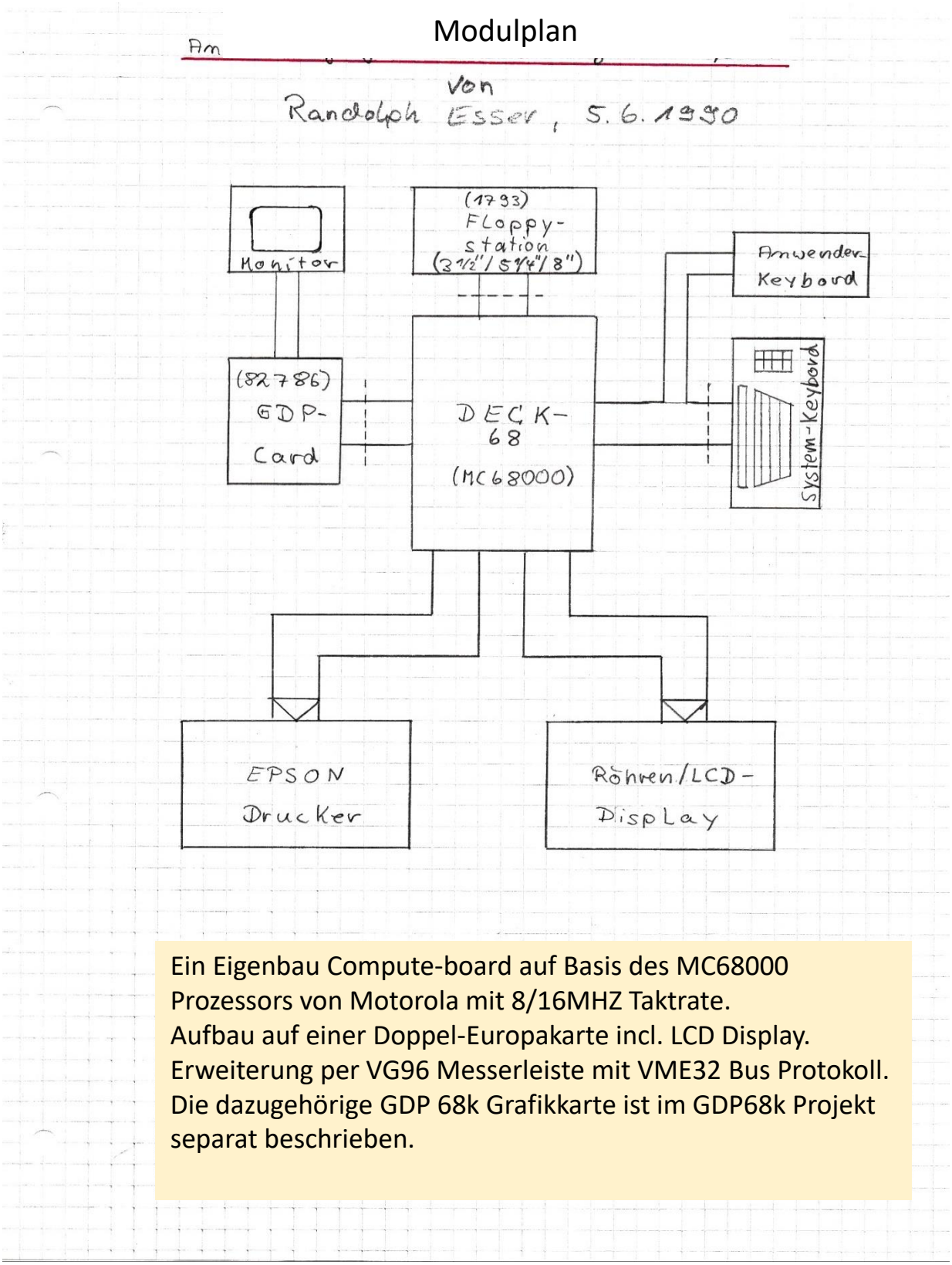
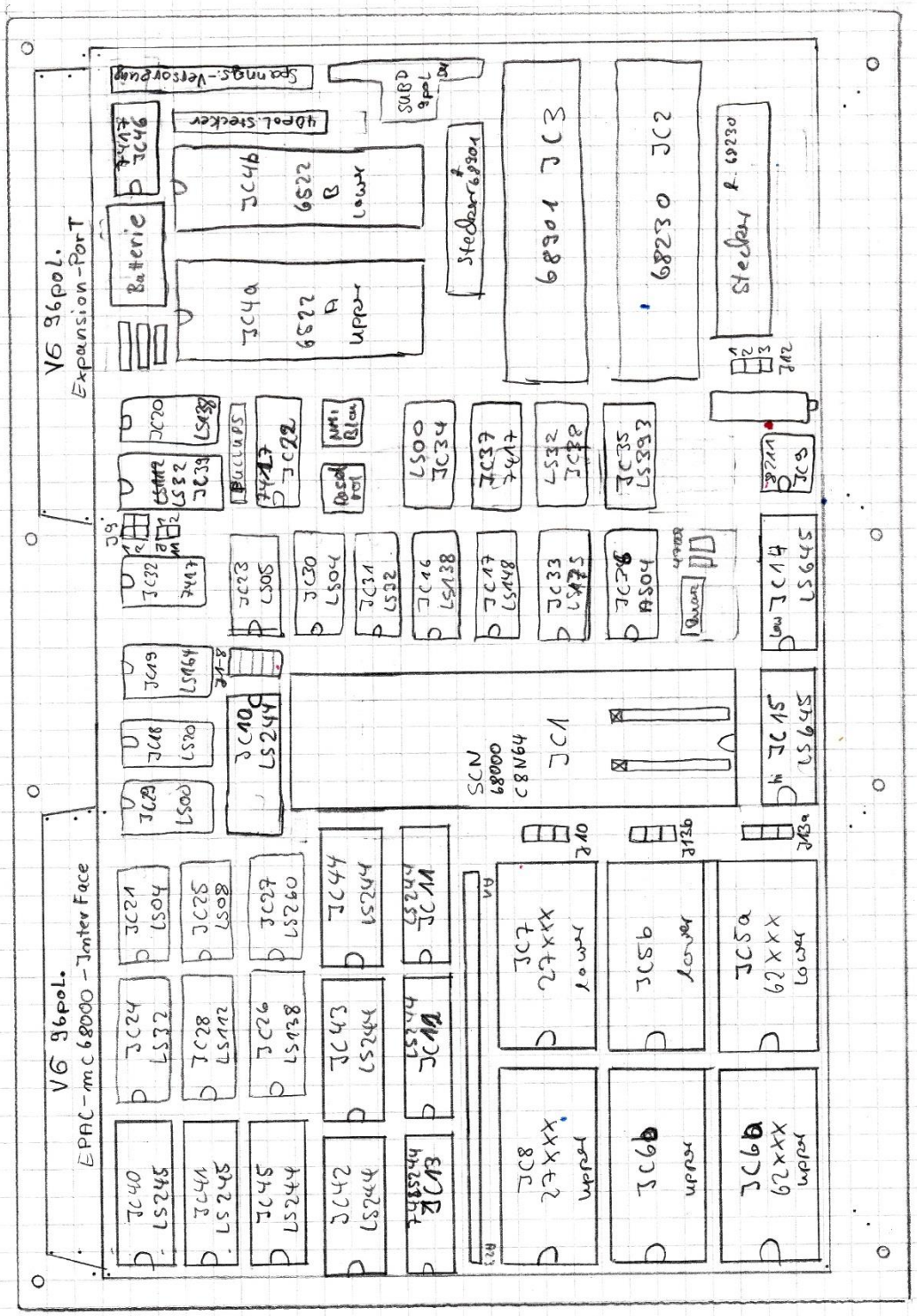
DECK-68k Compute Board

Modulplan

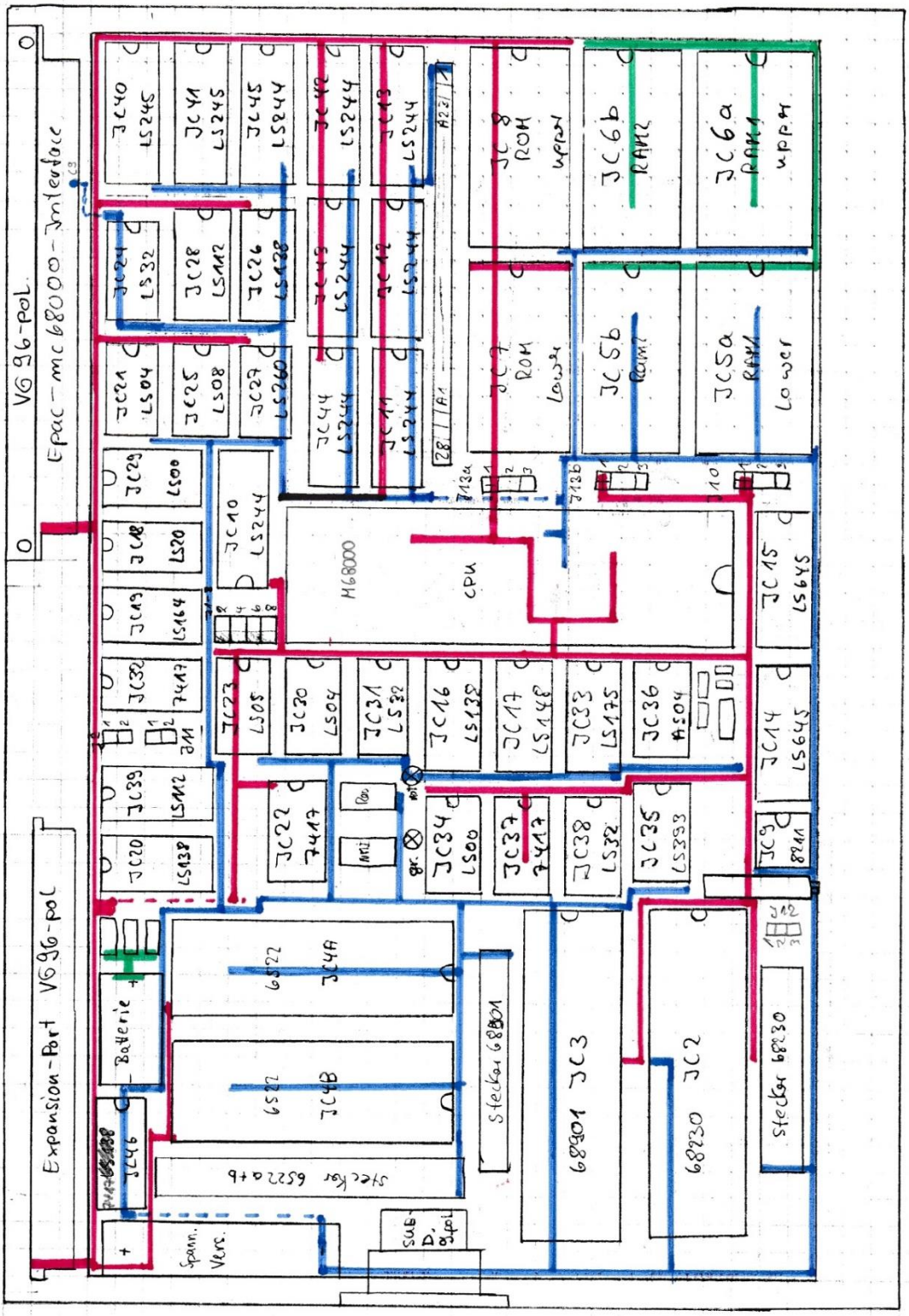


Ein Eigenbau Compute-board auf Basis des MC68000 Prozessors von Motorola mit 8/16MHZ Taktrate.
Aufbau auf einer Doppel-Europakarte incl. LCD Display.
Erweiterung per VG96 Messerleiste mit VME32 Bus Protokoll.
Die dazugehörige GDP 68k Grafikkarte ist im GDP68k Projekt separat beschrieben.

Bestückungsseite - Bauteilplan

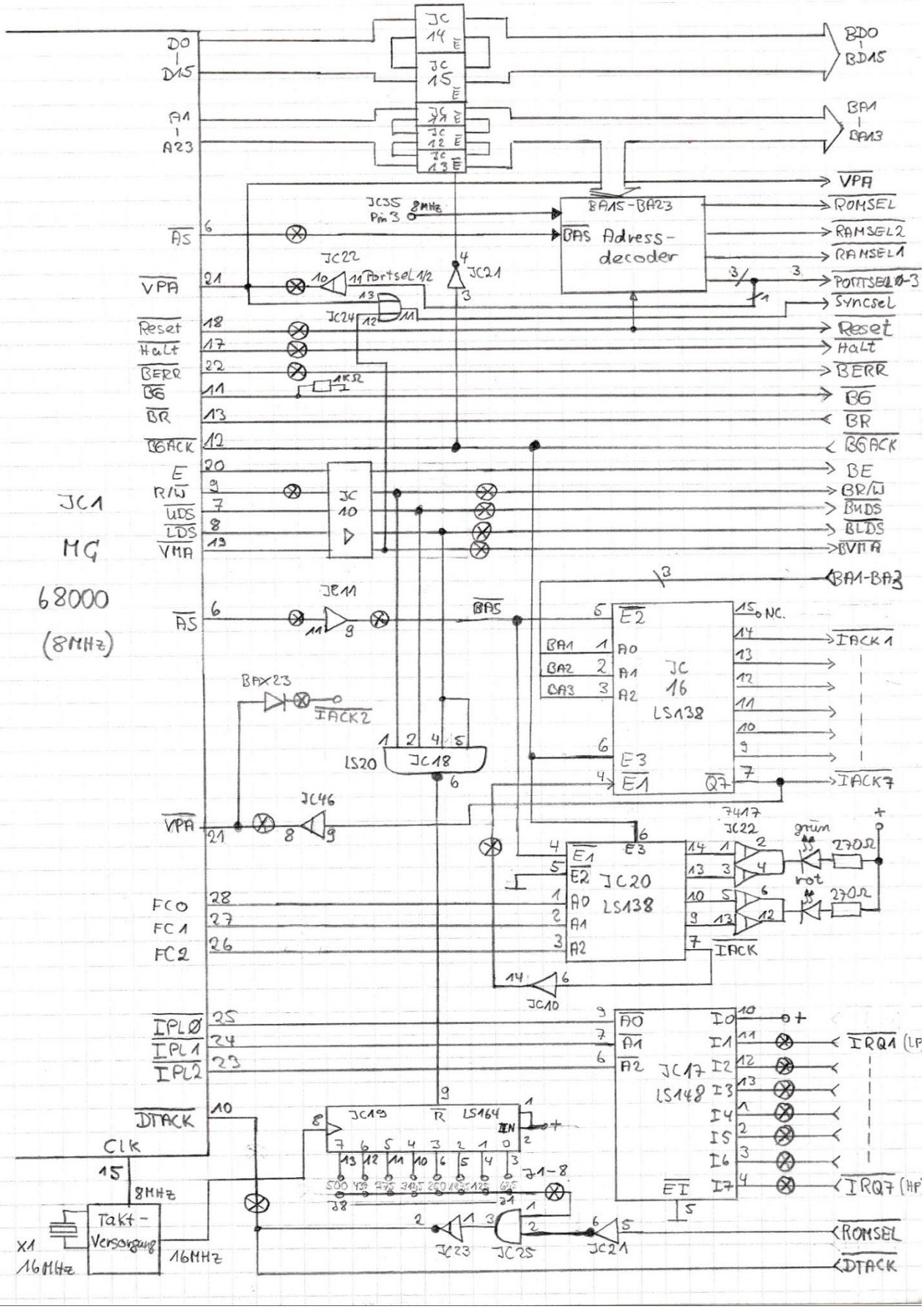


Lötseite - Bauteilplan

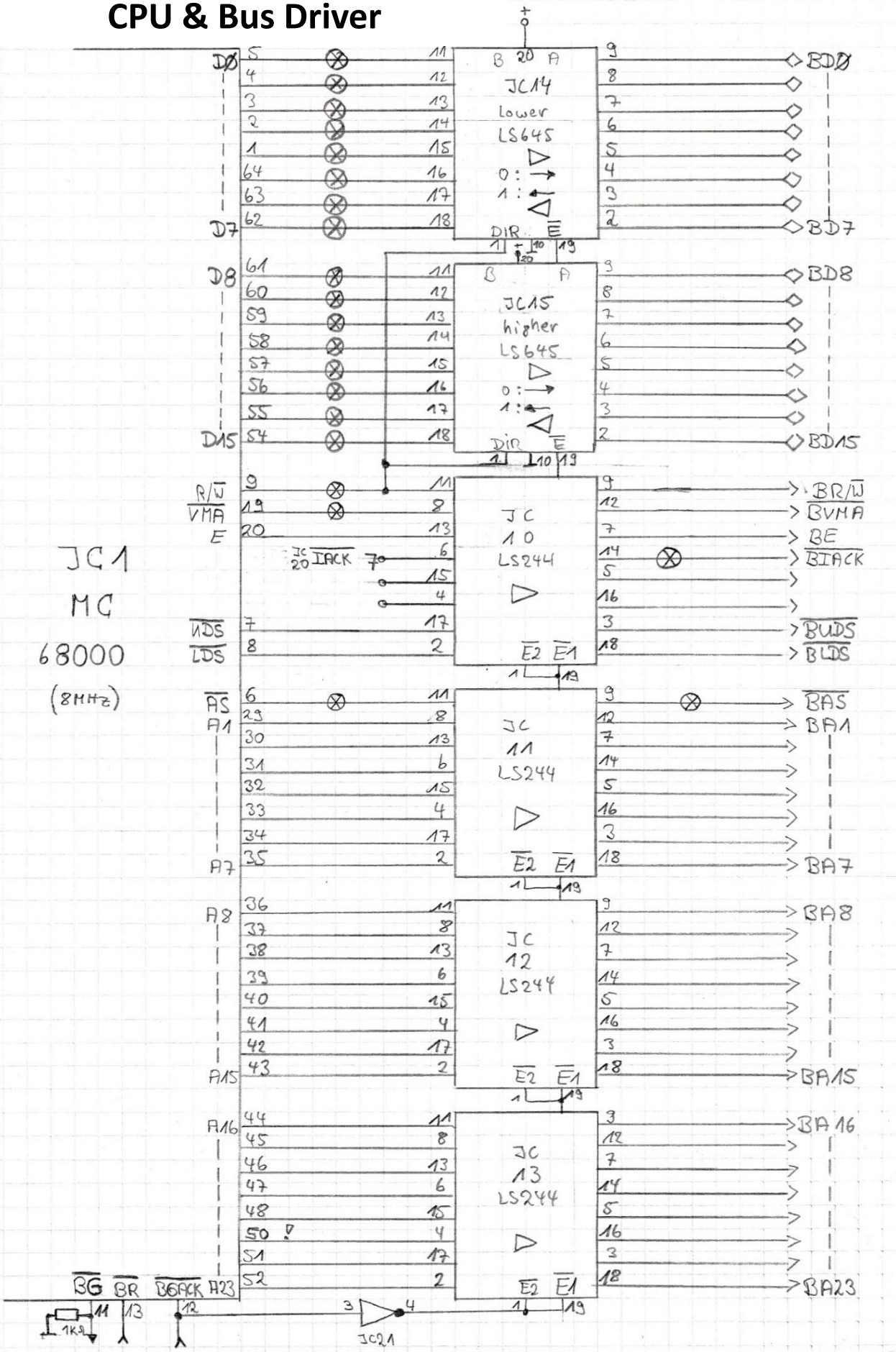


Rot = +5V Leitung
Blau = Gnd Leitung
Grün = Standby Power (Batterie gestützt)

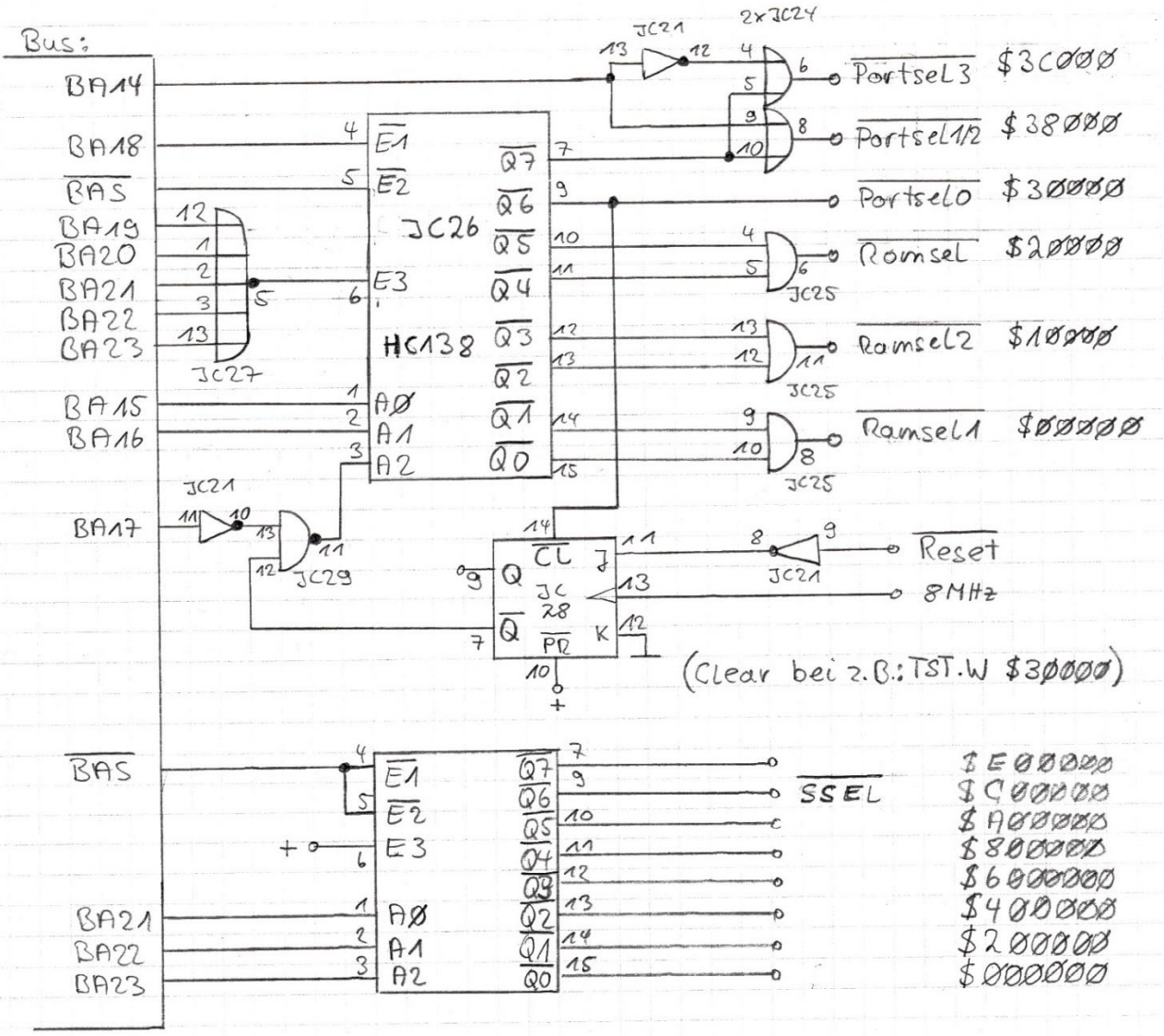
CPU & IRQ Logik



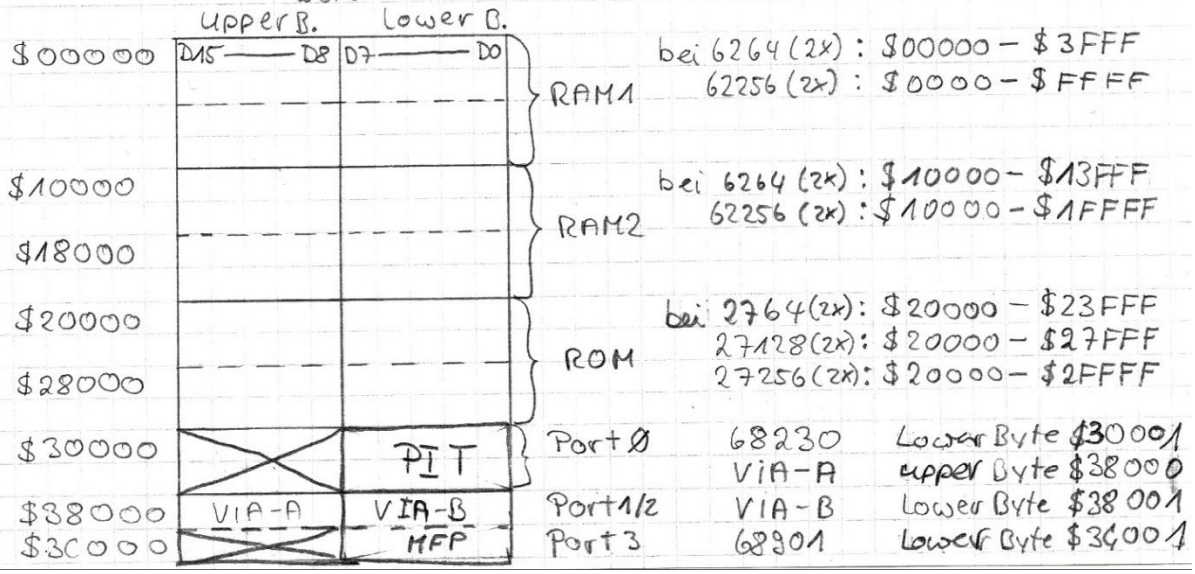
CPU & Bus Driver



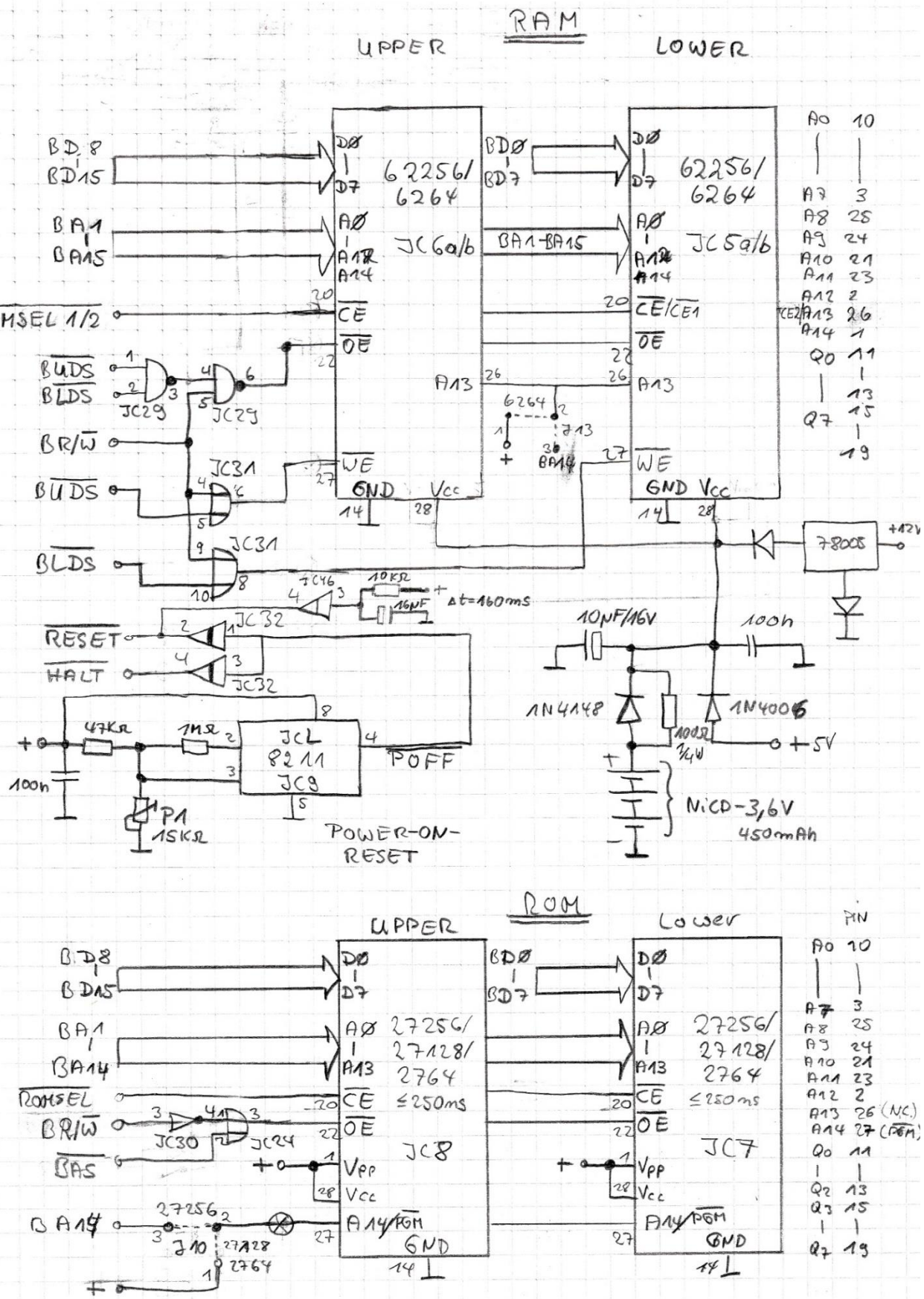
Adressdekoder



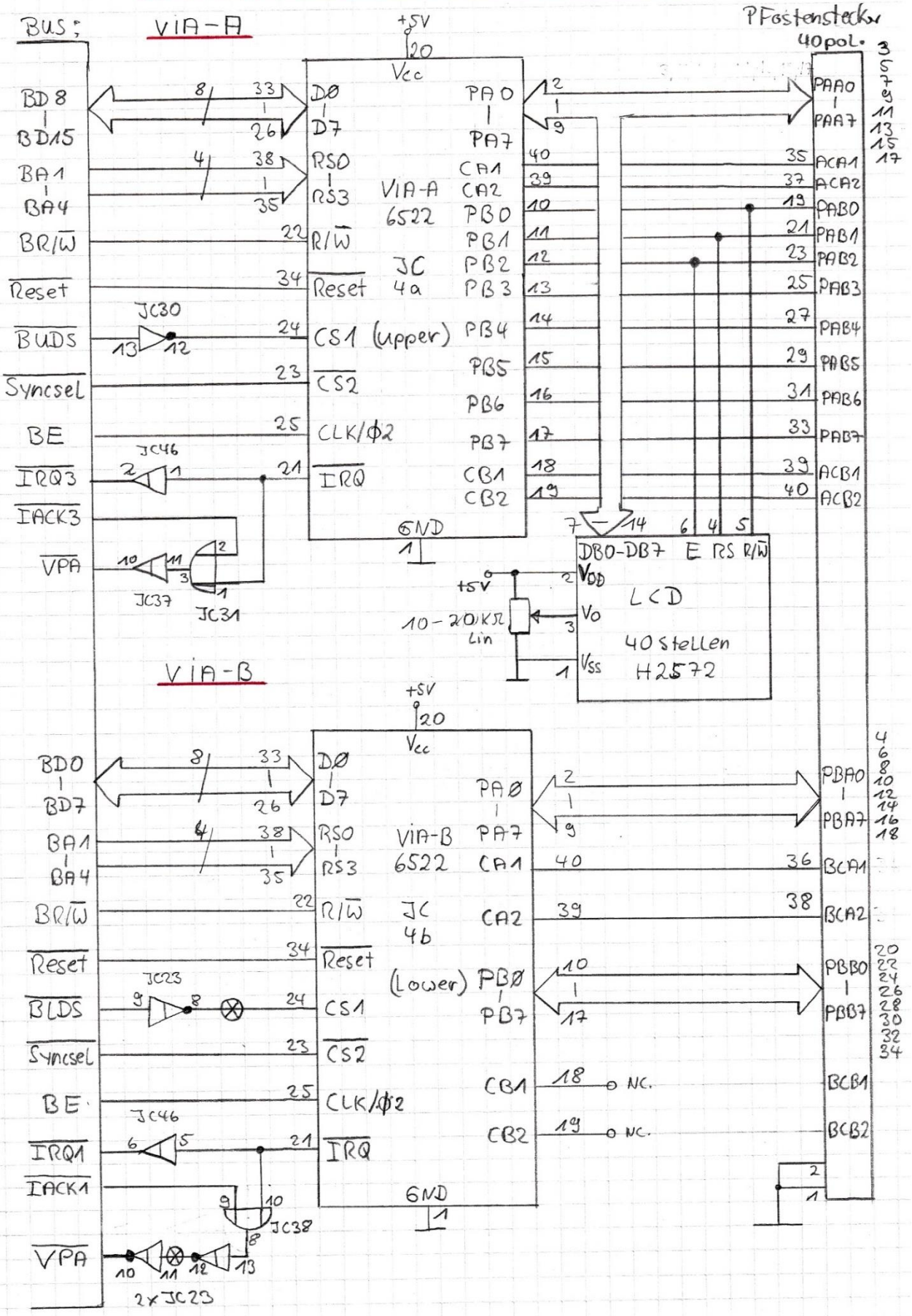
Speicheraufteilung (16-Bit-Speicher!)



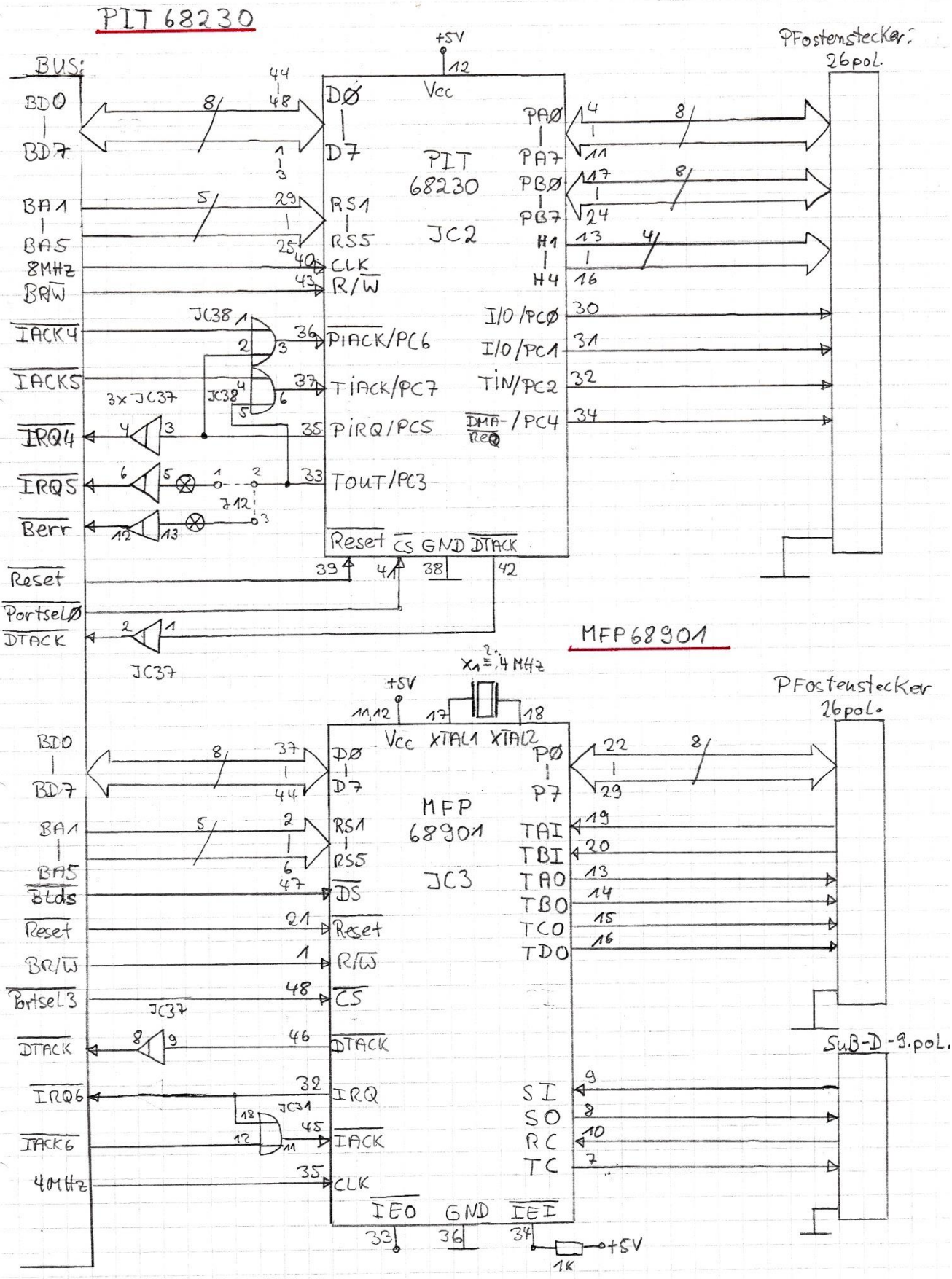
Ram/ROM Address Mapping



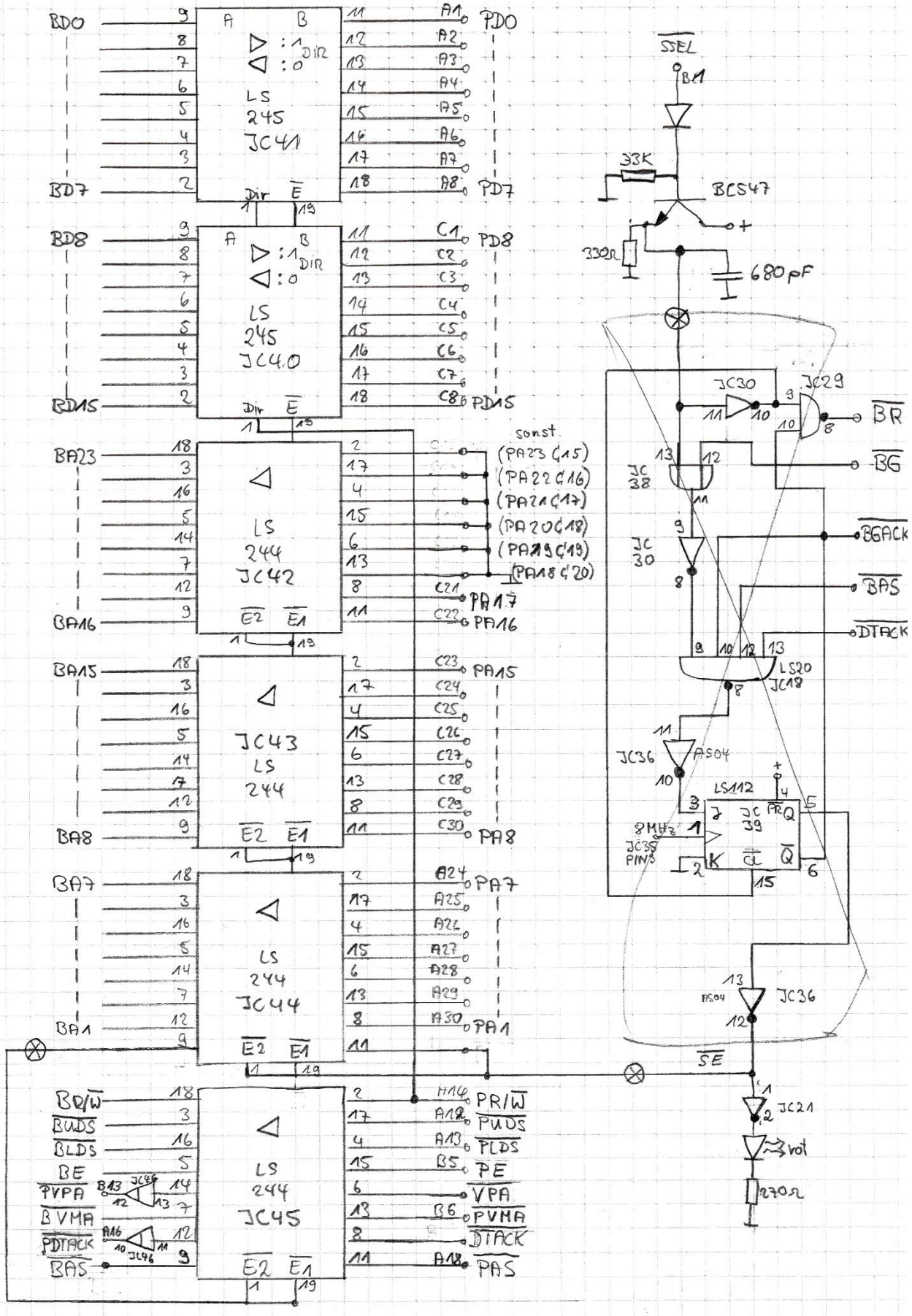
Peripherie-IO Prozessoren (VIA's)



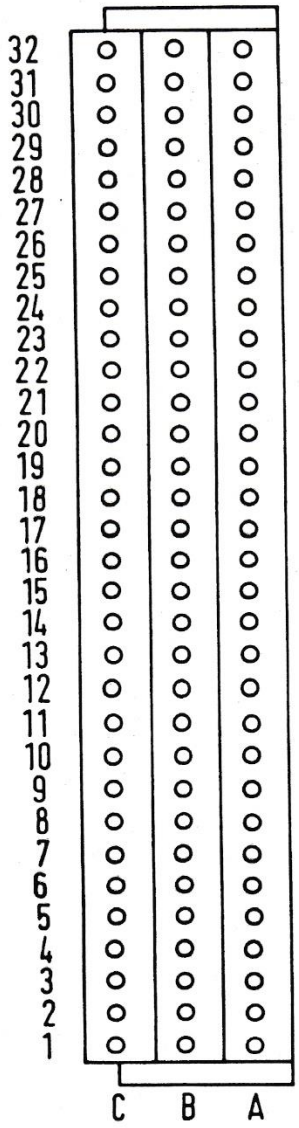
Peripherie Prozessoren -> IO Ports



EPAC mc68000 PigyBack Interface



VG Expansion Board Interface



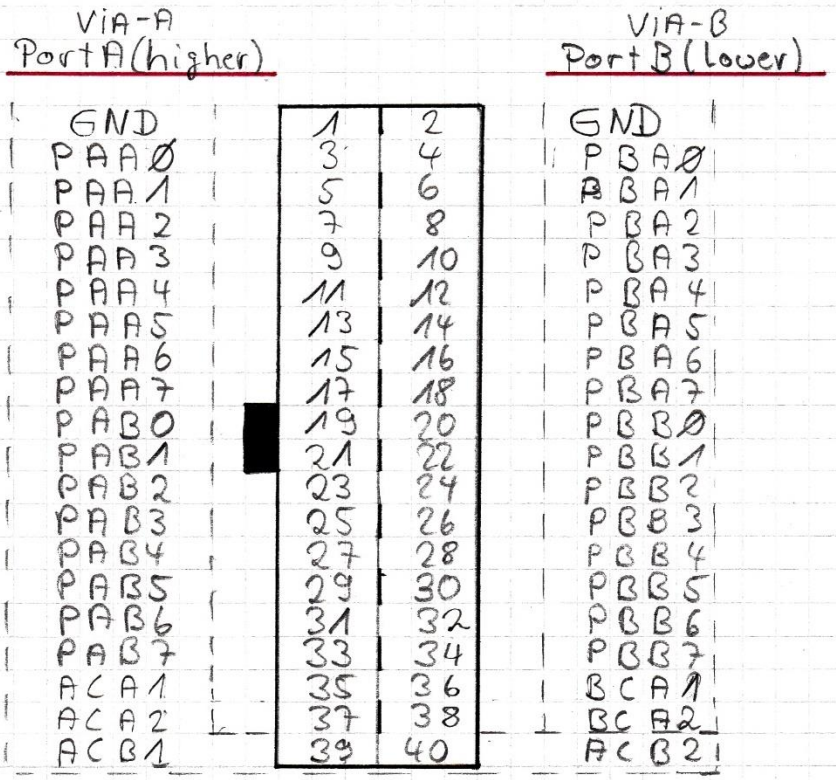
96pol. VG-Leiste

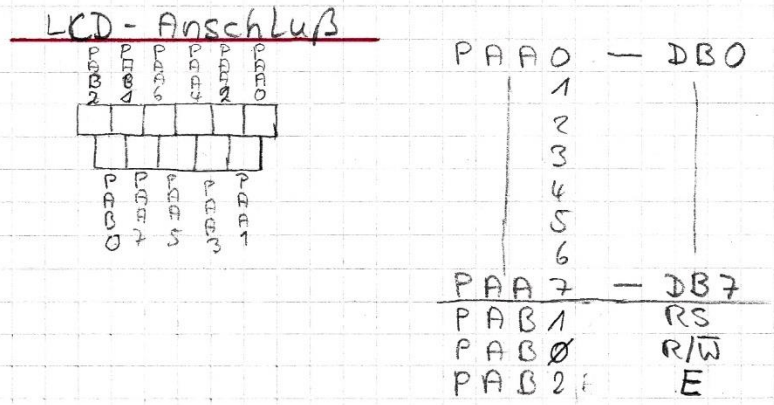
Reihe C	Reihe B	Reihe A
+5V Stromvers.	+5V Stromvers.	+5V Stromvers.
+12V Stromvers.	+5V UCMOS	-12V Stromvers.
BA8	IRQ1	BA1
BA9	IRQ2	BA2
BA10	IRQ3	BA3
BA11	IRQ4	BA4
BA12	IRQ5	BA5
BA13	IRQ6	BA6
BA14	IRQ7	BA7
BA15	GND	EXP
BA16	FLOPPY	FC2
BA17	NC	FC1
BA18	GND	FC0
BA19	+12V Stromvers.	GND
BA20	+12V Stromvers.	BAS
BA21	GND	GND
BA22	NC	DTACK
BA23	NC	GND
GND	LPSTB	BR/W
NC	VPA	BLDS
RESET	BR	BUDS
BERR	EXT	GND
HALT	NC	16 MHz
GND	POR	GND
BD15	NC	BD7
BD14	BR/W	BD6
BD13	BVMA	BD5
BD12	BE	BD4
BD11	BG	BD3
BD10	NC	BD2
BD9	BGACK	BD1
BD8	SSEL	BD0

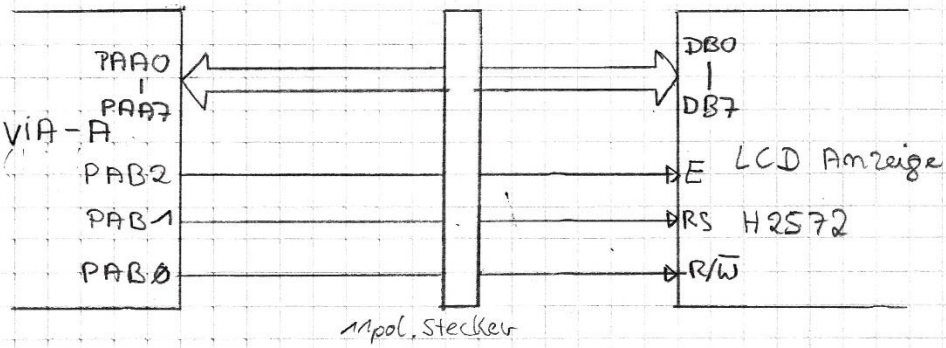
Alle mit NC (no connection) gekennzeichneten Leitungen sind mit 1kOhm-Pullup's versehen.

1

VIA IO Port Pinning

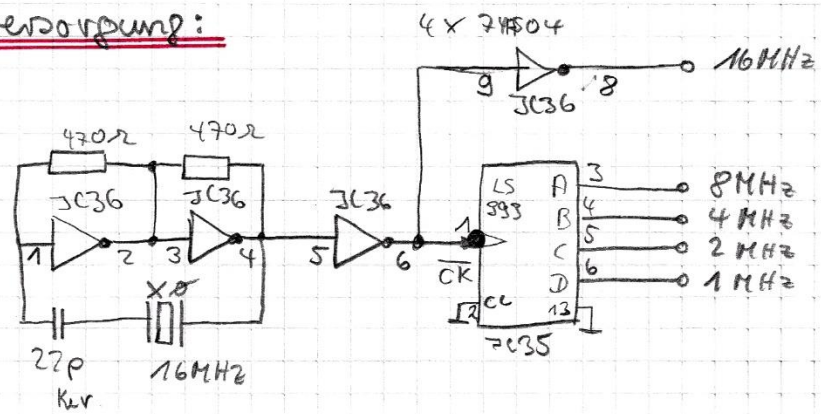






Pins Eingang	Typ	Pins Ausgang	Bezeichnung / Art	pro Gehäuse vorhanden	benötigt
1,3,5,13,11,9		2,4,6,12,10,8	Inverter. 74LS04	6	12/6 ←
1,3,5,13,11,9		2,4,6,12,10,8	Inv. o.K. 74LS05	6	6 ✓
1,3,5,13,11,9		2,4,6,12,10,8	Treiber o.K 74LS17	6	18 ✓
2,4,6,8,17,15,13,11		18,16,14,12,3,5,7,9	Tristate|unidir. 74LS244/41 Bustreiber	8	30 ✓
2-9	A B	18-11	Tristate|bidir. 74LS245	8	16 ✓
1 4 13 9 / 2 5 12 10		3,6,11,8	AND 74LS08	4	4 ✓
1 4 13 9 / 2 5 12 10		3,6,11,8	AND o.K. 74LS09	4	0 ✓
1 4 13 9 / 2 5 12 10		3,6,11,8	NAND 74LS00	4	8 ✓
2 5 12 9 / 3 6 11 8		1,4,13,10	NAND o.K. 74LS01	4	0
1 13 / 2 12 / 4 10 / 5 9		6,8	4-fach NAND 74LS20	2	2 ✓
1 4 / 2 11 / 3 10 / 13 9 / 12 8		5,6	5-fach-NOR 74LS260	2	1 ✓
1 4 13 9 / 2 5 12 10		3,6,11,8	OR 74LS32	4	12 ✓
2 5 12 9 / 3 6 11 8		1,4,13,10	NOR 74LS02	4	0 ✓

Taktversorgung:

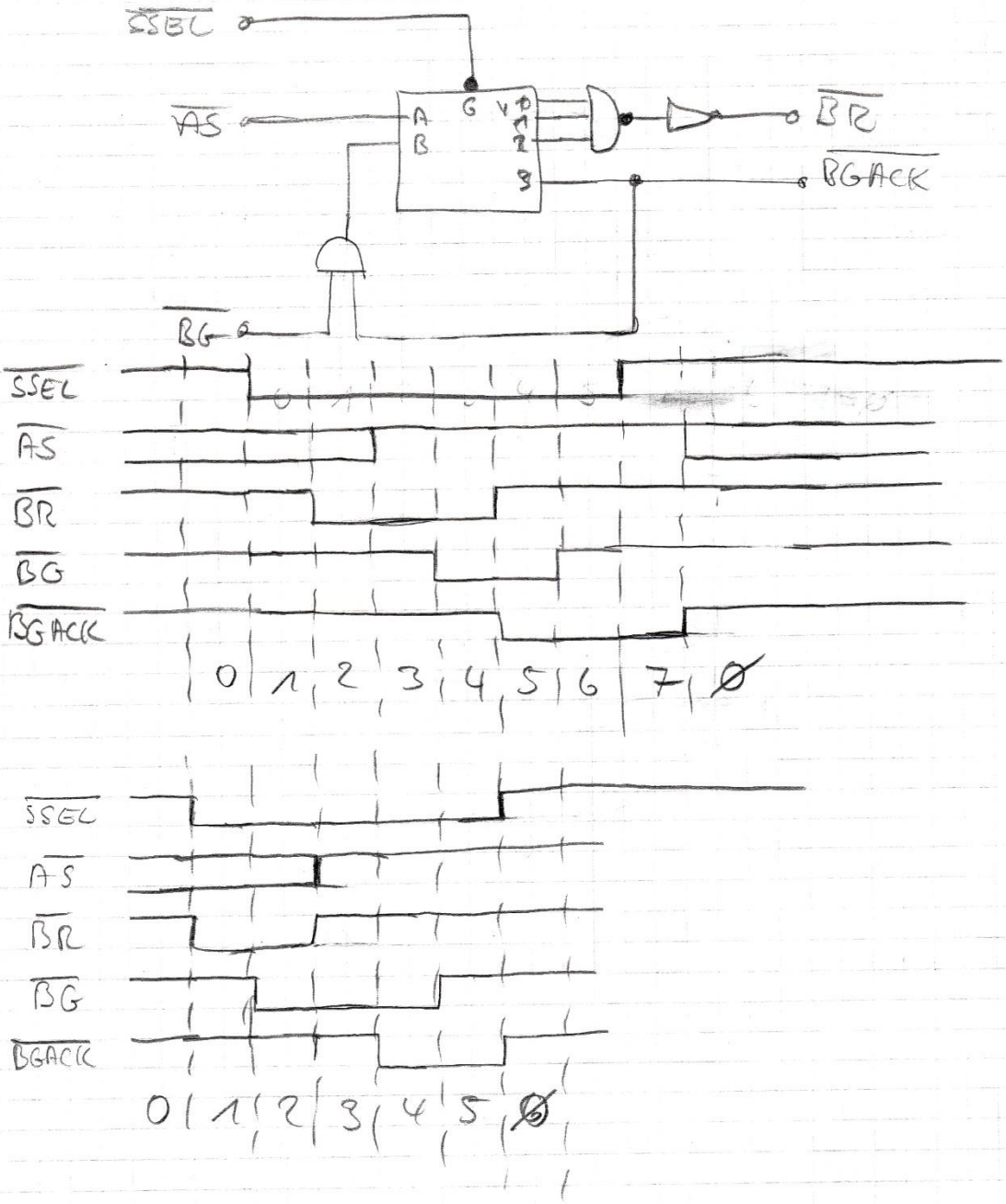


Bauteile-Liste (CPU-Platine)

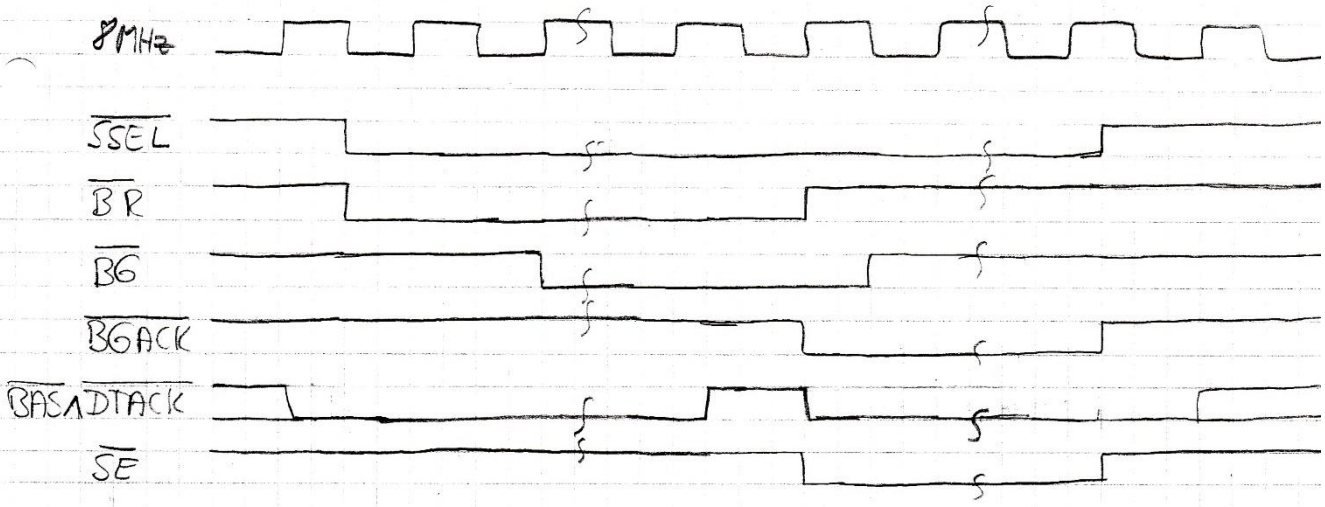
Menge	Bezeichnung	Pin-Gehäuse	
1	MC 68000 8MHz	64	IC 1
1	MC 68230 8MHz	48	IC 2
1	MC 68901 4MHz	48	IC 3
1	MM 6522A 2MHz	40	IC 4
2	MM 62256/120ns	28	IC 5,6
2	27256/250ns	28	IC 7,8
1	ICL 8211	8	IC 9
8	74LS244	20	IC 10,11,12,13,42,43,44,45
4	74ALS645/245	20	IC 14,15,40,41
3	74LS138	16	IC 26,20,16
1	74LS148	16	IC 17
1	74LS393	14	IC 35
1	74LS164	14	IC 19
2	74LS112	16	IC 28, IC 39
1	74LS175 (HC)	16	IC 33
1	74LS260	14	IC 27
1	74LS20	14	IC 18
3	74LS32	14	IC 24,31,38
3	7417	14	IC 1,22,32,37
2	74LS04	14	IC 21,30
1	74AS04	14	IC 36
1	74LS05	14	IC 23
1	74LS08	14	IC 25
2	74LS00	14	IC 1,29,34

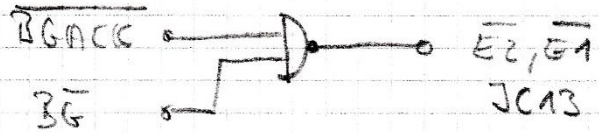
1	Sockel: 8 pol
17	14 pol
7	16 pol
6	20 pol
4	28 pol
1	40 pol
2	48 pol

2	Buchsenleiste 32 pol.
1	Quarz 16 MHz
1	12 MHz
5	Jumper
1	Spindeltrimmer 15 KΩ/1%
1	Widerstand 47 KΩ/1%
1	1 MΩ/1%
1	Diode 1N4148
2	LED rot
1	LED grün
1	1N4006
1	Batterie Lithium 3V/450mAh
23	Pullups 10 KΩ
40	Abblock-Kondensatoren
	Kondensator Elko 10µF/16V

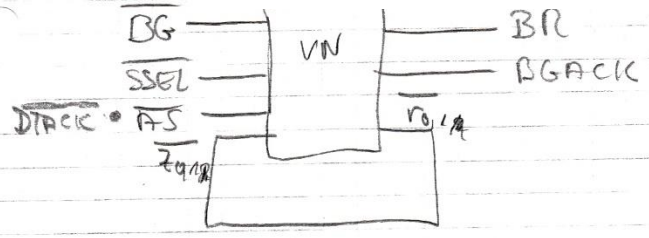

SSEL
AS
A
B
G
V
0
1
2
3
BR
BGACK
BG
SSEL
AS
BR
BG
BGACK
0 1 2 3 4 5 6 7 Ø
SSEL
AS
BR
BG
BGACK
0 1 2 3 4 5 Ø

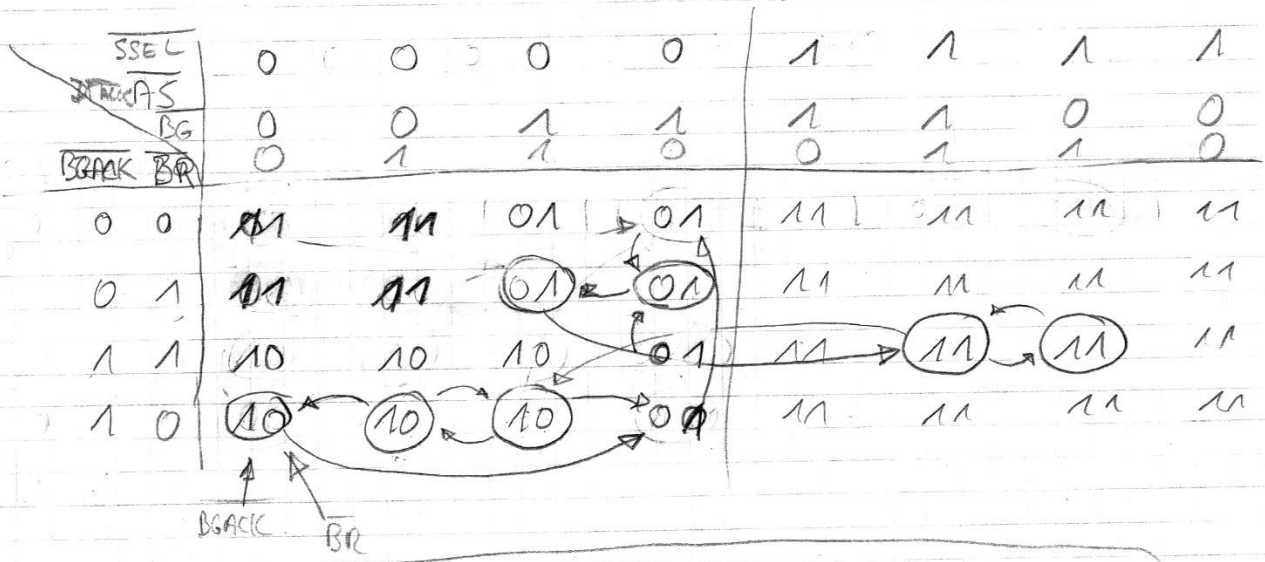
Bus-Übergabe zum mc68000-Interface





POC – M68k PigyBack memory handshake logic

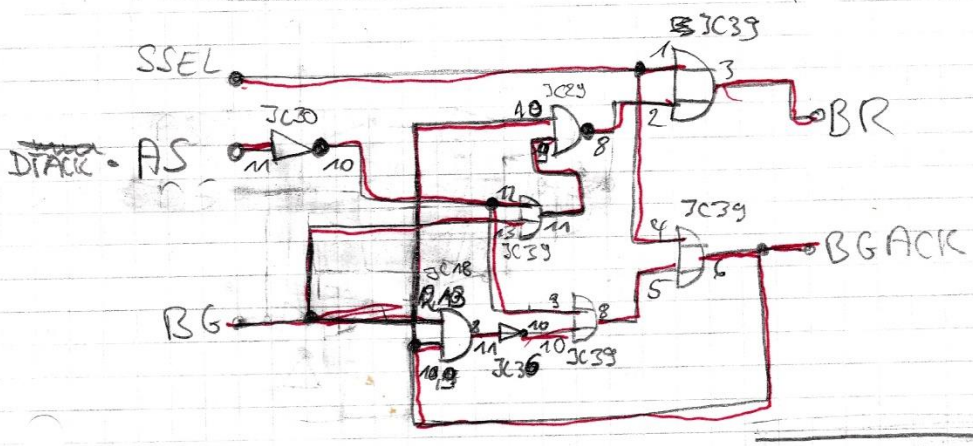




$BR = \overline{BGACK} \cdot \overline{AS} \vee \overline{BG} \cdot AS \cdot \overline{BG} \vee SSEL$
$= AS \cdot (\overline{BGACK} \vee \overline{BG}) \vee SSEL$
$AS \cdot (\overline{BGACK \cdot BG}) \vee SSEL$

$BGACK: \overline{AS} \vee (BG \cdot BGACK) \vee SSEL$

$BR = AS \cdot (BGACK \vee (\overline{SSEL \cdot BG})) \vee SSEL$
$= AS \cdot (\overline{BGACK} \cdot SSEL \vee BGACK \cdot \overline{BG}) \vee SSEL$



$BR = \overline{BGACK \vee (AS \cdot \overline{BG})} \vee SSEL \triangleq \overline{BGACK \cdot (\overline{AS} \vee BG)} \vee SSEL$

$BGACK = \overline{AS} \vee (BG \cdot BGACK) \vee SSEL$

CP/M BIOS Interface Specification

Bios-Aufruf TRAP #3

```
trap hndl:
        cmpi    #nfuncs,d0
        bcc     trapng
        lsl     #2,d0
        lea     biosbase,a0
        movea.l 0(a0,d0.w),a0
        clr.l   d0
        jsr     (a0)
trapng: RTE

biosbase  dc.l  _init
                wboot
                constat
                conin
                conout
                lstout
                pun
                rdr
                home
                seldsk
                settrk
                setsec
                setdma
                cread
                cwrite
                liststt
                sectran
                sofunk      (Bios 17)
                getseg
                getiob
                setiob
                flush
                setexc

nfuncs  EQU     (*-biosbase/4)
```

Bios head:
Systemadressen

```
fdccmd    equ     /Floppybase/
fdctrk    equ     fdccmd+2
fdsec     equ     fdccmd+4
fdcdat    equ     fdccmd+6
fdcst     equ     fdccmd+8

gdpbase   equ
gdpreg    equ

floadrg   equ     $42       Vector f. FLoppy
intdef.   equ     $FFFFFD   Autovec. f. Int Nr. 6

copyfop   equ               String mit Endo
der_out   equ               Der Fehler D7
```

```
ctrLst:     tst.b   ctrstat      'Lstrdy'-Flag -> Status
            bmi     Lstrdy
            RTS                 'not ready'
Lstrdy:     move.L  cbufsz,D2    Buffer-Länge
            cmp.L   cbufcn,D2    Druckerzähler.
            bne     LstOK
            rts                 'not ready'
LstOK       move.W  #$FF,D0
            RTS
```

Disk-Rout.

```
home:       clr.b   SekTrK      Spur 0
            RTS
Seldsk:
seltr0:
seltr1:
SetTrK:
setsec
sectran
setdma
flush
noflush
getseg:     move.L  #memrgn,D0
            RTS
getiob:     move.B  iobyte,D0
            RTS
setiob:     move.b  D1,iobyte
            RTS
setexc:     andi.L  #$FF,D1
            LSL     #2,D1
            movea.L D1,A0
            move.L  (a0),D0
            move.L  d2,(a0)
noset:      RTS
```

```
soFunk      cmpi    #sfuncs,D2      Sonderfunktionen
            bcc     soung           Funktion n. vorhande
            LSL     #2,D2           Bios Funkt. *4
            Lea     soudbase,a0     Bios Sprung-Tafel
            movea.L 0(a0,d2.w),A0   Adresse der Funktion
            jmp     (a0)
soung       RTS

soundbase   dc.L    SetsHL
            dc.L    baudrt
            dc.L    setstep
            dc.L    LCD
            dc.L    mc68K
            dc.L    EPSON
            dc.L    Display
sfuncs      EQU     (*-soudbase)/4
```

↓

- Disk-Sonderfunktionen !?
- Daten-Definitionen

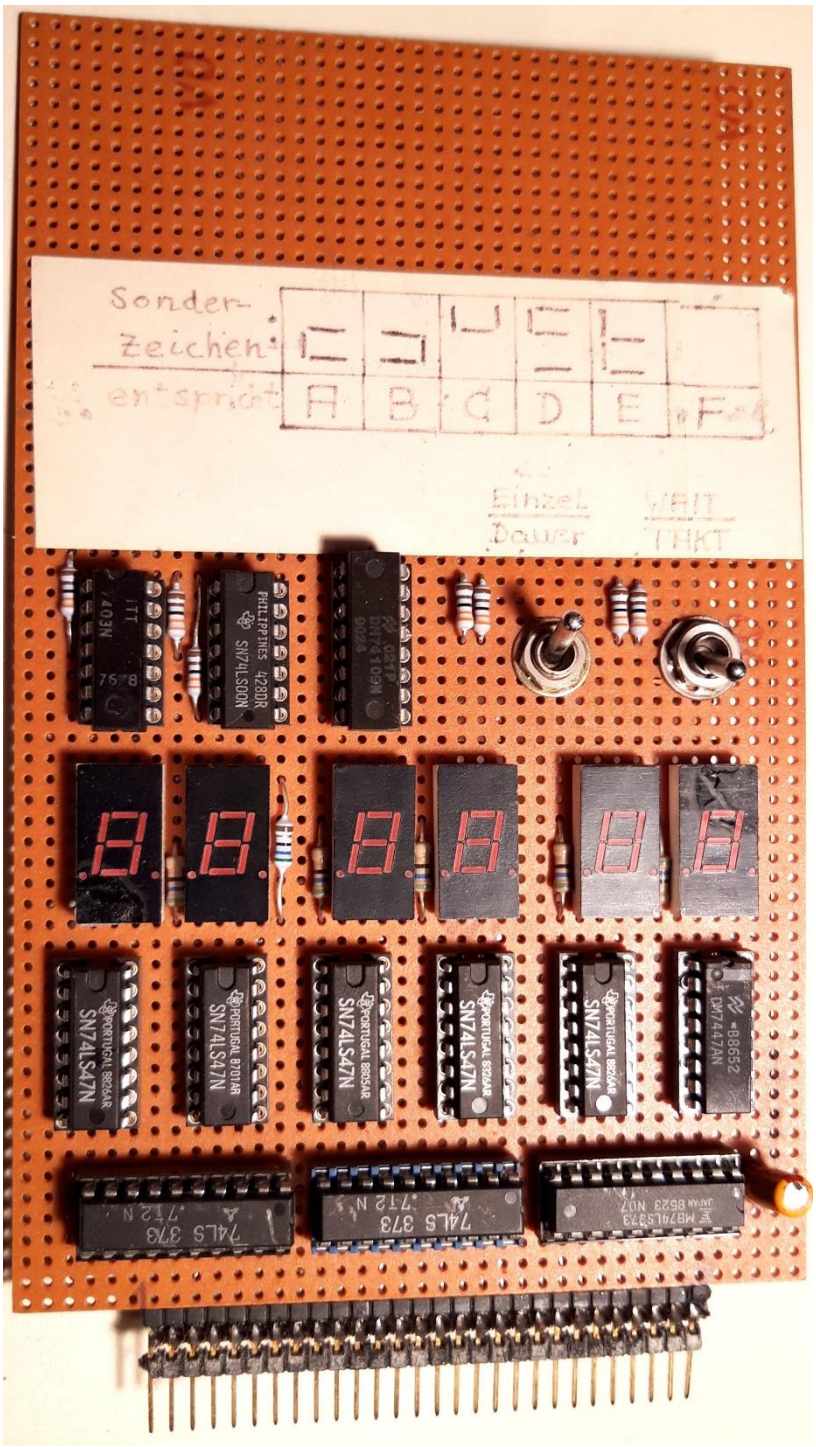
Im Supervisor-Modus schalten:

```
MOVE.L    (A7)+, Supmem          Stack inhalt retten
MOVE.L    $20, Supmem+4          Sprung für Privileg verl. retten.

PEA       SUPERV1
MOVE.L    (A7)+, $20
Move      #$2000,SR
Move.L    (Supmem+4),$20
Move.W    #$4EF9, Supmem-2       ; JMP vor Rücksprungadresse setzen -> zurück zum alten PRG
JMP       (Supmem-2)

SUPERV1   Move.W    #$2000,(A7)   neue Routine bei Privileg-
          RTE                     verletzungen.
```

Address Monitor Debug Board – TOP Side



Wird auf die Pfostenleiste parallel zu den DRams aufgesteckt.
Über die Kippschalter kann Einzeltakt Bearbeitung zum Debugging
Des Bootstrap Codes erreicht werden (ab Addresse 0x0).

Address Monitor Debug-Board – Buttom Side