3 " 3 " III

Auswahl 6 Segmente 34 - 36

- X1. Segm. 1 Bank I u
- X2 " 2 II u
- X3 " 3 III u

X5	Segm.	4	Bank I <sub>0</sub>
X6	"	5	<u>II</u> <sub>0</sub>
X7	"	6	<u>III</u> <sub>0</sub>

X0/x4 deaktivieren

A<sub>3</sub> muß immer 0 sein!

Resetzustand: ~~Segm.~~ Bank I aktiviert!

Programmierung OUT n (A beliebig)

OUT r (C vorher laden)

Modifizierung

- 2 PROM ( 74 S 287 ) Adreßbereich (2K-Schritte)  
 ( 74 S 571 ) Bankfreigabe
- Wickel

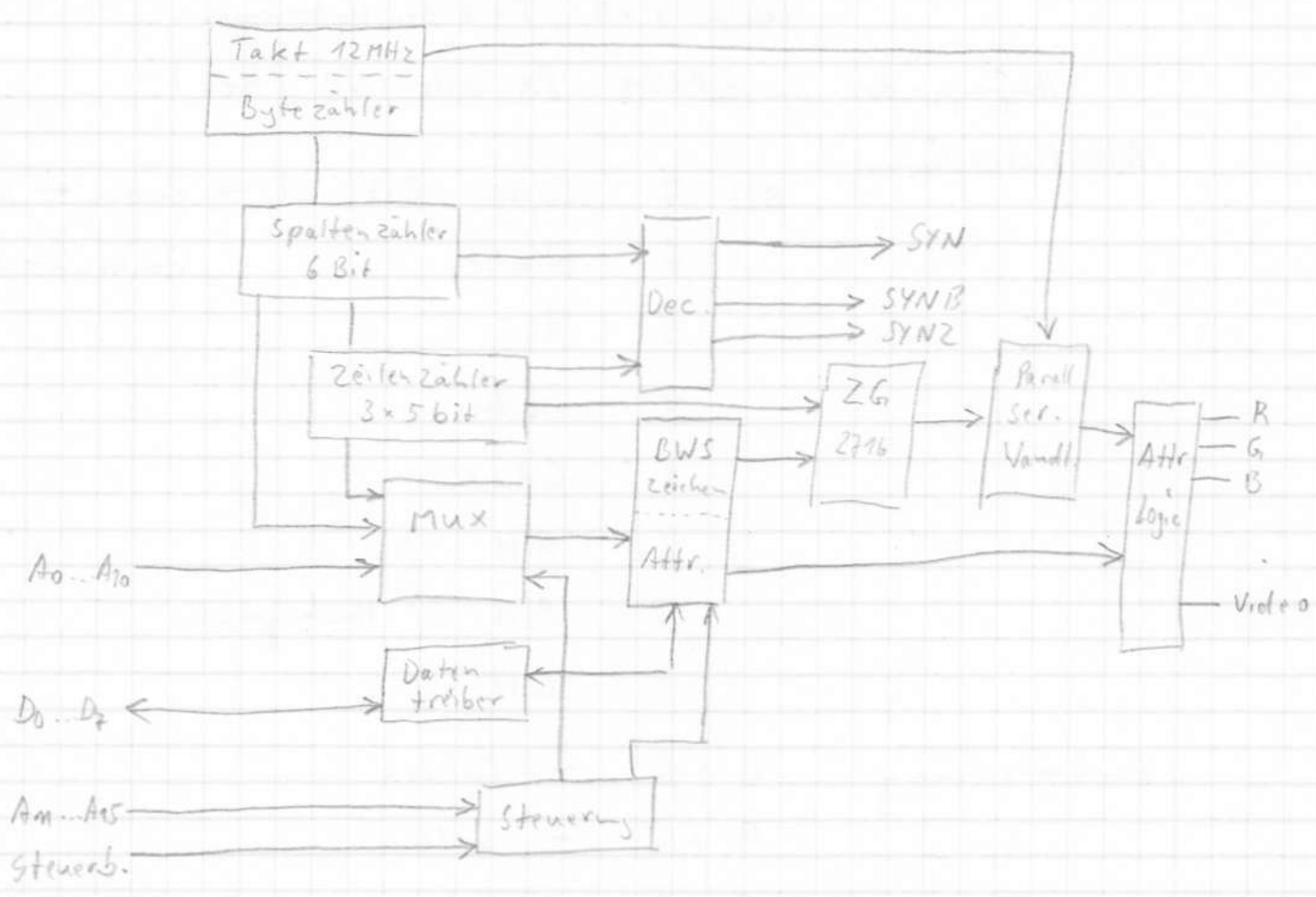
[ Aust. MON 2 mit DFA (A000 - AFFF)

• Dunkelsteuerung	A000 - A03F	1. Zeile
• Dest. des Zeilenanschlusses	A800 - AFFF	A+H.

]

# 17 QFA

- Farbmonitoransteuerung mit  $64 \times 32$  Zeichen und pro Zeichen  $8 \times 8$  pixel  
8 Vordergrund und Hintergrundfarben  
Cursor und Blinkdarstellung
- Bildwiederholer als Dual-Port-RAM (Zweitor)
  - 2 KB Zeichenspeicher 8x U214
  - 2 KB Attributspeicher (Farbe + Blinkgeschw.)
- Zeichengenerator für 256 Zeichen



- zeitliche Anforderungen

stehendes Bild  $f_{\text{Bildh}} = 50 \text{ Hz}$

Bildauslenk. Y	20 ms	312 Zeilen	
		256 Information	16,4 ms
		56 Dunkelast.	3,6 ms

Zeilen auslenk. X	64 $\mu\text{s}$	768 pixel	
		512 pixel Inform.	42,7 $\mu\text{s}$
		256 Dunkelast.	27,3 $\mu\text{s}$

Bildpunktfrequenz

83 ns

→ 6 MHz Videofrequenz

12 MHz Punktfr. (linear)

BWS - Zugriffe

- von Bildst. (Lesen)

- von CPU (Lesen + schreiben)

2 Modi:

- uneingeschr. Buszugriff  
(schnelles Schreiben u. Lesen)

- Buszugriff nur in Drehhaltungsperiode  
(sauberes Bild, WAIT)

Zeichencodierung: 256 versch. Codes - 8 x 8 pixel - Feld

00 - FF

20 - 7E ASCII - Codes

Zeich 5 x 7, 5 x 8 bei Unterlägen

Anfangsadr. in EPROM: 8x Code - Nr. (hex)

Attributcodierung:

Bit 7 Cursor - Bit 1: → Zeichen ↔ inv. Z.

6 Blinken Zeichen ↔ Leerz.

5 ROT

4 BLAU

3 GRÜN

2 RH

1 BH

0 GH

Hintergrund

Farbmischung

B + G - Türkis

R + G - Gelb

R + B - Purpur

R + B + G - Weiß

Einstellung d. QFA

- Anfangsadresse in 4 KB Schritten S1/1 .../14 A72 ... A75: X000 Zeichen  
X800 Attr.

- Zugriffsmodus S3

- Sperrung durch MEMDI S2

- Sperrung " MEMDI 1 S4/1  
2 12

18

- MEMDI - Signal-generierung SS
- Bei Adr.-zugriff auf QFA

Monitoranschlüsse:

- s/w - Monitor K 7222 (MON 2)

X4: 10 pol.

	A	B
1	5P	-
2	00	00
3	/SYN	-
4	-	00
5	VIDEO	-

- Farbmonit. K 7226.20 (MON 3 digital)

X3: 1 rot                   Anzahltrüber: 74538  
 2 grn.  
 3 bl.  
 4 sync

Koppelbus	A 21, C 21	MEMDI	} MC 80-LA
	C 15	/SYN	
	C 16	VIDEO	
	C 24	INTENS	
	A 15	/RT	} Farbmischung
	A 14	/GN	
	A 13	/BL	
	A 16	/RTH	
	C 14	/GNA	} Farbausgleichskarte → FBAS
	C 12	/BLH	
	A 10	/ROLL	} Rollen um 1 Zeile hoch
	C 4	INIT	

Weitere Anschl. für Vollgrafikspeicher



## VIS 2A:

- BWS nicht vom Bus erreichbar (nicht im Haupttyp.)
- wird über E/A-Adr. erreicht
- Vollgrafik 512 x 256 Punkte
- Frequenz = 12,5 MHz pro Zeile 800 pixel

### Programmierung:

Bildkommandos : Löschen, anstarten, rollen ↑/↓, ←/→,

Punkt kommandos : Setzen, löschen, invertieren

OUT r            r → Daten  
                  B → A8...15  
                  C → A0...7

C: b<sub>4</sub>...b<sub>7</sub> - Ste-Adresse (Cx) Adresse MC80.30  
      b<sub>3</sub> - Bildsektor (links, rechts)  
      b<sub>2</sub>...b<sub>0</sub> - Funktionscode

B: Y-Koordinate bzw. Opcode

r: X-Koordinate

### Bild aufbau:

Bildkommando dunkel      LD C, 0C4H  
                                  LD B, 00  
                                  OUT A

Punkt kommando's            LD A, 0  
                                  LD B, 0            B, FF  
                                  LD C, 0C1H  
XX: OUT A                    OUT B  
      INC A                    DJNZ XX  
      INC B  
      JRNZ XX

---

Bei Farbangeabe über 3 VIS 2A, siehe Anloge

### Programmierung:

an G6A : nur Bildkommandos  
an GEA : Bild- und Punkt kommandos

19

Farbmonitore:

- Bildröhren: Schlitzmaske

Typen: Diagonale 42 — 0,66 Pitchabstand unter Breite  
 51 — 0,82  
 67 — 0,82

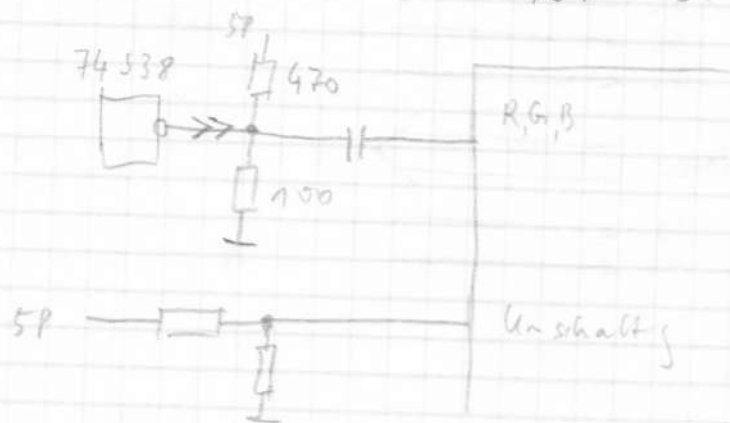
	Diagonale	Pitchabstand	unter Breite	Anz. Spalten
	42	0,66	32 cm	480
	51	0,82	40 cm	488
	67	0,82	52 cm	658
in Entw.	51 (20")	0,47	(640 x 480 pixel)	
	(14")	0,31		
Weltspitze		0,31		
		0,21	(1024 x 1024)	

- Bildempfehlung im Fernsehgerät

- über HF - Eingang (PAL - Dek.)
- über FBAS ( " " )
- über RGB - Eing.

- Nachrüstung: Serie 4000 o. Robotron

Lit.: RFE 1/87 S. 31-34



SYN → → FBAS

- Mon 3 K722G

Variante

- 10

- 20

digit. E.

- 11

- 21

anal. E.

Diagonal: 51 cm

Einbau

Auftrag

Bildfeld: 250 x 360 mm

Punktfeld: 448 x 288 pixel

64 x 32 Zeichen mit 7 x 9 Pixelraster

Bild 50 - 60 Hz

Linie 60 - 64  $\mu$ s

Gesch. Leitungen < 100 m

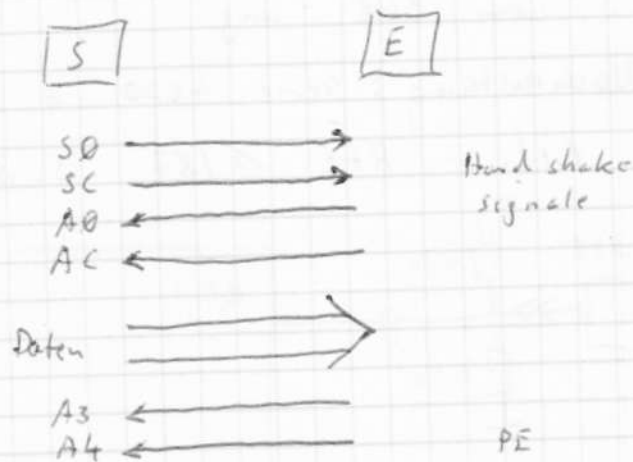
## Geräteanforderungen:

- Verarbeitung / Transfer (DMA, Arithmetik, Busverstärker, Emulation, Datenrettung, Speicherkapazität)
- Arbeitsspeicher / RAM, ROM
- Massenspeicher / KMG, Floppy AMF + 1.6 (AFS 2 + 1.6 f. CSOS)
- Ein- / Ausgabe / Zusatzmonitore, Signal-E/A, Datenübertragung
  - ↳ Datenprüfung: - Parität
  - CRC-Prüfsumme
  - Echolieren

## Schnittstellen:

parallele: IFSP

- TTL-Pegel
- $I_{aus} \geq 40mA$
- $I_{ein} \leq 1,6mA$
- Länge 15m



## Centronics

serielle: IFSS asynchron (IFLS synchron)  
V24 (RS 232)

IFSS: 20mA    0  $\hat{=}$  0...3mA  
1  $\hat{=}$  15...25mA

Länge: 500m (9600 Baud)

V24: 2 Datenleitungen, > 3 Steuerleitungen,  $\leq 30m$   
Pegel: 0  $\hat{=}$  3...15V    1  $\hat{=}$  -2...-15V

LD IX, (F900)

CALL 12A1 Dir schreib

1371 Reserv. ein

1378 " aus

1380 Rückspule

137C 1 Block lese (DF311 H steht)

13A5 Bandmaße schreibe nach jed. Block

13A9 Schließlücke z.B. Band lösen

13AD Block schreibe BC - Blocklänge  
HL - Blockanfang

1397 Bandmaße mehr vorwärts A = Anz. Bandm.

139E " " nächster

13BB A = 11H 1 Block vorwärts

A = 15H 1 " nächster

• Kommando puffer anlegen

~~WRITE~~ name, ... Parameter FF

DE auf Anfang KOPU stelle und CALL (WRITE) + xx  
echter Anfang

**CALL XXXX**

Addr.

• Doppelt große Schrift:

LD A, 11H groß LD A, 1 klein

LD (DFB42), A

CALL 0BD9H

Folgevektor (bel. Adr.)	+ 0	0	Lesen
		2	Schreiben
	+ 1	LW	Wc 14.2
	+ 2	Länge	Name
	ab + 3		Name
	:		
	FF		

Aktivität	OPEN	}	Datei
	WRITE		
	CLOSE		
	OPEN	}	Dir
	WRITE		
	CLOSE		

OPEN REQ. (damit Reserv. ein)

Band rückspulen

Res. aus

Dir. schreiben

Res. ein

Rückspulen

Res. aus

Close Req.