

$\rightarrow$ D	74H00 - 10	MONOF	74921 - 5
$\rightarrow$ D	7400 - 87	MONOF	74123 - 4
$\rightarrow$ D	7404 - 11	(8)	74151 - 1
$\rightarrow$ D <sup>oc</sup>	7405 - 3	PISO	74165 - 1
$\rightarrow$ D <sup>oc</sup>	7407 - 1	4X4 RAM	74170 - 4
$\rightarrow$ D	7408 - 5	HEX-D	74174 - 3
$\rightarrow$ D	7410 - 5	QUAD-D	74LS175 - 4
$\rightarrow$ D <sub>s</sub>	7413 - 1	UP/DOWN BCD	74LS192 - 812
$\rightarrow$ D <sup>s</sup>	7414 - 8	UP/DOWN BIN	74LS193 - 14
$\rightarrow$ D <sup>oc</sup>	7416 - 2	$\rightarrow$	74LS367 - 5
$\rightarrow$ D HV	7426 - 14		49704 - 7
$\rightarrow$ D	7432 - 7		49705 - 2
$\rightarrow$ D	7428 - 1		555 - 6
$\rightarrow$ D <sup>BUFF</sup>	7437 - 1		
$\rightarrow$ D	7440 - 1		
(10/4)	7442 - 6		
BCD-DEC	7445 - 2		
BCD-7seg	7446 - 1		
BCD-7SEG	7448 - 5		
2X J-K	7473 - 1		
2X D	7474 - 3		
2X J-K	7476 - 2		
FULL ADD	7483 - 1		
4BIT COMP.	7485 - 2		
$\rightarrow$ D	7486 - 4		
4X16 RAM	7489 - 2		
BIN OR	7493 - 6		
	74118 - 1		

# SINGLE OPERAND INSTRUCTIONS

- ✓ CLR(B) } SOP(B).MOD (7)
- ✓ COM(B) }
- ✓ INC(B) }
- ✓ DEC(B) }
- ✓ NEG(B) (17) n
- ✓ TST(B) SOP(B).NONMOD (8)
- ✓ ROR(B) } (12)
- ✓ ROL(B) }
- ✓ ASR(B) }
- ✓ ASL(B) }
- ✓ SWA(B) (11)
- ✓ ADC(B) } SOP(B) MOD (7)
- ✓ SBC(B) }
- ✓ SXT }
- ASH } no SOP la do imajo abilito 0 72 RDD zoto 20
- ASHC } mavedame no isti shoni 2 DOP imbuturbejon

(32)

### DOUBLE OPERAND INSTRUCTIONS

- ✓ MOV(B)       $SM\phi \cdot DM\phi$  (17)j      MOV, DEST ·  $\overline{SM\phi \cdot DM\phi}$  (4)
- ✓ CMP(B)      DOP(B) · NON MOD ·  $SM\phi \cdot DM\phi$  (17)l      DOP(B) ·  $\overline{SM\phi \cdot DM\phi}$  (3)
- ✓ ADD      DOP(B) · (MOV+SUB) · MOD ·  $SM\phi \cdot DM\phi$  (17)k —      DOP(B) ·  $\overline{SM\phi \cdot DM\phi}$  (3)
- ✓ SUB      SUB ·  $DM\phi \cdot SM\phi$  (17)m —      DOP(B) ·  $\overline{SM\phi \cdot DM\phi}$  (3)
- ✓ BIT(B)      DOP(B) · NON MOD ·  $SM\phi \cdot DM\phi$  (17)l      DOP(B) ·  $\overline{SM\phi \cdot DM\phi}$  (3)
- ✓ BIC(B)      } DOP(B) · (MOV+SUB) · MOD ·  $SM\phi \cdot DM\phi$       DOP(B) ·  $\overline{SM\phi \cdot DM\phi}$  (3)
- ✓ BIS(B)      }
- ✓ MUL      } (18)
- ✓ DIV      }
- ✓ ASH      } (23) (SOP) } As' dve modato pod SOP insstrukcije
- ✓ ASHC      } (SOP)
- ✓ XOR      (3)

(st) ... naučeno u no stream - flow - diagrams

# PROGRAM CONTROL INSTRUCTIONS

- ✓ BR (17)<sub>a</sub>
- ✓ BNE
- ✓ BEQ
- ✓ BPL
- ✓ BMI
- ✓ BVC
- ✓ BVS
- ✓ BCC \*
- ✓ BCS \*\*
- ✓ BGE
- ✓ BLT
- ✓ BGT
- ✓ BLE
- ✓ BHI
- ✓ BLOS
- ✓ BHIS \*
- ✓ BLO \*\*
- ✓ JMP (9)
- ✓ JSR (10)
- ✓ RTS (17)<sub>b</sub>
- ✓ MARK (17)<sub>i</sub>
- ✓ SOB (17)<sub>h</sub>

Nugetas (17)<sub>a</sub> li do flow ro Branch in urison  
 completas. deajto alunas ro testouje di lo  
 ston oli ne

\* } ste identici.

\*\*

MISCELLANEOUS INSTRUCTIONS

- ✓ EMT
  - ✓ TRAP
  - ✓ BPT
  - ✓ IOT
- (26) (17)

- ✓ RTI
  - ✓ RTT
- (17)c

CONDITION CODE OPERATORS

- ✓ CLC
  - ✓ CLV
  - ✓ CLZ
  - ✓ CLN
  - ✓ CCC
- (17)f

- ✓ SEC
  - ✓ SEV
  - ✓ SEZ
  - ✓ SEN
  - ✓ SCC
- (17)g

- ✓ HALT
- ✓ WAIT (17)d
- ✓ RESET (17)e

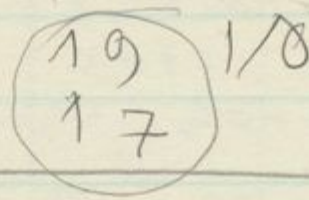
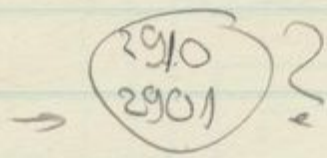
✓ NOP

~~✓ SPI~~

- ✓ MFPI \* (13)
- ✓ MTPI \*\* (14)
- ✓ MFPD \* (13)
- ✓ MTPD \*\* (14)
- ✓ MFPS ✓ (16)
- ✓ MTPS ✓ (15)

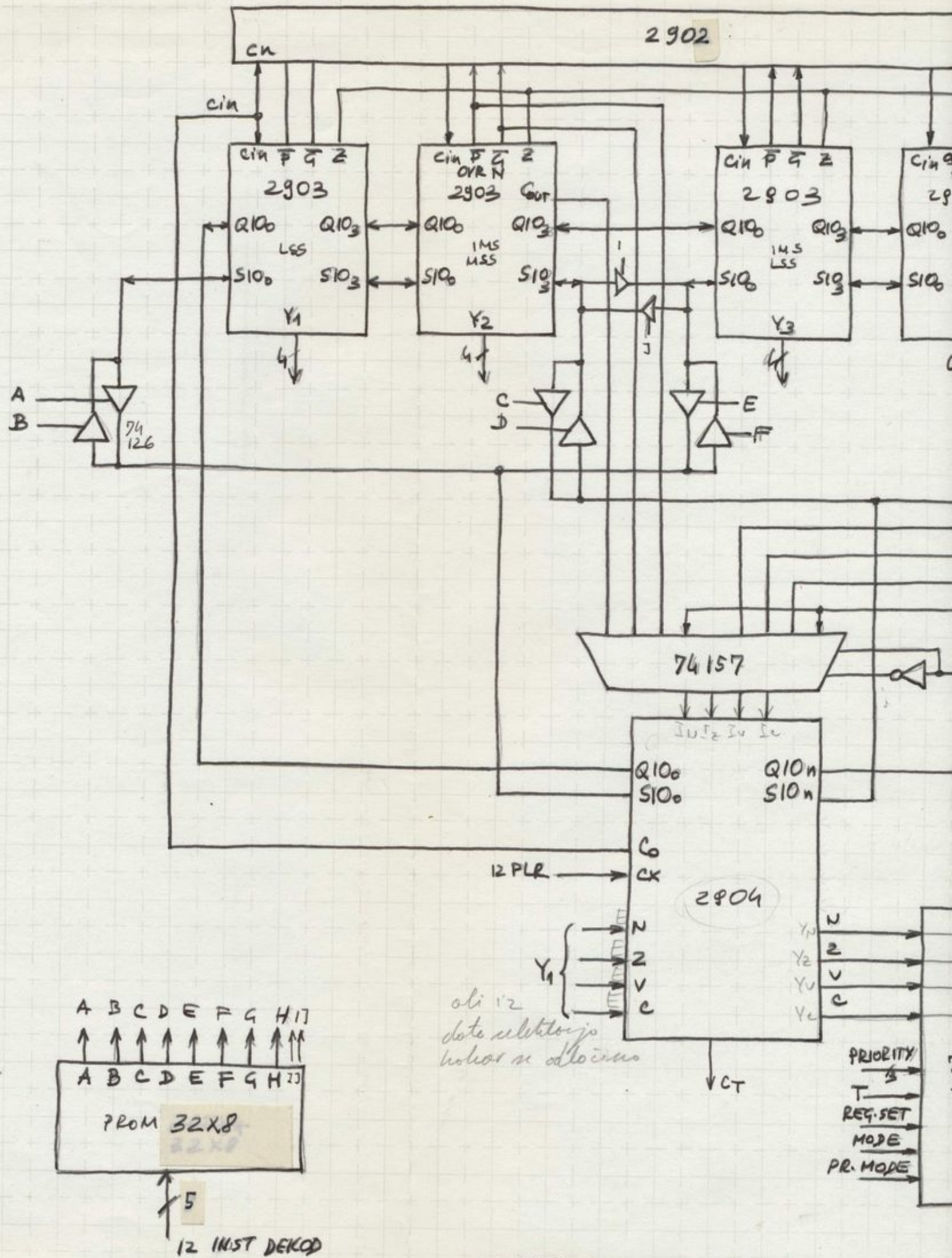
MAPPING ROM OF VTE) E1007

Control unit ↓ optional  
Memory base Δ other optional



- SYS 29105A
- + 32K RAM EXP
- + ~~MS~~ WCS
- + CCU
- + 3X Prototyp. c.
- + Extender

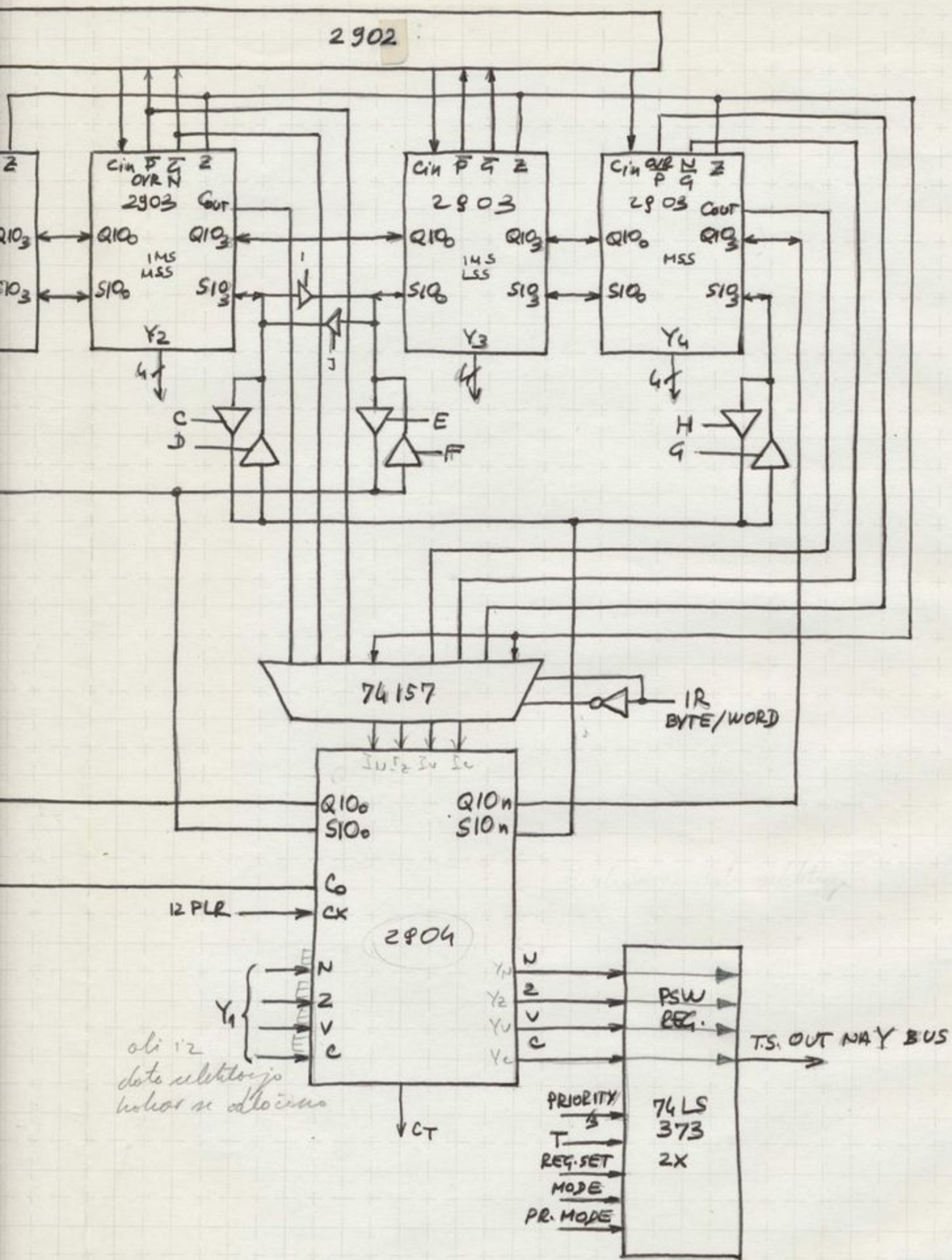
$2K \times 128 \equiv 4K \times 64$



oli iz  
date selektorja  
vohov in odločimo

SHIFT ROTATE CONTROL IN F





SHIFT ROTATE CONTROL IN PSW LOGICA

sestaneb EF. 10.4.1991

Visrout Dobnikov Koder Pipam  
Trehor

- Ali lahko vemo kakre univerze v Nemčiji imajo to opremo.

\* sistem 28/05A

\* software crosscompiler no mini.

} informacijski center

- Traci enote, je dodatek k sistemu

- Lastni development sistem - software

- Prebr. komunik. interfejsa

- Iznos 1600 sistem vključit kot celoto 90M?? Ali je možno investirati.

- Stefan (Zumer je delal nek crosscompiler)

- Martin ??

- Ali lahko kot firma iz sejanke kvote odlayi ob letosnjem rajnu elektronicke?

Ali je možno čisto dojeti vnos do gre v sklop univerze.

- Zoceni vnos ??? Koliko časa je potrebno za razvoj mProg

- Dobro od kontrolo (glede univere v Linzu)

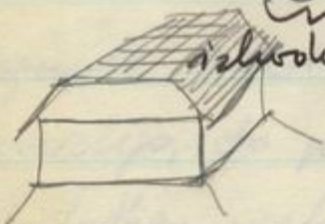
- Unibus timing. za Pajona UNIBUS SPEC. PART 4 KOPICA sem dostavl Dobnikovju

WILLI HESSE

A-2345 BRUNN a GELB. Industriestraße B13

telefon 0 22 36 / 866 31-0 telex 79337

99 43 ob desetih 15.



izkušeno st austrija

V MEMOJI

POSILAJ BO TELEX Z NASLOVI FAKULTET IN

Z IMENI LJUDI NA KATERE JE LAHKO OBRNEMO

IEEE FRANC BRATKOVIĆ FE SOBA 206

17.4.81 FE

Vivout, Dobnikor, Vuelko, Kodler, Pipon  
Trelor, Motybe

- SUB instrukcija ca SMO.DMO pogledati no sistema  
holo a izvaja.
- Ali imamo holisne stipendiate v III letniku

**312 988 int 37**

ZDENEKA VELKAVEN

Gouica Romeo	III	let	} FE
Zivec Vojko	IV		
Fetho Ipa	IV	let	
Hriban Tomaz	IV		
Hren Marko	IV	let	
Volc Vido	III	let	

7.5.81 FE

Vivout Vuelko Kodler

Trelor

Doujon bo pogledal za software v ZDA

# DATA TRANSFER VEZJE K2-1

Ta logika nadzira stanje UNIBUSA to pomeni, da nadzira signale BBSY, MSYN, CL, CO, detektira poritete napake in bus errors. BE.

## KONTROLNO VEZJE

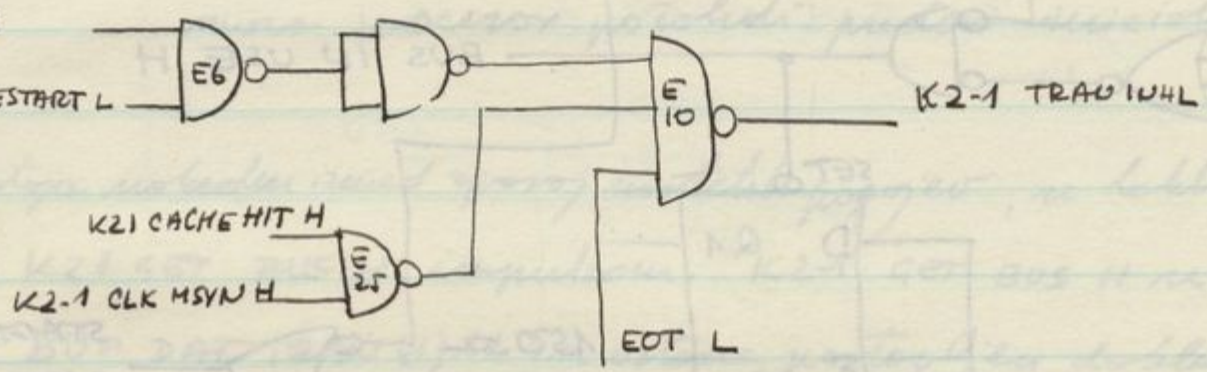
### PROCESOR CLOCK INHIBIT

Uni prenosi podatkov se iniciirajo s signalom K2-8 BUF DATA INH. Ta signal pride iz PLR (E96).

Ko gre K1-5 TAP 30H v stanje H in signal kombinira z K21 ABORT RESTART L (ki je nicer normalno v stanju L). S tem ustvarimo K21 TRAN INH L in ustavimo procesorski clock dokler proces

K1-5 TAP 30 H

K21 ABORT RESTART L



prenosu ni konpletiran. Popolzi hi so potrebni za to da ustavimo proc. clock so se K2-1 CACHE HIT H in K2-1 CLK MSYN H.

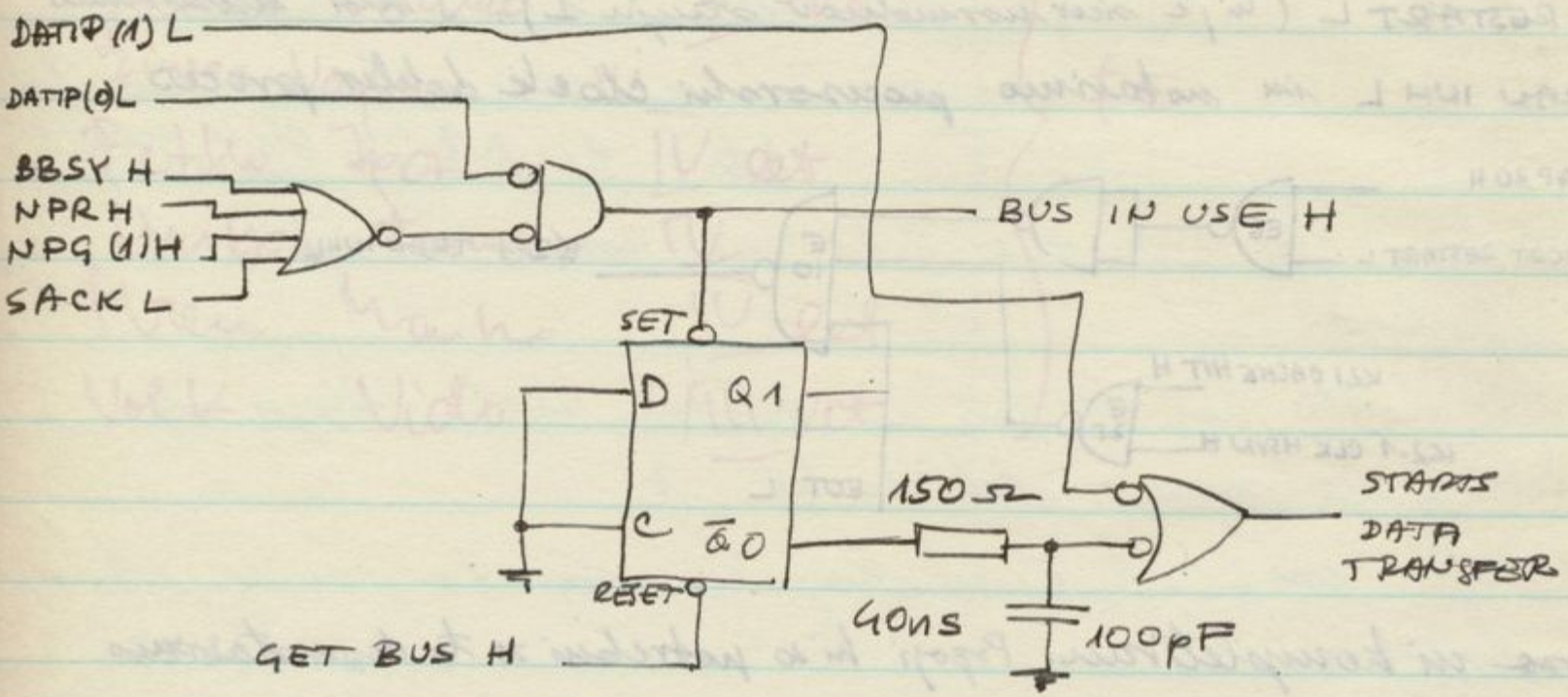
CACHE HIT L se pomeni da imamo cosche pomnilniško operjo in je CACHE upotevil, da imamo podatke, ki se zoliterajo.

Deset bus adres linij se prenese na JA konektor na M826J modulu. Ko je postavljen K2-1 START TRAN L (med DATI cilom) tokrat CACHE uporablja BA linije zato da upotevi ali ima podatke. Ce no podatki v CACHE pomnilniku, se postopaj signali CACHE HIT L in TRI STATE AMUX L. Ti signali povrocijo, da procesor abortira prenos na UNIBUSU ter sprejme podatke preko naslednje podatkovne poti:

Zamirni signal CACHE HIT L bo postaril K2-1 CACHE HIT H.  
 To signal bo zapet po spustil procesorjev clock, ko gre  
 K2-1 CLK MSYN H  $\approx$  H (to je približno 150 ns potem, ko  
 gre K2-1 START TRAL L)

SINHRONIZACIJA NA UNIBUSU

Sinhronizacijska logika arbitrira kdo bo izvajal kontrolo nad  
 BUSOM. Logični 1 (H) (+3V) na vnosu flip flopa E31  
 določa, da je unibus zaseden.



K2-2 NPR H Periferija je postarila NPR in čeli takoj kontrolo nad  
 busom

K2-1 BBSY H Delo določeno periferne enote je imo kontrolo nad Busom  
 in je postarila BBSY signal

K2-2 NPQ(1) H NPR naprava je zahtevala kontrolo nad busom in  
 kd 11-6 procesor je izdelal NPQ. Obstoji veliko tehnično  
 stanje, kjer je NPR naprava je prepoznala  
 NPQ in je spustila NPR, ni pa in postarila  
 SACK ali BBSY.

K2-2 NO SACK

Nopmans je zahtevala kontrolo nad Unibusom KDM-E je izdal GRANT, nopmans je vrnila SACK L. Na temu koliko obstaja pogoj ho obstaja samo SACK L so hoteli čes meduo periferija postovi BBSY.

K2-1 DATIP (0) L

Kodor je to signal v stanju TRUE, tolnot prevlada nad ostalimi signali. To signal indicira, da procesor izvaja DATIP (Read / Modify / Write) operacijo. in ima kontrolo nad Unibusom (postovljen, i BBSY).

BUS SSYN L

NPD nopmans je ncar lahko doble GRANT vendar mora počakati da procesor sprosti BBSY prenos podatkov n se veduo kompletira, zato mora procesor počakati meduo inicilizira drugega.

Či ne obstoja nobeden izmed zgoraj nastetih pogojev, n lahko E31 retira z K21 GET BUS H impulzom. K21 GET BUS H n postavi K2-8 BUF DAT TRAN (1) H in ostane postovljen dokler ne gre K2-1 GET BUS H v LOW ob TAP 30 H. Postavitve E31 začne prenos.

BUS CONTROL

Ko je E31 FF retiran pične DATA TRAN vezje s prenosom podatkov toho, do postavi K2-1 ENAB ADDRS L in o tem sproži naslednje ukrepe.

- 1) Enablira bus address driverje BUS A15: A00 na K1-6
- 2) Enablira BBSY driver K2-1
- 3) Enablira kontrolne signale BUS CO in BUS C1

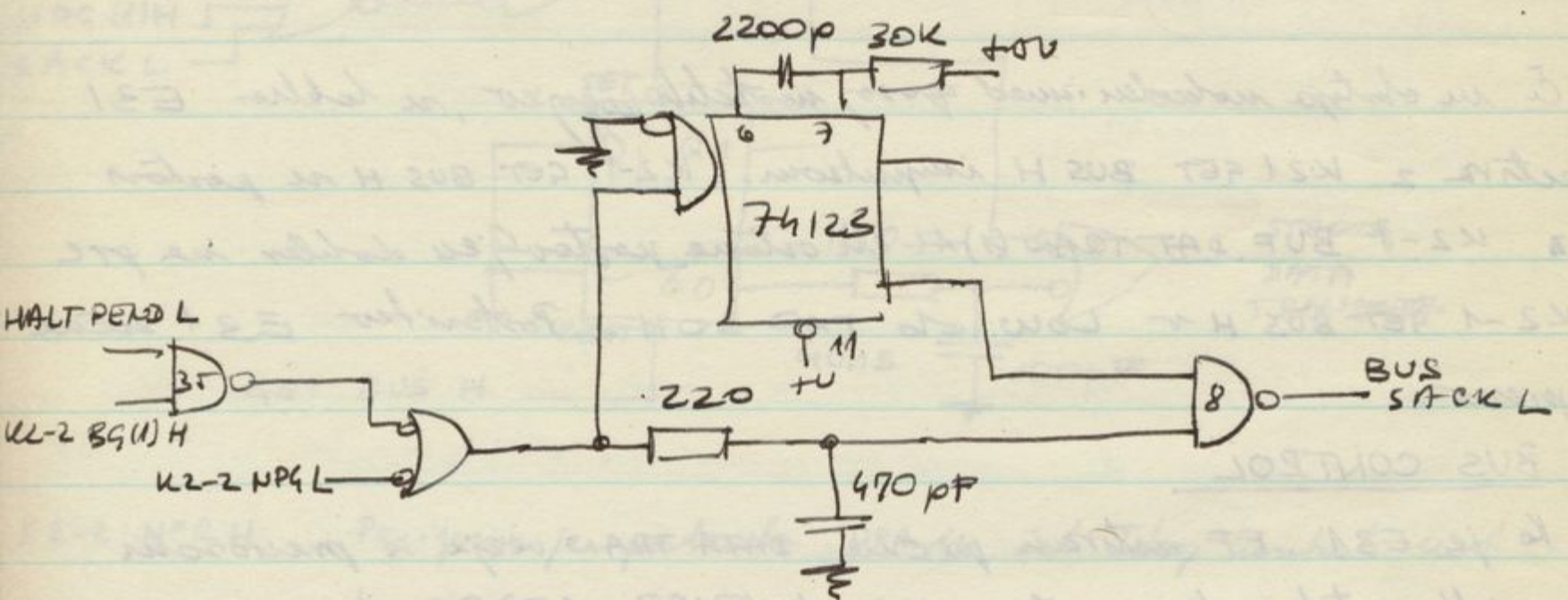
CA	CO	
0	0	DATI
0	1	DATP
1	0	DATO
1	1	DATOB

dejansko stanje teh kontrolnih linij je obločeno z: K2-8 BUF CO(1) H in K2-8 BUF C1(1) H.

- 4) Enablira bus data driverje BUS D00-BUS D15 a n izvaja DATO operacijo.

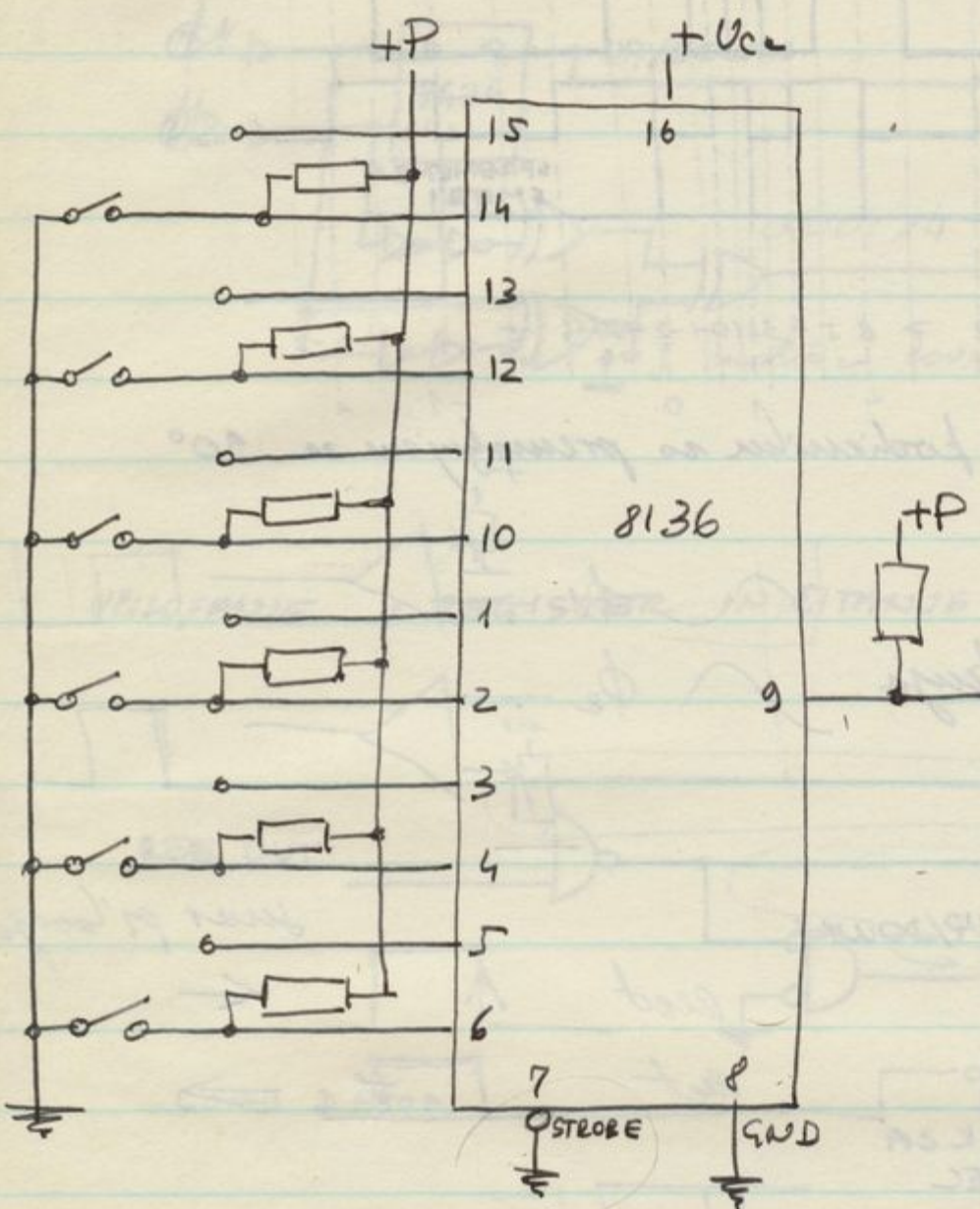
## NO SACK timeout verzija

Verzija na slici postavlja signal BUS SACK L na unikus  $\bar{c}$  neposredno nakon što je zabilježila kontrolu nad unikusom ne postoji SACK u vremenu 22  $\mu$ S nakon što je bila grant linija omogućena. Grant signali K2-2 BGL i K2-2 NPG L se odvijaju u verziji E35. Izlud iz E35 je enable signal na NAND vrata EP i na prozi monoflop E14. Izlud iz E8 je BUS SACK L. Monoflop zadržava <sup>izdajnički BUS SACK L</sup> 22  $\mu$ S, jer produkuje 22  $\mu$ S impulse koji omogućuju enable signal E8. Kad god je postavljen BUS SACK L, hoće to poručiti da hoće procesor spustiti GRANT liniju, hoće potom upliva usvojeno to do verzije spusti BUS SACK L.

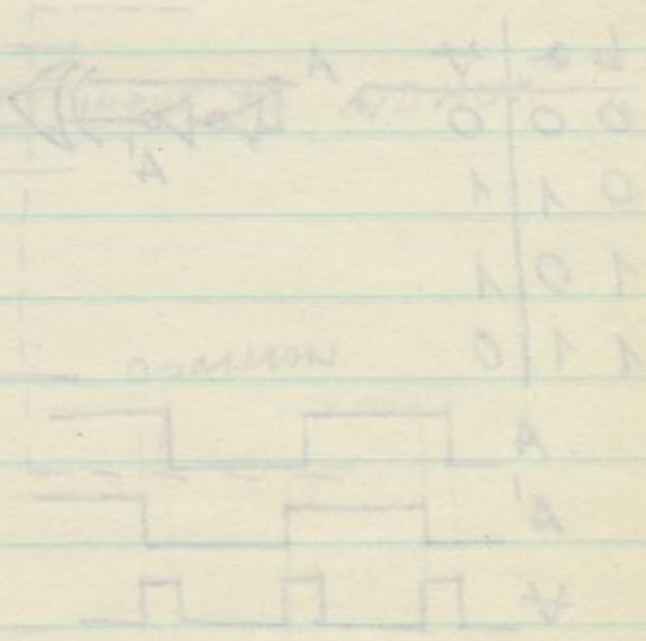


To verzija preporučuje, da bi se procesor obavio  $\bar{c}$  je grant linija postavljena i  $\bar{c}$  nikada dobili BUS SACK od neposredno, hoće zabilježila kontrolu nad busom.  $\bar{c}$  neposredno, hoće zabilježila kontrolu nad busom vrme BUS SACK to verzija se hoće postaviti BUS SACK, hoće hoće grant linija kodla predusitice monoflop. K2-2 HALT PEND L i K2-2 BGL (H) se vrme hoće konjunkcija do preporučeno <sup>do hoće signala</sup> HALT GRANT. <sup>izlud iz E8</sup>

# 8136 ADRESNI DEKODER

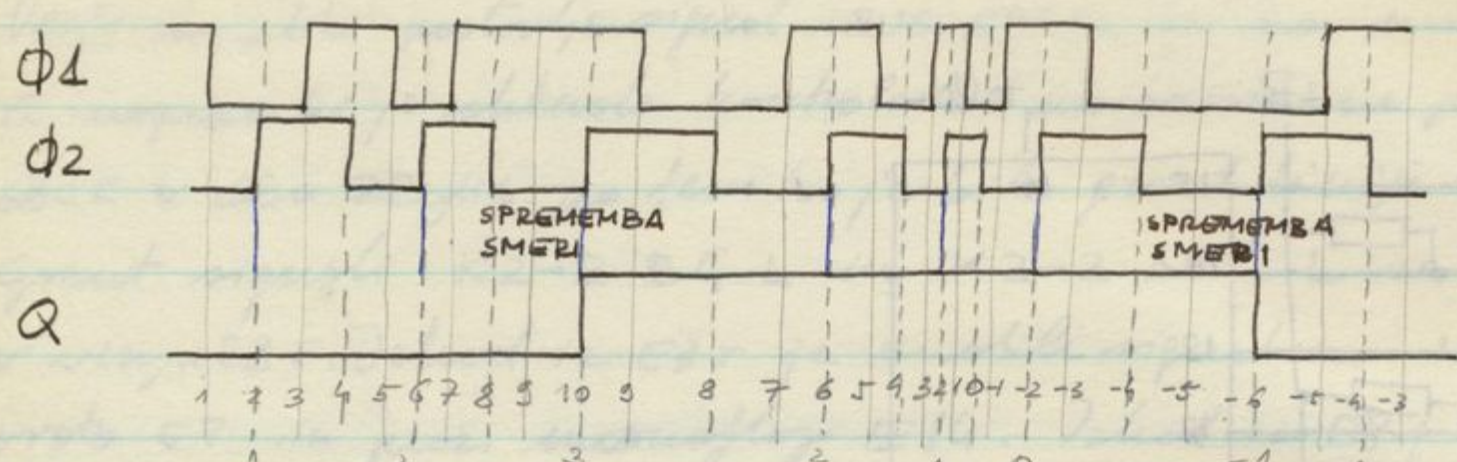


$0,22 \mu$





# DETEKTOR SMERI GIBANJA INKREMENTALNEGA KODIRANJA

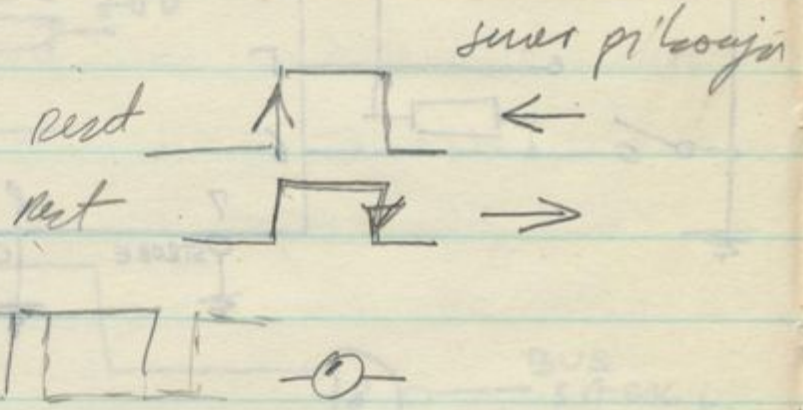
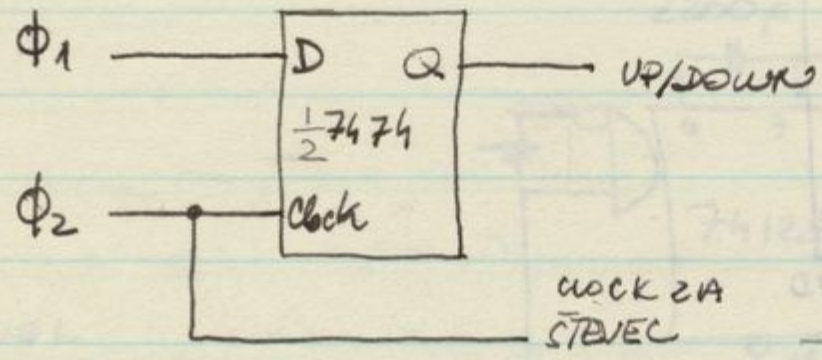
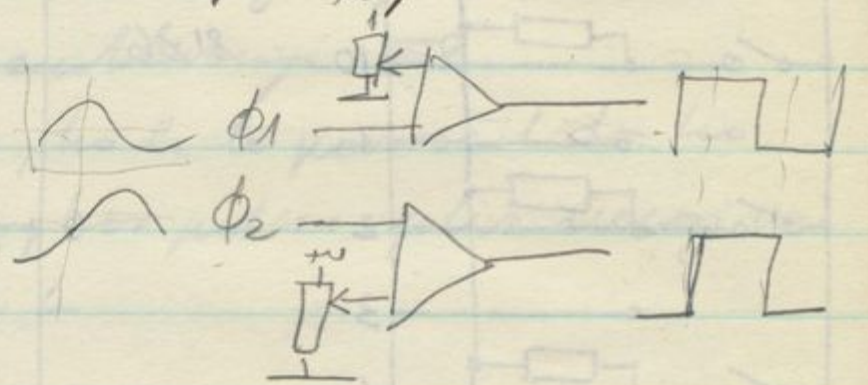


Fotocelice v instrumentalnem kodiranju so premaknjene za 90°

$\phi_1 = D$

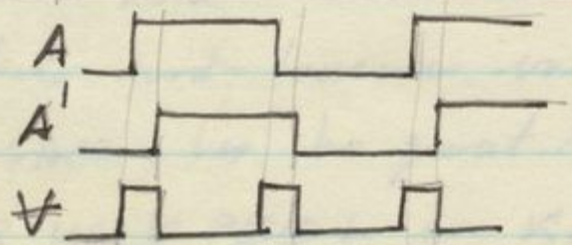
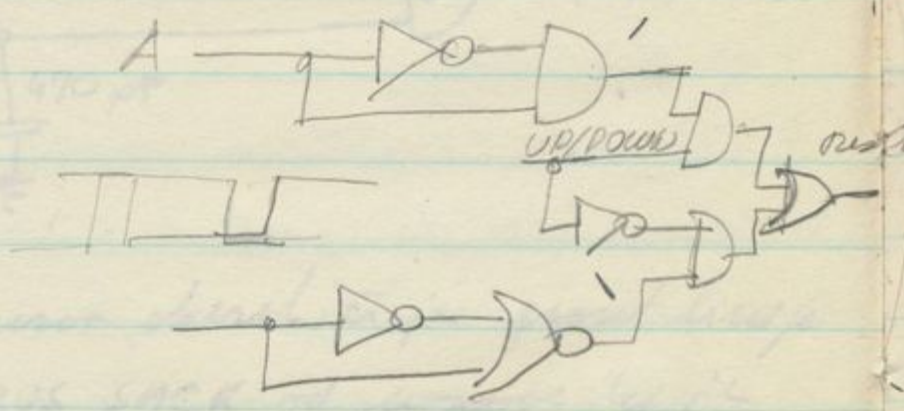
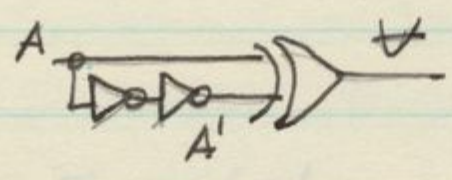
$\phi_2 = \text{clock}$

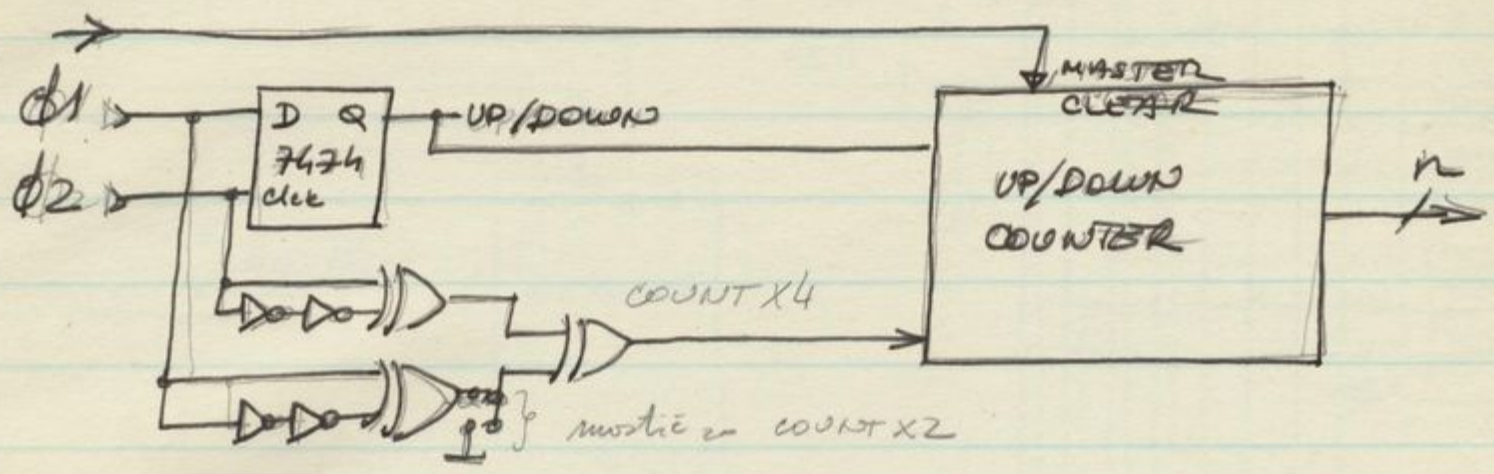
$Q = \text{detekcija smeri vrtenja}$



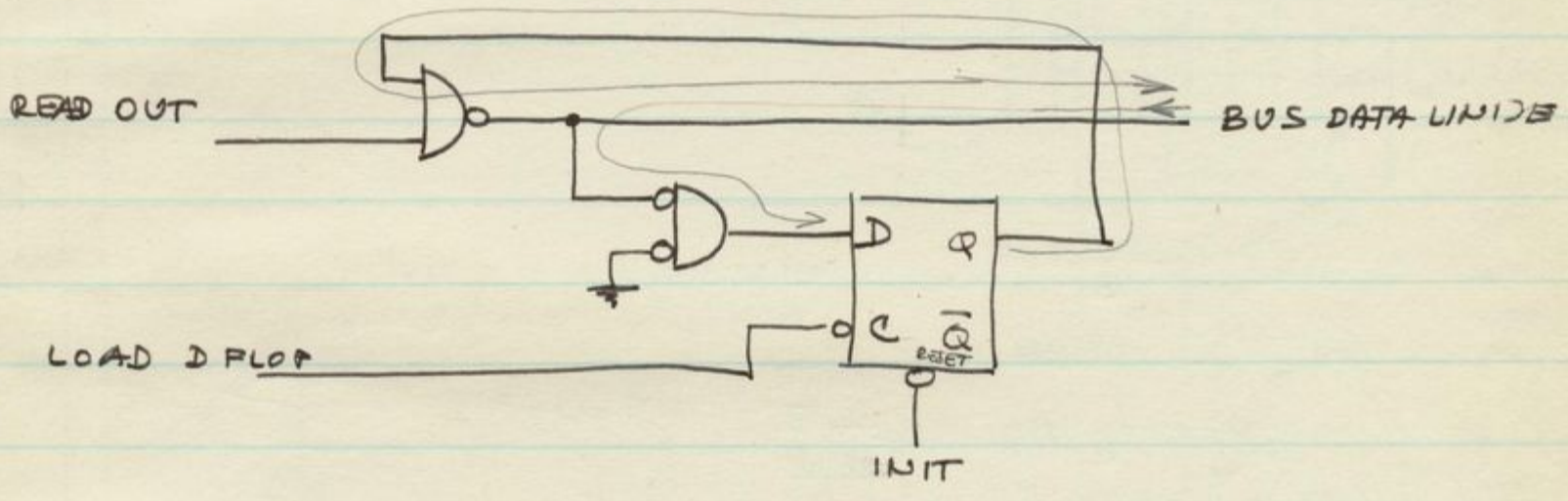
## DETEKTOR FRONTE

b	a	$\nabla$
0	0	0
0	1	1
1	0	1
1	1	0



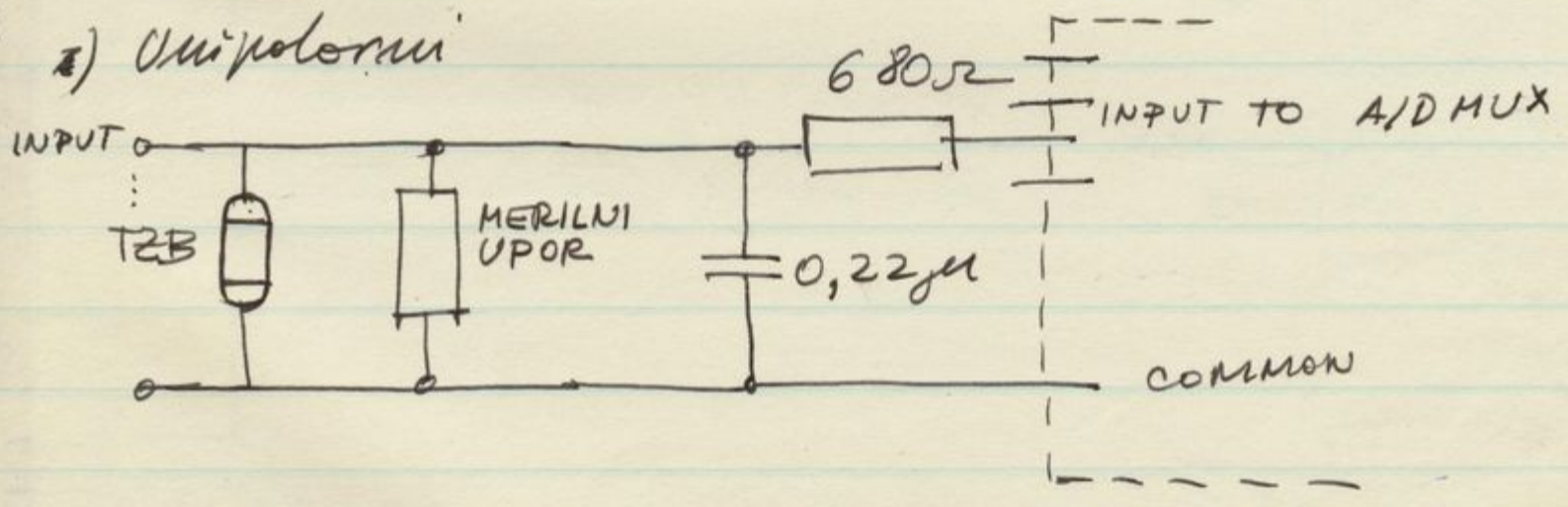


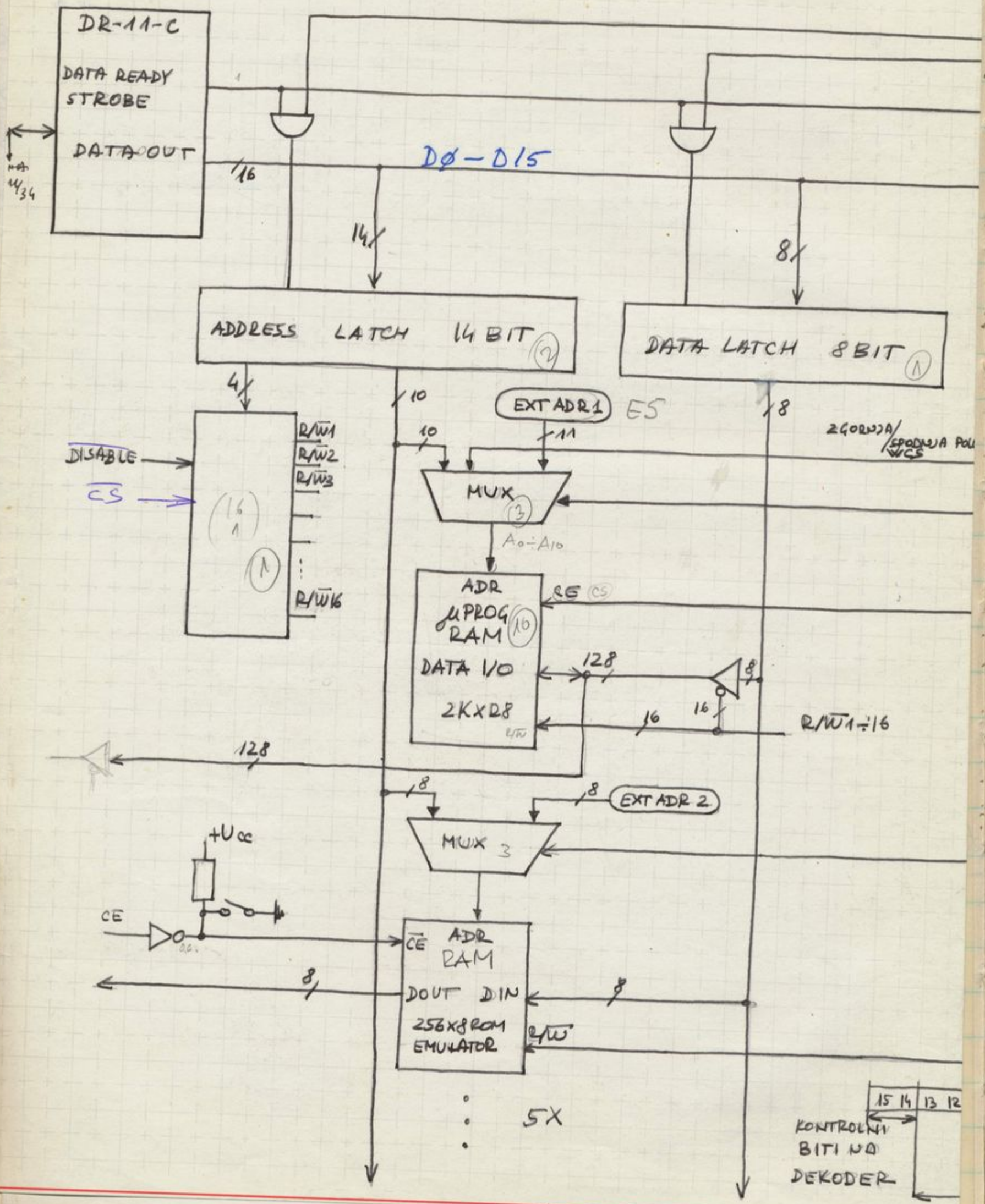
UPISOVANJE V REGISTER IN ČITANJE 12 REGISTRA NA U-BUS

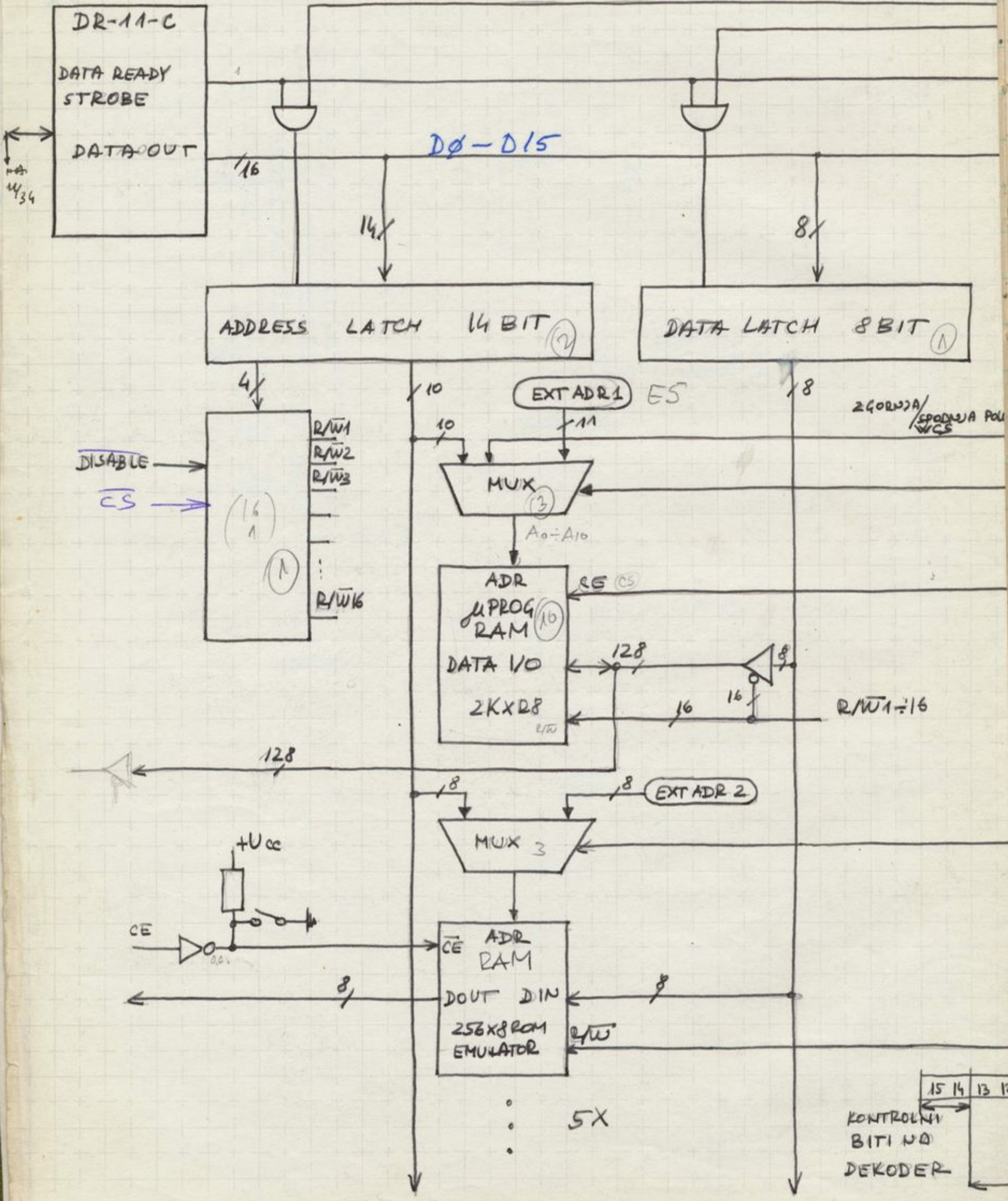


MERILNI VHODI V A/D KONVERTER PREKO MUX-a

a) Unipolarni







D0-D15

ADDRESS LATCH 14 BIT

DATA LATCH 8 BIT

EXT ADR 1 E5

ZGORNJA / SPODNJA POLA  
WCS

MUX

ADR UPROG RAM  
DATA I/O  
2Kx28  
R/W

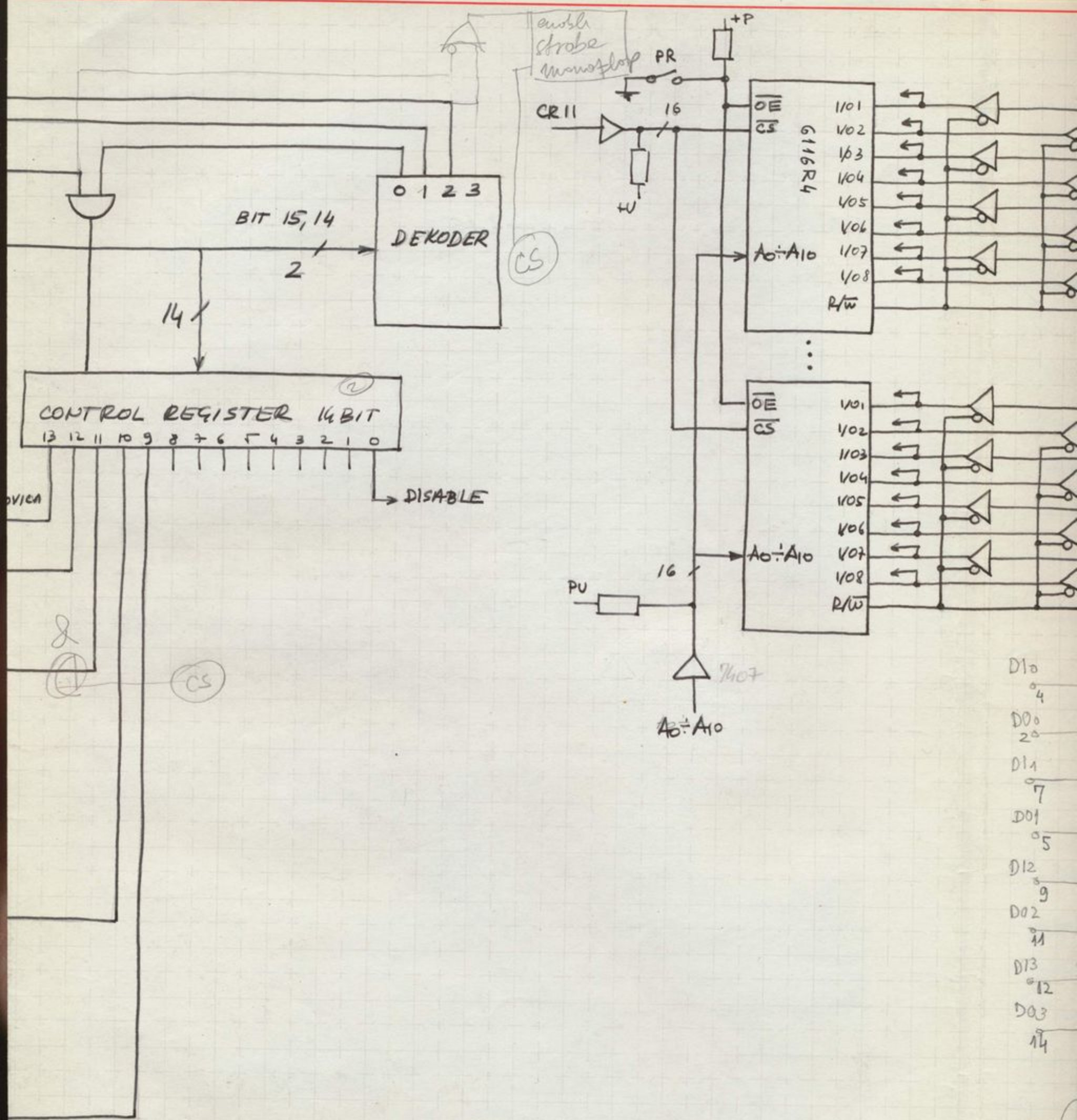
R/W1=16

MUX 3

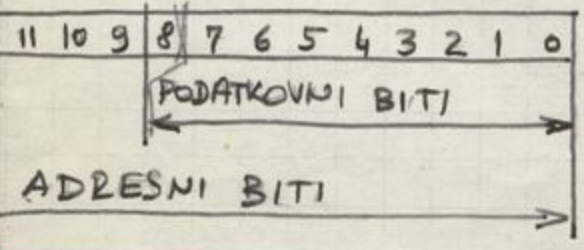
ADR RAM  
DOUT DIN  
256x8 ROM EMULATOR  
R/W

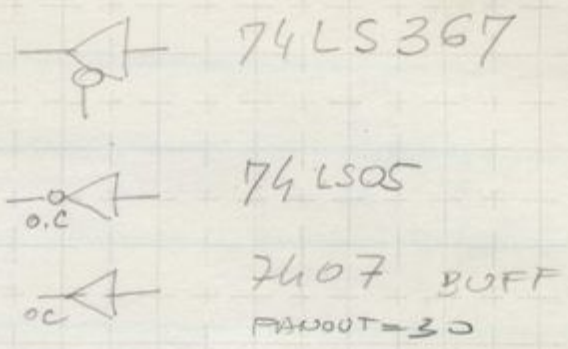
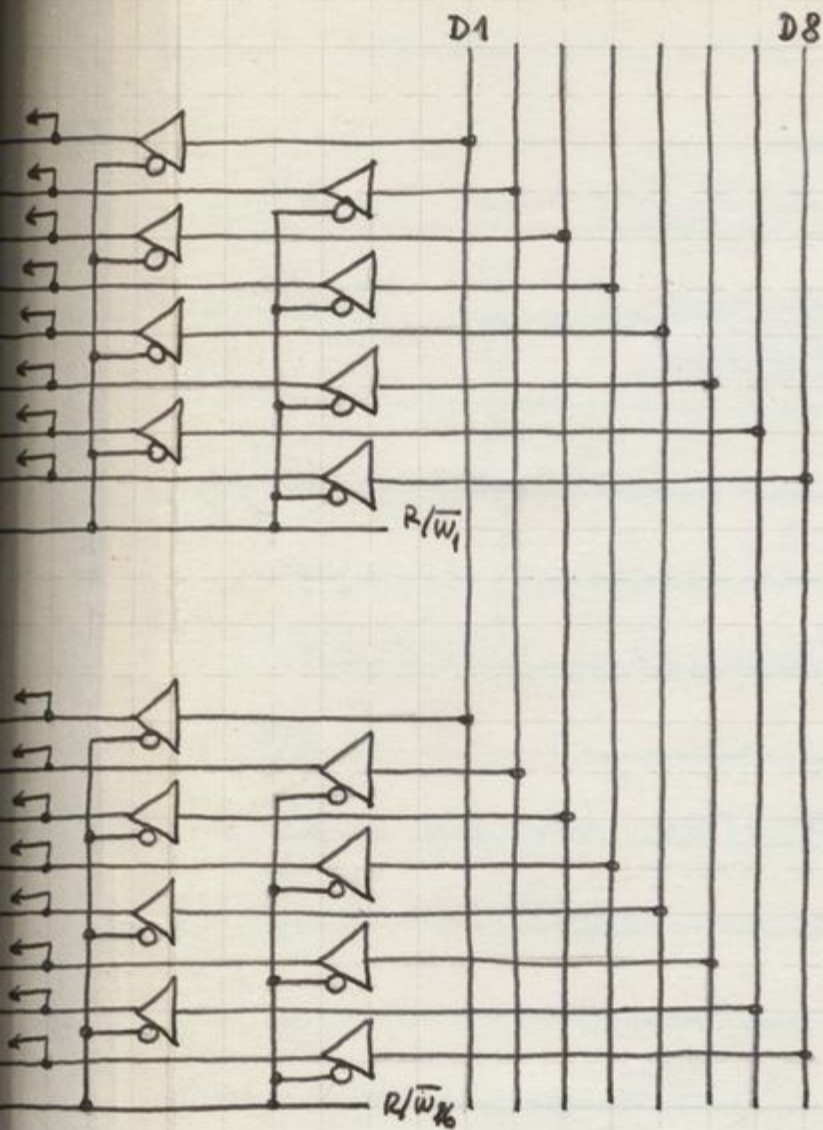
5X

KONTROLNI BITI NO DEKODER  
15 14 13 12

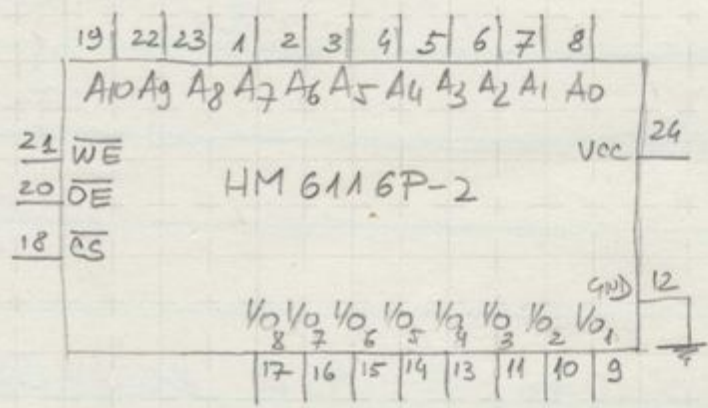
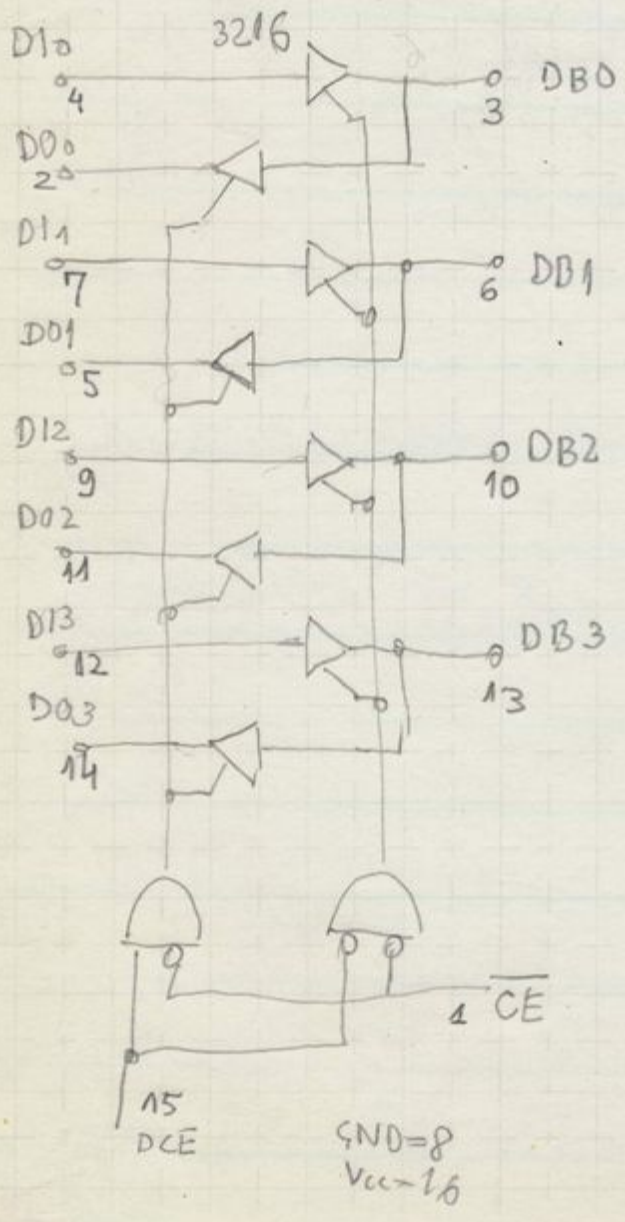
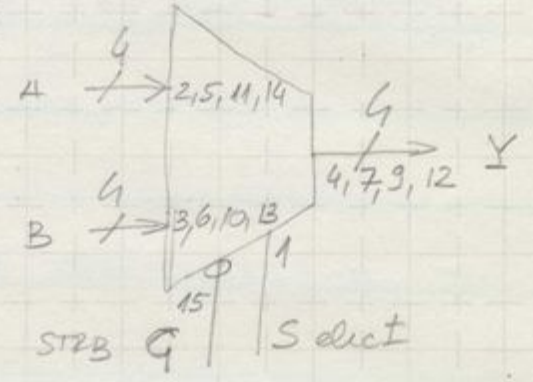


- D10 4
- D00 20
- D11 7
- D01 5
- D12 9
- D02 11
- D13 12
- D03 14





74157



WE mora biti H med uremi adresnimi tranzistorji  
 VPIS se izvri ob  $\overline{WE} = L$  in  $\overline{CE} = L$

# LSI-11 BUS

36 dvostranih signala  
2 enostrane signala  
redeliter

- 18 DATA/ADDRESS LINIJ BDAL <17:00>
- 6 kontrolnih linij za prenos podataka  
BBS7, BDIN, BDOUT, BRPLY, BSYNC, BWTBT
- 3 DMA kontrolne linije BDMG, ~~BDOUT~~ BDMR, BSACK
- 6 INTERRUPT kontrolnih linij BEVNT, BIAK, BIRQ4, BIRQ5, BIRQ6  
BIRQ7
- 5 sistemskih kontrolnih linij BDCOK, BHALT, BINT, BPOK, BREF

INTERRUPT ACKNOWLEDGE BIAK } sta fizično enostrane in sta  
DMA GRANT BDMG } povezani v stilu DAISI CHAIN.

Sprejeto se ta signala na dve input pin-ih (BIAK1 ali BDMG1),  
Preko vrat na modulu se vrne na izhodnih spolkah BIAKO ali BDMGO  
Sprejeto je njih naprave z visjo prioriteto in jih oddajojo  
naprave z nižjo prioriteto.

SIGNAL SPONKA

**BBS7L** AP2 BANK 7 SELECT - Bus master portari BBS7L, kadar I/O  
naprava odnese zgorajih 4K adresnega področja.  
Sedaj je postavljen signal BSYNC L, signal BBS7L  
pa ostane aktivni za trajanje adresnega dela  
bus cikla

**BSYNC L** AJ2 Synchronizacijski signal BSYNCL portari Bus master  
do indicira da je portari adresa na BDAL <0:17>L.  
Transfer teče dokler BSYNCL ni neaktiv.

SIGNAL SPOUKA

- BIAKI L AM2** Interrupt Acknowledge in interrupt acknowledge output.
- BIAKO L AN2** To je interrupt acknowledge signal, ki ga generira procesor v odgovor na interrupt request **BIRQ L**. Procesor postavi **BIAKO L**, ki ga vodi na **BIAKI L** pri prvi napravi na BUSU. Če je zahtevala interrupt, bo inhibirala prehod **BIAKO L**. Če ni postvila **BIRQ L**, bo naprava postavila **BIAKI L** na naslednjo (LOW PRIORITY) napravo preko **BIAKO L** pina preko **BIAKI L** pina na napravi z nižjo prioriteto.
- BIRQ L AL2** Naprava postavi to signal kodor so aktivni interrupt enable in interrupt request flip flopi. Če je PSW bit 7 = 0, procesor odgovori z acknowledge request tako da postavi **BDIN L** in **BIAKO L**.
- BDIN L AH2** Podatkovni vhod **BDIN L** se uporablja za 2 tipa BUS operacij:
  - = Kodor je postavljen med **BSYNC L** časom, tolnost **BDIN L** implicira input prenos z ozivom na trenutni BUS master in zahteva odziv **BRPLY L**. **BDIN L** se postavi kodor je master pripravljen sprejeti podatke iz slave naprave.
  - = Kodor je postavljen brez **BSYNC L**, indicira, da je prišlo do interrupt operacije. Master mora desker podatke, ki ustrezajo iz **BRPLY L**.
- BRPLY L AF2** se postavi, ko odgovor na **BDIN L** ali **BDOUT L** in med IAK prenosom. Generira ga slave naprava, da indicira da je postavila podatke na **BDAL** bus ali pa da je sprejela podatke s tega busa.



**BDOUT** AE2 Izhod podatkov - BDOUT. Kodov je postavljen implicitno, do 20 podatki na  $BDAL < 0:15 > L$ , in do se izvaja prenos 12 bus master uopneve. BDOUT L se desker glede na podatke na busu. Slove uopneve odgovarja na BDOUT L signal s postavitvijo BRPL L do konceptiva prenosa.

**BWTBT L** AK2 Write/Byte - BWTBT L se uporablja na dva načina za kontrolo bus cikla.:

- Postavlja se med predajo fronte signala BSYNC L, da indicira, da ~~se~~ izhodno shemico (DATO di DATOB) in ne vhodno shemico.
- Postavlja se med BDOUT L v DATOB bus ciklu za adresiranje byte.

**BIUIT L** AT2 Inicializacija uporablja se za inicializiranje oz. brisanje vseh uopnev priključenk na 10 bus. Signal se generira kot odziv na power up pogoj ali ka z izvrševanjem RESET instrukcije.

BDAL liniji nosijo odrese in podatke. Kodor je na teh linijah odra, ki je postavljen bus master, je aktiven BSYNC L signal na spouhi A12.

Adresni delodur sestavlja dva čipa E12 in E23. Obe sta 8136 adresna komponenti. Signal BBS7L na AP2 določa ali je bilo odbranjeni zgorajih 4K spouhino, kjer po memory mapu ležijo odrese registrov enote DRV. Ostali del odrese noslovino z programiranimi stikoli W1 do W10. Tam kjer je v odresi 0 delujemo stikolo.

Kadorkoli DRV sprejema ali oddaja podatke - BDINL BDOUTL se z zohornitajo RC konstante 475 1000 pF (50ns) enotno PROM, ki generira kromilne signole.

## DATI

### BUS MASTER

#### ADRESIRA NAPRAVO

- POSTAVI ADRESE NA BDAL (15:00)
- POSTAVI BBS7 ČE JE ADRESA 12 UPPER 4K
- POSTAVI BSYNC L

#### ZAHTEVA PODATKE

- ODSTRANI ADRESE 12 BDAL IN NEGIRA BBS7 L
- POSTAVI BDINL

#### ZAKLJUČI KLOS PODATKOV

- SPREJME PODATKE IN ODGOVORI Z NEGACIJO BDINL

#### ZAKLJUČI BUS CIKLA

- NEGIRA BSYNC L

### SLAVE

#### DEKODIRA ADRESE

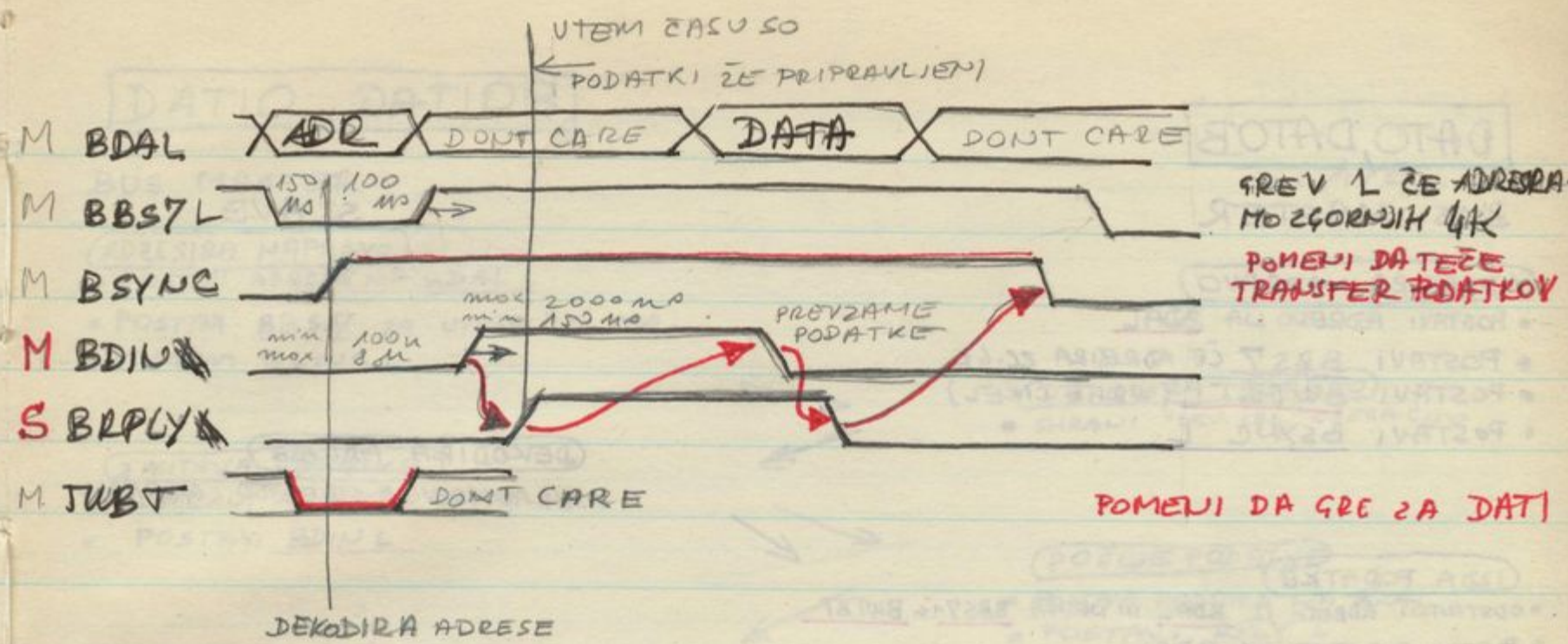
- STORE "DEVICE SELECTED" OPERACIJA

#### VNESE PODATKE

- POSTAVI PODATKE NA BDAL
- POSTAVI BRPLY L

#### KOMPLETIRANJE OPERACIJE

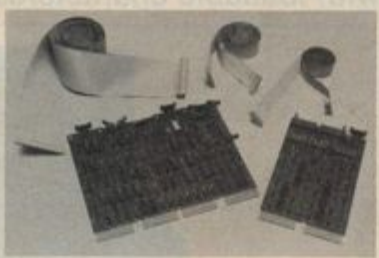
- NEGIRA BRPLY L



NOVEMBER  
R 1982 COMPUTER PRODUCT NEWS 19

**BUS INTERPRETER**  
for Unibus and Q-bus

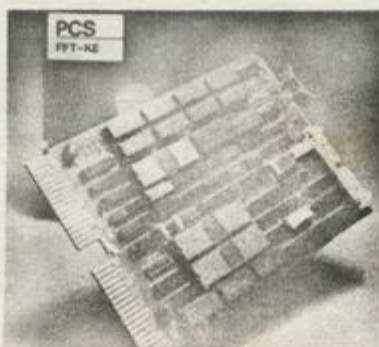
The interpreter assembly with built-in bus repeater, Model MDB/MLSI-DWQ11, permits a PDP-11 based system to use more compact Q-Bus memories and controllers; or an LSI-11 system to use Unibus memories and controllers. It presents one bus load to the main system and the add-on bus can support up to 19 DC bus loads. Key feature is that devices on the add-on bus can be assigned higher or lower DMA and serial interrupt priority than other devices on the main system. The interpreter can be inserted into any slot of either a Unibus or a Q-Bus main system with other interfaces or memories located ahead of, or after it. Priority is determined by where the interpreter is inserted in either the Unibus or Q-Bus slots. **MDB Systems Inc, 1995 North Batavia St, Orange, CA 92665, USA.**



CPN 1961 Enter this number on your Reader Service Card for free detailed information

62 MARCH 1983 COMPUTER PRODUCT I

**FFT BOARD**  
is Q-bus compatible



Based on the AMD 2903, the FFT-KE Bit-Slice-Processor is built on a dual-slot board which connects directly to Q-bus based systems using LSI 11/2 or LSI 11/23 computers. Data transfer is organized independently from the KE board via DMA. All data use DEC compatible 32-bit floating point format. Routines offered include FFT and inverse FFT, vector addition (VADD) and multiplication (VMUL), organization of a data-field transfer between different memories (MOV) and power spectrum calculation (POW). The system is supplied with a PROM set, floppy disk and documentation. **PCS GmbH, 36 Pfälzer-Wald-Str, 8000 München 90, Germany.**

CPN 6202 Enter this number on your Reader Service Card for free detailed information

1982



**Corporate Headquarters**  
1995 N. Batavia Street, Box 5508  
Orange, CA 92667-0508  
Tel. 714-998-6900 TWX: 910-593-1339  
FAX: 714-637-4060

**MDB Systems U.K., Ltd.**  
Basingstoke, Tel. 44 0256 464767  
TELEX: 858389 MDBSYS G

In the western states MDB products are also distributed by **WYLE LABORATORIES** ELECTRONICS MARKETING GROUP

# DATA, DATOB

BUS MASTER

SLAVE

## ADRESIRA NAPRAVO

- POSTAVI ADRESO NA BDAL
- POSTAVI BBS7 ĆE ADRESIRA 2G.4K.
- POSTAVI BWTBTL (WRITE CIKEL)
- POSTAVI BSYNC L

## DEKODIRA ADRESIB

- SHRANI "DEVICE SELECTED" OPERACIJO

## IZDA PODATKE

- ODSTRANI ADRESO IZ BDAL IN NEGIRA BBS7 in BWTBT
- POSTAVI PODATKE NA BDAL
- POSTAVI BDOUT L

## SPREJME PODATKE

- SPREJME PODATKE IZ BDAL
- POSTAVI RPLY L

## KONĀA PRENOS

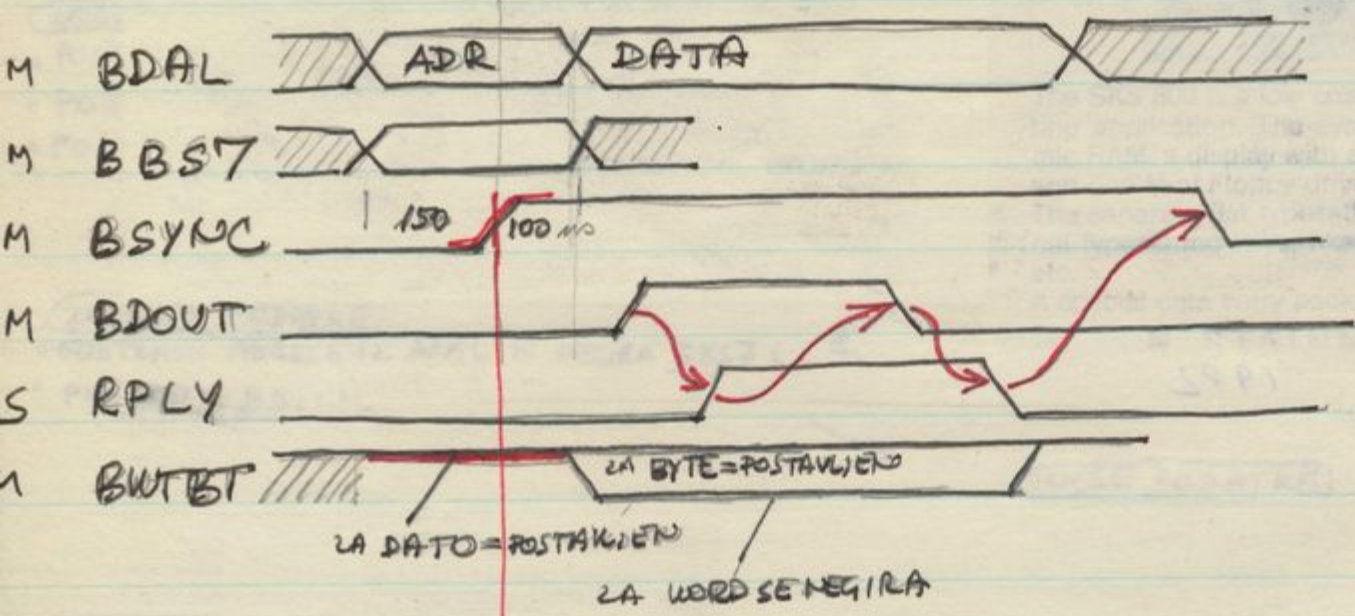
- NEGIRA BDOUT L (IN BWTBTL) ĆE GRE ZA DATOB
- ODSTRANI PODATKE IZ BDAL

## KOMPLETIRA OPERACIJO

- NEGIRA RPLY L

## ZAKLJUĀI BUS CIKEL

- NEGIRA BSYNC L



NAPRAVA PREVLAME  
ADRESO + TOJA JE  
PRENOS DATO

# DATIO, DATIOB

## BUS MASTER

## SLAVE

### ADRESIRA NAPRAVO

- POSTAVI ADRESE NA BDAL L
- POSTAVI BBS7 ZA UPPER 4K ADL.
- POSTAVI BSYNC L

### ZAHTEVA PODATKE

- ODSTRAVI ADRESE Z BDAL
- POSTAVI BDIN L

### ZAKLJUČI VHOJNI TRANSFER

- SPREJME PODATKE IN ODGOVORI Z ZAKLJUČKOM BDIN L

### IZDA PODATKE

- DA PODATKE NA BDAL
- CE JE BYTE PREJOS DA BWTBT L
- POSTAVI BDOUT

### ZAKLJUČI IZHODNI PREHOS

- ODSTRAVI PODATKE Z BDAL L IN L
- NEGIRA BDOUT L

### ZAKLJUČI BUS CIKEL

- NEGIRA BSYNC L IN CE JE BIL BYTE TUDI BWTBT

### DEKODIRA ADRESO

- SHRANI "DEV. SEL." OPERACIJO

### POSLE PODATKE

- POSTAVI PODATKE NA BDAL
- POSTAVI RPLY L

### KOMPLETIRA PREJOS

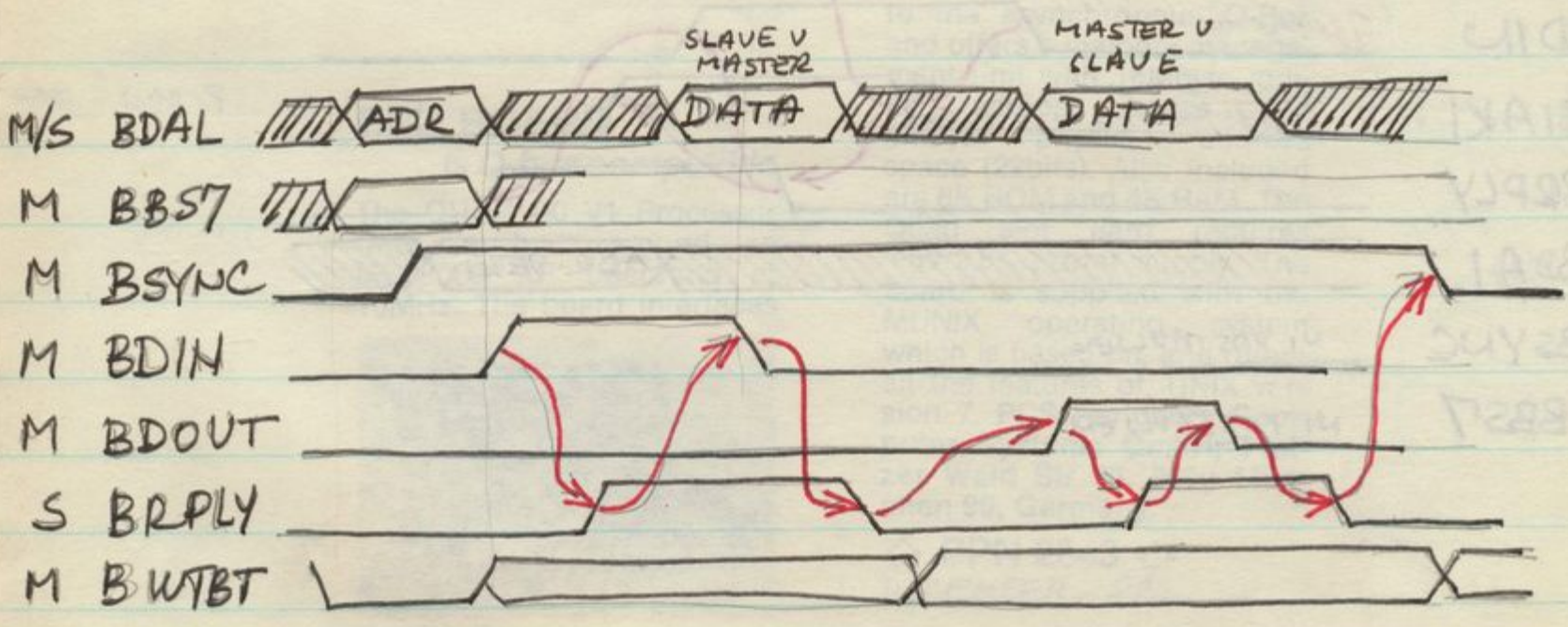
- ODSTRAVI PODATKE
- NEGIRA RPLY L

### PREVZAME PODATKE

- SPREJME PODATKE Z BUS A
- POSTAVI RPLY L

### ZAKLJUČI OPERACIJO

- NEGIRA RPLY L



DIO, DIO

**INTERRUPT**

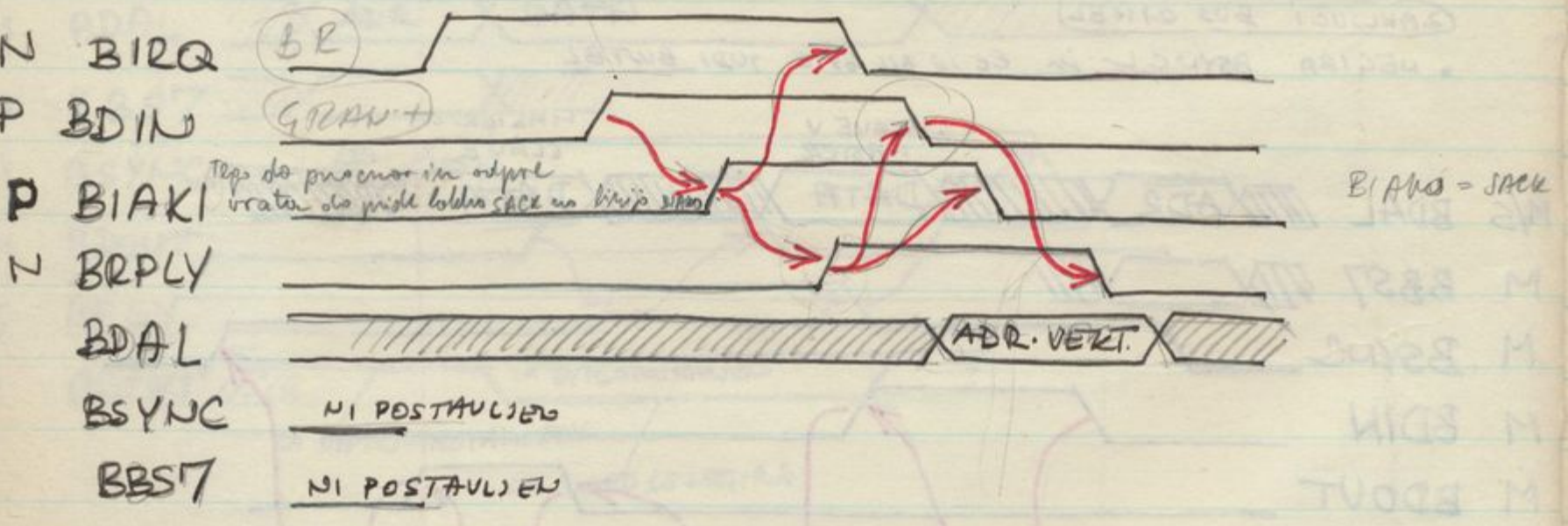
PROCESOR

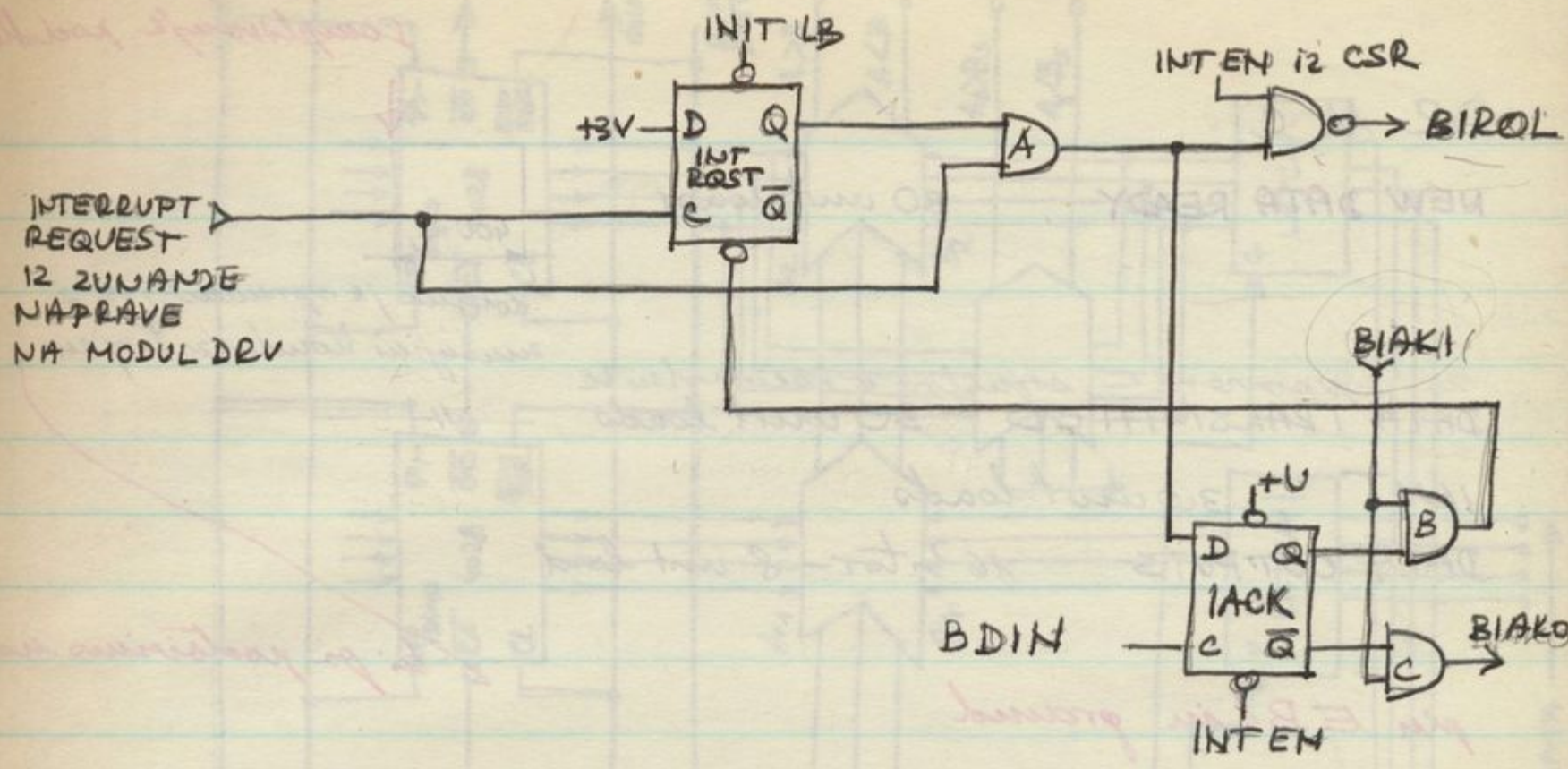
NAPRAVA

- STROBIRA INTERRUPT**
  - POSTAVI BDIN L
- GRANT REQUEST**
  - PAUZA I<sup>u</sup> POSTAVI BIAKO L
- SPREME VEKTOR, ZAKLJUČI REQUEST**
  - SPREME ADRESO VEKTORA
  - NEGIRA BDIN L I<sup>n</sup> BIAKO L
- IZVEDE INTERRUPT**
  - SHRAVI PC I<sup>n</sup> PS NA STACK
  - PREUZAME NOV PC I<sup>n</sup> PS IZ ADRESE VEKTORA
  - IZVRSI INTERRUPT RUTINU

- INICIALIZIRA REQUEST**
  - POSTAVI BIRQ L
- SPREME BDIN L**
- SPREME BIAKI L**
  - SPREME BIAKI L I<sup>n</sup> I<sup>n</sup>HIBIRA BIAKO L
  - DA VEKTOR NA BDAL (0-15) L
  - POSTAVI BRPLY
  - NEGIRA BIRQ L
- KOMPLETIRA PROMOS VEKTORA**
  - ODSTRANI VEKTOR IZ BDAL BOSA
  - NEGIRA BRPLY L

P = PROCESOR  
N = NAPRAVA

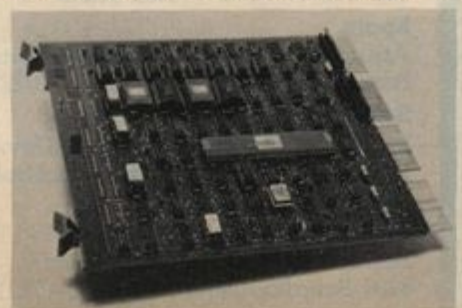




Zunanje vezje do interrupt request in procesorci vpiše "1" v INT RQST FF. Če je v CSR registru dovoljen interrupt, do naprave DRU signal BIRQL. Procesor odgovori na BIRQL s signalom BDIN L. To vpiše "1" iz INT RQST FF v IACK FF in s tem briše INT RQST FF preko BQ vrat ob BIACKI signalu, kar pa do procesor, prehod signala BIACKI skozi modul ho u blokiru na C vratih.

**CPU BOARD**  
is Q-bus compatible

The QU 68000 V1 Processor Card is built around an MC68000  $\mu$ P operating at 10MHz. The board interfaces



to the asynchronous Q-Bus and offers a memory management unit with 16Mbyte logical address space and 4Mbyte physical address space (22bits). Also included are 8K ROM and 4K RAM. The Quad slot card requires +5V/2.5A power supply. The board is supplied with the MUNIX operating system which is based on, and offers all the features of, UNIX version 7. PCS Periphore Computer Systems GmbH, Pfälzer Wald Str 36, 8000 München 90, Germany.

⬆ CPN 2843 ⬆  
DECEMBER 82

DR 11 C

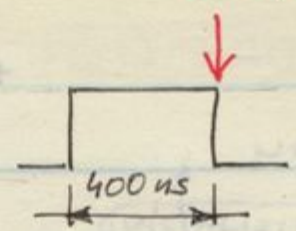
NEW DATA READY - 30 unit loads

DATA TRANSMITTED - 30 unit loads

INIT - 30 unit loads

DATA OUTPUTS 16 bit - 8 unit load

*Simplifikuje podoflor*

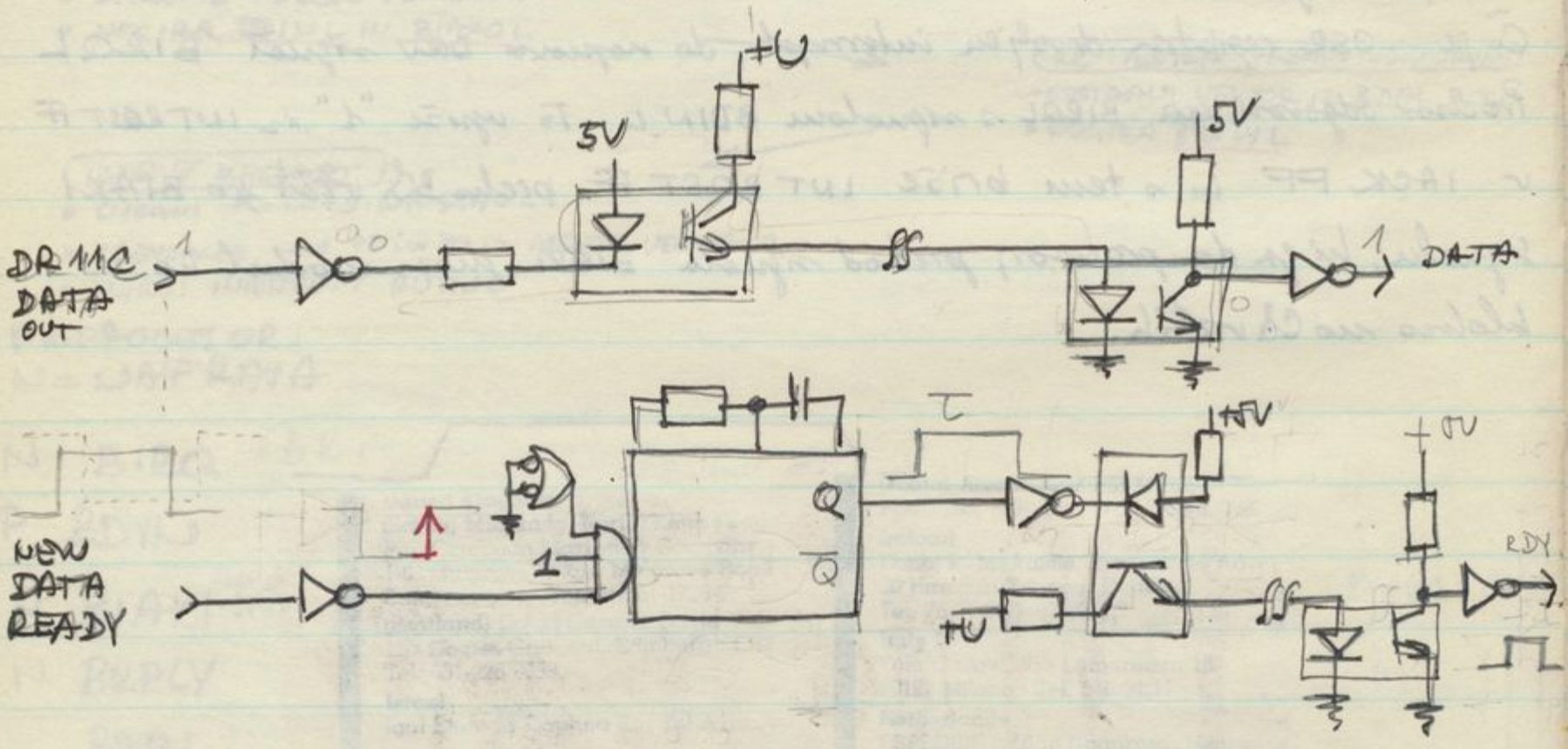


*dužina je spremljiva z zmanjšanim kondenzatorjem*

*pin EB1 in ground*

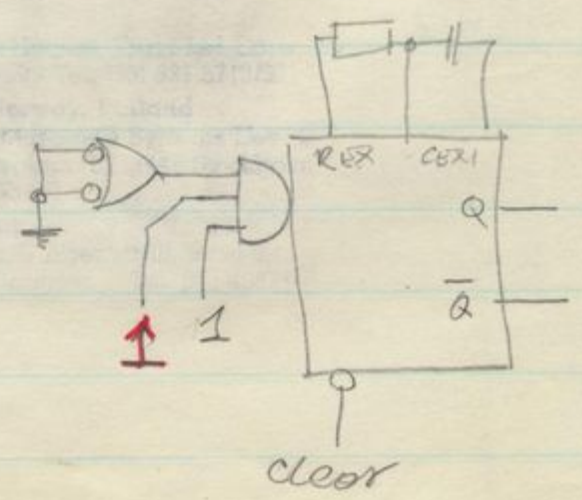
*ki po postavitvi nos*

Prenos iz DR 11 C na kabel - vozilato z polvodno ločljivo

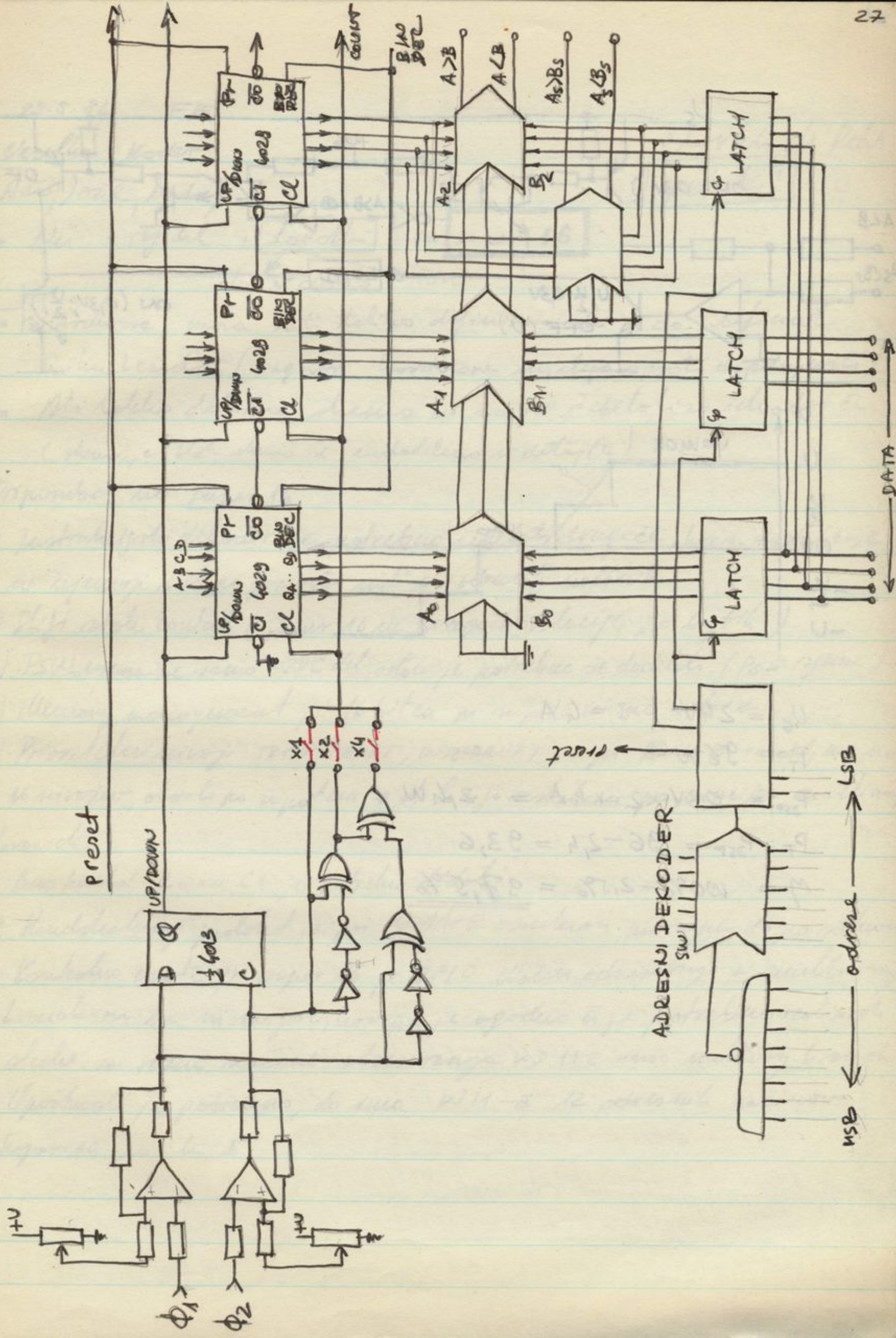


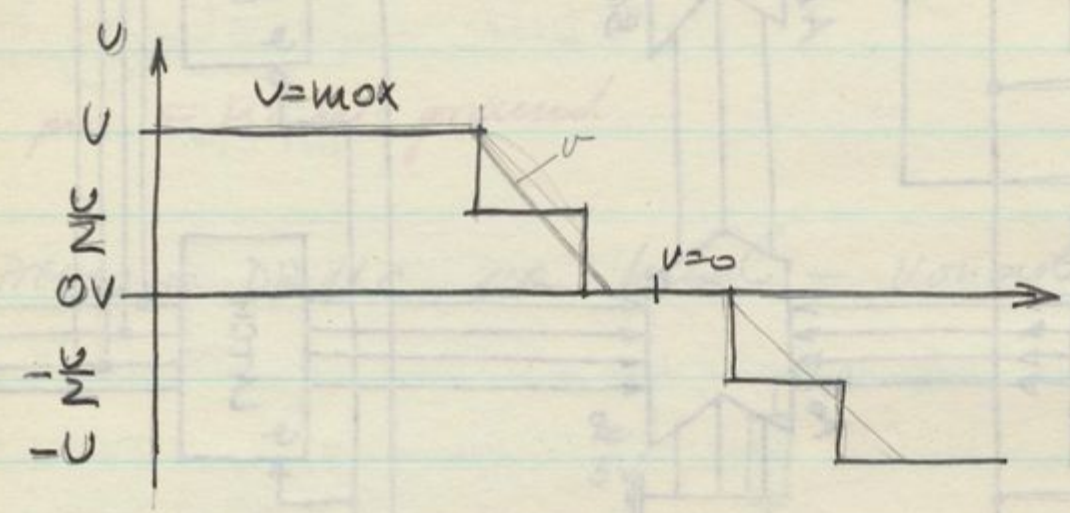
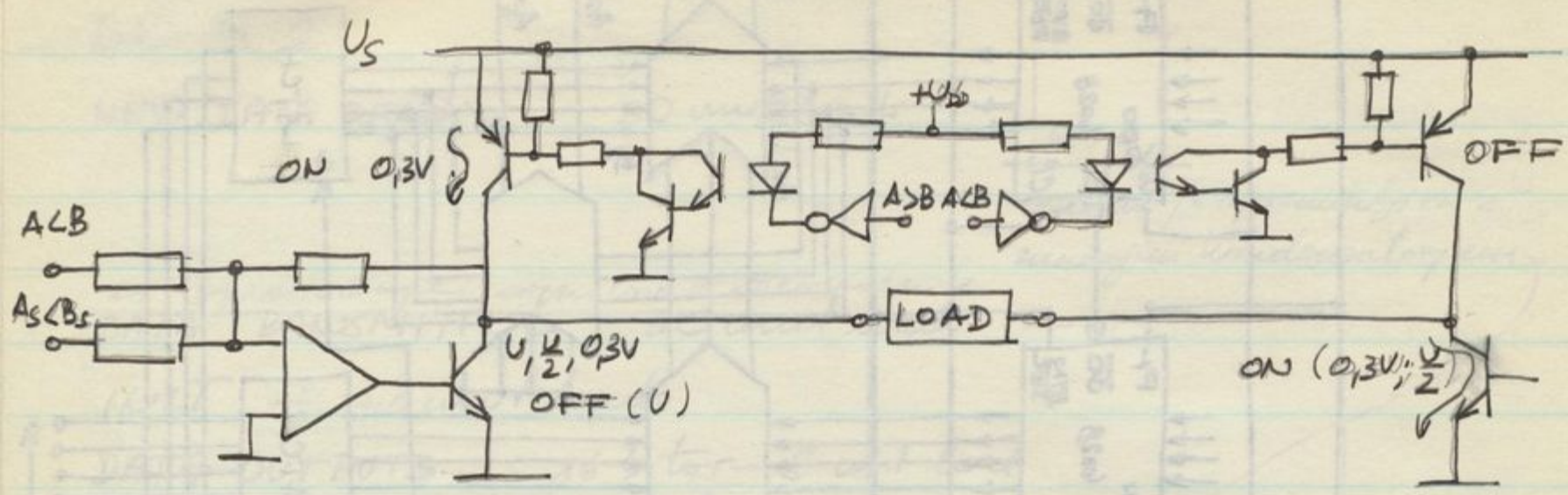
$$t_w = 0.32 R_T C_{EXT} \left(1 + \frac{0.7}{R_T}\right)$$

74122 |  $R_{EXT} = 5K \div 25K$   
 $C_{EXT} = \text{manjšan}$









$U_s = 24V ; I = 4A$   
 $P_T = 96W$   
 $P_{DST} = 0,3V \times 2 \times 4A = 2,4W$   
 $P_T - P_{DST} = 96 - 2,4 = 93,6$   
 $\eta = 100\% - 2,5\% = \underline{\underline{97,5\%}}$

29.5.84 FE

Verelko, Koder

Aco, Jozil, Mudej

Arhiv dr. Pehli  
pet 12<sup>h</sup>

- Ali + rep del + **dodolki** iz **super 16**  
byta swob → PSW

- Hardware mora biti toliko definiran da u do definiciji mikroberuda (odgovor: hardware mislijo sprati doprinyati)
- Ali toliko dolzino shema bi no jo izdelati z izdelatice.  
(shema je blok shema in nedodelana v detajlu)

### Priponbe no super 16

- 1) Instrukcijski dekoder je potrebuso izdelati drugace, ker n instrukcije ne ujemajo niti po formatu niti po adresnih nacinih.
- 2) Shift rotate control (super 16 ne omogoča rotacije po bytih)
- 3) PSW njuno se samo NZVC del, ostalo je potrebuso se dodolati. (Poi izjeme)
- 4) Memory management je 16 biten ni sup. 16 in ne ustrez.
- 5) Prioritetni nivoji TRAP, ABORT, INTERRUPT so pri KDM-E razdeljeni na 16 nivojev, obrti po u potem z at lo cijo s skokom po principu multiway branch.
- 6) Bus kontrol Co in C1 je potrebuso dodati.
- 7) Handshaking protokol je pri KDM-E osimuliran, pri super 16 po simuliran.
- 8) Kontrolna enota pri super 16 je 2810 Nacin adresiranja z multiway branch-on An ni mogoče, ker po je agodno č je potrebuso realizirati shema za razne nacine adresiranja KDM-E ino multiway branch. Uporabiti je potrebuso, da ima KDM-E 12 adresnih nacinov super 16 po le 5.

SPISEK POVEZAV ZA CONTROL STORE & ROM EMULATOR (BUS - POVEZAVE)

A0 — E78/2 — R11/2 — E96/8 — E87/8 — E81/8 — E102/8 — E111/8 — E114/8  
 E10/8 — E15/8 — E22/8 — E27/8 — E36/8 — E42/8 — E51/8 — E57/8  
 E66/8 — E72/8 — E59/2 — E63/2 — E44/2 — E48/2

A1 — E78/4 — R10/2 — E96/7 — E87/7 — E81/7 — E102/7 — E111/7 — E114/7  
 E10/7 — E15/7 — E22/7 — E27/7 — E36/7 — E42/7 — E51/7 — E57/7  
 E66/7 — E72/7 — E59/5 — E63/5 — E44/5 — E48/5

A2 — E78/6 — R9/2 — E96/6 — E87/6 — E81/6 — E102/6 — E111/6 — E114/6  
 E10/6 — E15/6 — E22/6 — E27/6 — E36/6 — E42/6 — E51/6 — E57/6  
 E66/6 — E72/6 — E59/11 — E63/11 — E44/11 — E48/11

A3 — E78/8 — R8/2 — E96/5 — E87/5 — E81/5 — E102/5 — E111/5 — E114/5  
 E10/5 — E15/5 — E22/5 — E27/5 — E36/5 — E42/5 — E51/5 — E57/5  
 E66/5 — E72/5 — E59/14 — E63/14 — E44/14 — E48/14

A4 — E78/10 — R7/2 — E96/4 — E87/4 — E81/4 — E102/4 — E111/4 — E114/4  
 E10/4 — E22/4 — E36/4 — E42/4 — E51/4 — E57/4 — E66/4 — E72/4  
 E62/2 — E69/2 — E47/2 — E54/2 — E15/5 — E27/5

A5 — E78/12 — R6/2 — E96/3 — E87/3 — E81/3 — E102/3 — E111/3 — E114/3  
 E10/3 — E15/3 — E22/3 — E27/3 — E36/3 — E42/3 — E51/3 — E57/3  
 E66/3 — E72/3 — E62/5 — E69/5 — E47/5 — E54/5

A6 — E89/2 — R5/2 — E96/2 — E87/2 — E81/2 — E102/2 — E111/2 — E114/2  
 E10/2 — E15/2 — E22/2 — E27/2 — E36/2 — E42/2 — E51/2 — E57/2  
 E66/2 — E72/2 — E62/11 — E69/11 — E47/11 — E54/11

A7 — E89/4 — R4/2 — E96/1 — E87/1 — E81/1 — E102/1 — E111/1 — E114/1  
 E10/1 — E15/1 — E22/1 — E27/1 — E36/1 — E42/1 — E51/1 — E57/1  
 E66/1 — E72/1 — E62/14 — E69/14 — E47/14 — E54/14

A8 — E89/6 — R3/2 — E96/23 — E87/23 — E81/23 — E102/23 — E111/23 — E114/23  
 E10/23 — E15/23 — E22/23 — E27/23 — E36/23 — E42/23 — E51/23 — E57/23

E66/23 - E72/23 - E68/2 - E75/2 - E53/2 - E60/2

A9 — E89/8 - R2/2 - E96/22 - E87/22 - E81/22 - E102/22 - E111/22 - E114/22  
 E10/22 - E15/22 - E22/22 - E27/22 - E36/22 - E42/22 - E51/22 - E57/22  
 E66/22 - E72/22 - E68/<sup>5</sup>22 - E75/5 - E53/5 - E60/5

A10 — E89/10 - R1/2 - E96/19 - E87/19 - E81/19 - E102/19 - E111/19 - E114/19  
 E10/19 - E15/19 - E22/19 - E27/19 - E36/19 - E42/19 - E51/19 - E57/19  
 E66/19 - E72/19 - E68/11 - E75/11 - E53/11 - E60/11

DIN 1 - E16/2 - R19/2 - E95/4 - E86/4 - E80/4 - E101/4 - E110/4 - E113/4  
 E9/4 - E14/4 - E21/4 - E26/4 - E35/4 - E41/4 - E50/4 - E56/4  
 E65/4 - E71/4 - E91/4 - E61/4 - E31/4 - E6/4

DIN 2 - E16/4 - R18/2 - E95/7 - E86/7 - E80/7 - E101/7 - E110/7 - E113/7  
 E9/7 - E14/7 - E21/7 - E26/7 - E35/7 - E42/7 - E51/7 - E57/7  
 E66/7 - E72/7 - E91/7 - E61/7 - E31/7 - E6/7

DIN 3 - E16/6 - R17/2 - E95/9 - E86/9 - E80/9 - E101/9 - E110/9 - E113/9  
 E9/9 - E14/9 - E21/9 - E26/9 - E35/9 - E42/9 - E50/9 - E56/9  
 E66/9 - E71/9 - E91/9 - E61/9 - E31/9 - E6/9

DIN 4 - E16/8 - R16/2 - E95/12 - E86/12 - E80/12 - E101/12 - E110/12 - E113/12  
 E9/12 - E14/12 - E21/12 - E26/12 - E35/12 - E42/12 - E51/12 - E57/12  
 E66/12 - E72/12 - E91/12 - E61/12 - E31/12 - E6/12

DIN 5 - E16/10 - R15/2 - E94/4 - E85/4 - E79/4 - E100/4 - E109/4 - E112/4  
 E8/4 - E13/4 - E20/4 - E25/4 - E34/4 - E40/4 - E49/4 - E55/4  
 E64/4 - E70/4 - E106/4 - E76/4 - E46/4 - E18/4

DIN 6 - E16/12 - R14/2 - E94/7 - E85/7 - E79/7 - E100/7 - E109/7 - E112/7  
 E8/7 - E13/7 - E20/7 - E25/7 - E34/7 - E40/7 - E49/7 - E55/7  
 E64/7 - E70/7 - E106/7 - E76/7 - E46/7 - E18/7

DIN 7 - E23/2 - R13/2 - E94/9 - E85/9 - E79/9 - E100/9 - E109/9 - E112/9  
 E8/9 - E13/9 - E20/9 - E25/9 - E34/9 - E40/9 - E49/9 - E55/9  
 E64/9 - E70/9 - E106/9 - E76/9 - E46/9 - E18/9

DIN 8 - E23/4 - R12/2 - E84/12 - E85/12 - E79/12 - E100/12 - E109/12 - E112/12

E81/12 - E13/12 - E20/12 - E25/12 - E34/12 - E40/12 - E49/12 - E55/12

E64/12 - E70/12 - E106/12 - E76/12 - E46/12 - E18/12

D0 - E28/3 - E84/3 - E93/3 - **AC1** конектор на бокеplane

D1 - E28/4 - E84/4 - E93/4 - **AD2**

D2 - E28/7 - E84/6 - E93/6 - **AD1**

D3 - E28/8 - E84/11 - E93/11 - **AE2**

D4 - E28/13 - E84/13 - E93/13 - **AE1**

D5 - E28/14 - E84/14 - E93/14 - **AF2**

D6 - E28/17 - E90/3 - E99/3 - **AF1**

D7 - E28/18 - E90/4 - E99/4 - **AH2**

D8 - E90/6 - E99/6 - **AH1**

D9 - E90/11 - E99/11 - **AJ2**

D10 - E92/6 - E99/13 - **AJ1**

D11 - E92/11 - E99/14 - **AK2**

D12 - E92/13 - **AK1**

D13 - E92/14 - **AL2**

D14 - E105/14 - **AL1**

D15 - E105/15 - **AM2**

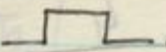
RDY - E108/9 - E108/5 - E108/2 - **BR1**

Повтор на BERA конектор (J2)

B D F J L N R T V X Z BB DD FF JJ LL NN RR TT VV  
A C E H K M P S U W Y AA CC EE HH KK MM PP SS UU

K1 - A	K5 - E	K9 - K	K13 - P	K17 - U	K21 - Y	K25 - CC	A3 - HH	A7 - MM	GND - SS
K2 - B	K6 - F	K10 - L	K14 - R	K18 - V	K22 - Z	A0 - DD	A4 - JJ	A8 - NN	CN - TT
K3 - C	K7 - H	K11 - M	K15 - S	K19 - X	K23 - AA	A1 - EE	A5 - KK	A9 - PP	GND - UU
K4 - D	K8 - J	K12 - N	K16 - T	K20 - X	K24 - BB	A2 - FF	A6 - LL	A0 - RR	GND - VV

# OPIS DELOVANJA CONTROL STORE MODULA Z ROM EMULATORI

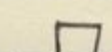
Povezava na računalnik 11/34 ali 11/23 gre preko modula DR 11 C li DEV. Na modul vstopa 16 podatkovnih vhodov D0 - D15 in RDY signal .

Podatki se vpisujejo v naslednje registre:

CSR (12 bitov)

ADR REG (14 bitov)

DATA REG (8 bitov)

Podatki D0 - D15 vsebujejo informacijo, ki se vpisuje v določen register in odraža tega registra. Adreso registra določata D15 in D14, ki vstopata v E105 (7442) dekodler. Ko pride signal RDY  in podatki preujejo v izhoni register. Vpisni signali za registre so

LOAD DATA REG

LOAD CSR

LOAD ADR REG

D15	D14
0	0
0	1
1	0

Ob vpran - adresni reg. s signalom LOAD ADR REG se sproži monovalentni multivibrator E109 (74121), ki z impulsom MON krmili  $\overline{CS}$  linije tako da troni signal  $\overline{MON} \& \overline{CSR} = \overline{CS}$

## CSR

D15 = 0 D14 = 1

CSR1 izhina  $\overline{CS}$  linije za CONTROL STORE v konjunkciji - MON.

CSR2 krmili  $\overline{OE}$  linije za CONTROL STORE

CSR3 krmili direktne vhod v E97 (74157) dekodler, ki služi za izbiranje poravnanih RAM-ov v CONTROL STORE DELU.

CSR4 služi za adresiranje zgornje polovice 1K-2K dela

pozicijskega prostora v CONTROL STORE.

CSR5 Notovlja MUX na položaj odčitavanja iz odčitavega registra odčitavanje iz zunanjega vira

CSR6 } krmilijo dekoder E98 (7442) ki služi za načrtovanje vpisa  
 CSR7 } (krmiljenje WE) v ROM emulatorje 1, 2, 3, 4  
 CSR8 }

CSR9 krmili SELECT linije ROM EMULATORSEU za upravljanje spujaj z MON signalom

		SIGNALNA RAM-u	FUNKCIJA																		
CSR1	0	$\overline{CS} = 1$	CONTROL STORE JE DISABLIRAN																		
	1	$\overline{CS} = \square$	v trajanju MON impulza. To je čas vpisa																		
CSR2	0	$\overline{OE} = 1$	Control store imo disablirane izhode																		
	1	$\overline{OE} = 0$	Izhodi iz control store so enabledi																		
CSR3	0	$\overline{WE} = \square$	RAMi ne izbirajo glede na odrese D13, D12, D11, D10																		
	1	$\overline{WE} = 1$	WE vhod v RAM-u v H to pomeni CONT. STORE JE DISABLIRAN																		
CSR4	0	ADR10 = 0	Izbran je spodnji del pomnilnika																		
	1	ADR10 = 1	—  — zgornji del —  — } velja samo ob vpisu																		
CSR5	0	SELECT S = 0	odčitavanje iz registrov - VPISOVANJE																		
	1	S = 1	odčitavanje externo - ČITANJE																		
CSR6			<table border="1"> <tr> <th colspan="2">CSR8 7 6</th> <th></th> </tr> <tr> <td>0 0 0</td> <td>VPIS = H</td> <td></td> </tr> <tr> <td>0 0 1</td> <td>VPIS 1 = 0</td> <td>IZBRAN JE ROM EMULATOR 1</td> </tr> <tr> <td>0 1 0</td> <td>VPIS 2 = 0</td> <td>————— " ————— 2</td> </tr> <tr> <td>0 1 1</td> <td>VPIS 3 = 0</td> <td>————— " ————— 3</td> </tr> <tr> <td>1 0 0</td> <td>VPIS 4 = 0</td> <td>————— " ————— 4</td> </tr> </table>	CSR8 7 6			0 0 0	VPIS = H		0 0 1	VPIS 1 = 0	IZBRAN JE ROM EMULATOR 1	0 1 0	VPIS 2 = 0	————— " ————— 2	0 1 1	VPIS 3 = 0	————— " ————— 3	1 0 0	VPIS 4 = 0	————— " ————— 4
CSR8 7 6																					
0 0 0	VPIS = H																				
0 0 1	VPIS 1 = 0	IZBRAN JE ROM EMULATOR 1																			
0 1 0	VPIS 2 = 0	————— " ————— 2																			
0 1 1	VPIS 3 = 0	————— " ————— 3																			
1 0 0	VPIS 4 = 0	————— " ————— 4																			
CSR7																					
CSR8																					
CSR9	0	S = 0	Upravljanje v RAMe ROM emulatorjev																		
	1	S = 1	Čitanje iz ROM emulatorjev																		



**DATA REGISTER**

D15=0, D14=0

Podobno iz registra paralelno vodeni na vse 110 Buffere za ROM-ov. V DATA REG. uporabljamo osmi osem bitov D0 ÷ D7

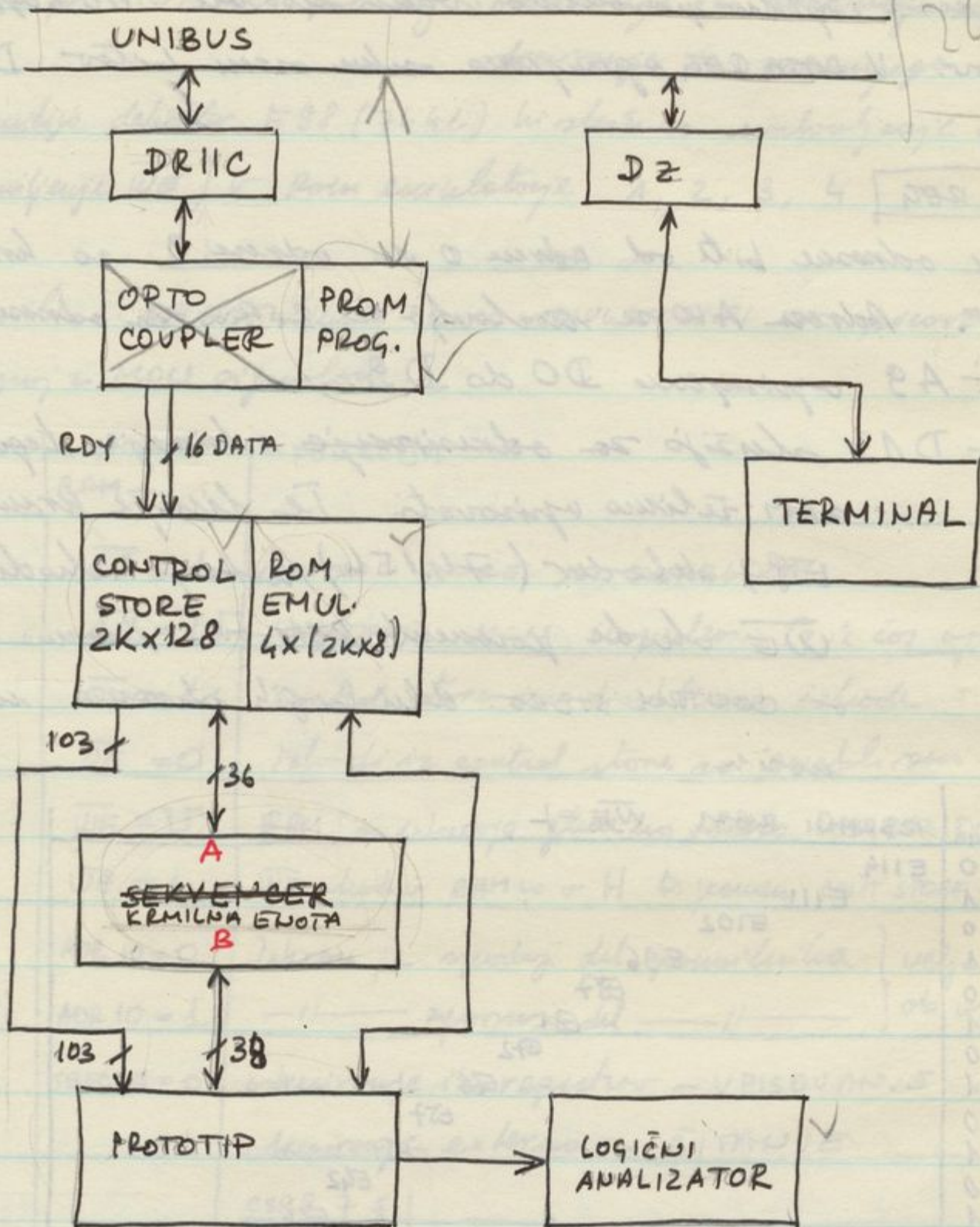
**ADRES REG.**

Upinje adresne biti od Adrese 0 do adrese 9 so krmljenje RAMOV. Adresa A10 u. uveljavlja iz CSR. Za adrese A0 ÷ A9 uporabljamo D0 do D9.

D10 - D13 služijo za odkrivanje izklopa tega v kateri RAM želimo upinovati. Te linije krmlijo E97 dekodler (74154) katerega izhodi uveljavljajo  $\overline{WE}$  vhodov poročenih RAM-ov. Imamo 16 RAM-ov v CONTROL STORO. deli ki jih shovamo na sledeč način

	D13	D12	D11	D10	IZBRANI RAM	$\overline{WE} = L$
0	0	0	0	0	E114	
1	0	0	0	1	E111	
2	0	0	1	0	E102	
3	0	0	1	1	E96	
4	0	1	0	0	E87	
5	0	1	0	1	E81	
6	0	1	1	0	E72	
7	0	1	1	1	E66	
8	1	0	0	0	E57	
9	1	0	0	1	E51	
10	1	0	1	0	E42	
11	1	0	1	1	E36	
12	1	1	0	0	E27	
13	1	1	0	1	E22	
14	1	1	1	0	E15	
15	1	1	1	1	E10	
	0/0	0/1	0/1	0/1		

RAZVOJNI SISTEM ZA BIT SLICE



**A**    11    ADRESE  
       25    DATA  


---

       36

**B**    14    TEST VHODI  
       4    2, 3, 4, 5  
       12    BR BUS  
       4    OR TEST  
       4    2 3 4 5  


---

       38

SEQUENCER KEMILNA ENOTA

Stavec se lahko prebrava iz PLR. Te linije bi pravočasno štavec se voolijo preko T.S. izhodov tudi na D linije enote 2911. Torej isto polje PLR se uporablja za next address filit omenjen branch filit in za COUNT vrednost. To je FORMATRAMS

Multipleksir testnih vhodov je realiziran tako do v enoti 29811 - (next address control unit) vlogo prova ali po nepripravno vrednost testnega vhoda. To omogoča, da se pojavni skoki izvaja npr ob erry set ali carry reset stanju.

Test vhod 0 je vezan na GND. To nam omogoča način da pojavne instrukcije pretvorimo v brezpojvne. To pomeni: katerokoli pojavno instrukcijo lahko izvajamo brezpojvno s tem da izberemo <sup>testni</sup> vhod 0, ki je vezan na GND in forniramo krmiljenje polaritete v invertirano ali neinvertirano stanje. To nam omogoča izvrševanje brezpojvnega JUMP, brezpojvnega JUMP TO SUBROUTINE in brezpojvnega RETURN FROM SUBROUT.

EXTENDED ENABLE CONTROL

Izvedeno z multipleksiranjem in 29811 (next address control) enoti. 29811 ima osem izhodov. S štirimi odhodi krmili 2911 enoto štirih izhodov po odvisno na 4-8 linijstok multipleksir. To so izhodi PLE, MAPE, CNT LOAD in CNT ENABLE

Kodori sta PLE in MAPE v LOW, potem sta 2Y1 in 2Y2 v LOW kdaj sta enob-  
lirna ali PLR branch odres ali po mapping PROGRAM. Če sta PLE  
in MAPE v HIGH potem je izhod 2Y3 in U16 v LOW s čemer  
quodlira T.S. izhode iz map pravno za določanje alternativne  
<sup>MAP</sup> odrese. (to je vektorški vhod.). Podatno je omogočen  
CNT LOAD in CNT ENABLE. Ti omogočajo stavec do n loada ali  
do skje preko izhodov 1Y1 in 1Y2.

VEKTORSKI VHOD

Možno uporaba za interrupt vektor (Interrupt request izhodi iz 2814 Vectorskega Priority Interrupt kontrolerja in lahko uporablja kot pogojni vhod v mux. Vektorske izhodne linije pa iz 2814 ustrezajo preho multiplexovane na bus. Mikroprocesor lahko potem skozi direktus v rutino hi strni posameznim interruptu. Druga možnost je da se ga uporablja za izvirnoje TROP rutin.

16 WAY BRANCH

Enota 28803 ima kot 16 way branch control, kadar se uporablja v povezavi z 2809 mikroprocesorskim shemencem AM28803 ima 4 test vhode,

- 4 instrukcijske vhode → <sup>Triglova kontrola na TEST</sup> vhodih in OR izhodih
  - 4 OR izhode → te vsemo direktus na 2802 OR vhode
- enable control

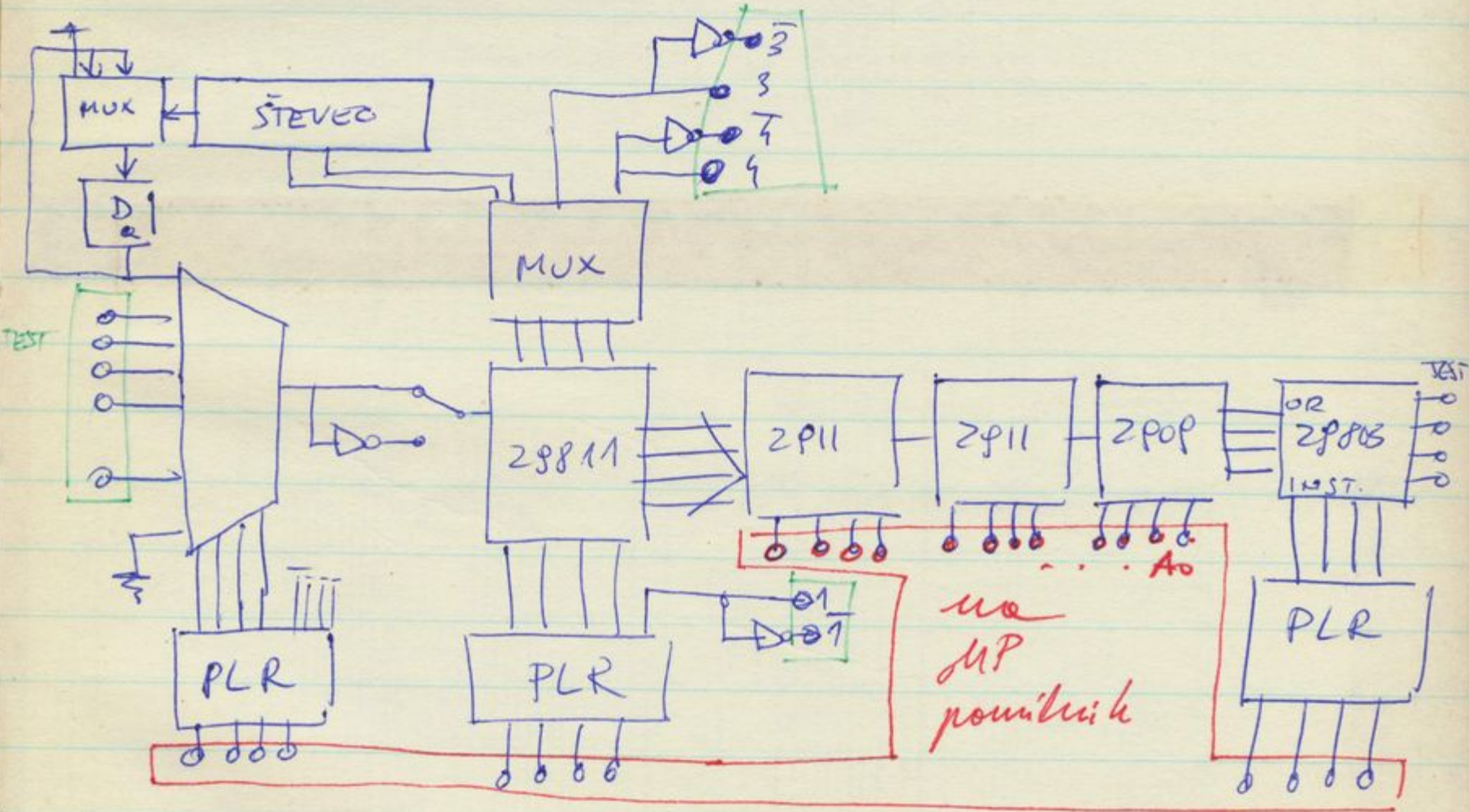
Instrukcijski vhodi I<sub>0</sub> - I<sub>3</sub> omogočajo 16 instrukcij F<sub>0</sub> - F<sub>15</sub>, ki lahko izbirajo 16 možnih kombinacij TEST vhodov in dojejo določen izhod no OR izhodih ki je odvisen od stanja vhodov ki jih testiramo.

- Instrukcija 0 ne testira nobenega vhoda. To je disable instrukcija.
- Instrukcija 1, 2, 4, 8 testira enega izmed vhodov in izvaja branch no eno izmed dveh bered.
- Instrukcij 3, 5, 6, 9, 10, 12 testirajo 2 vhoda in lahko določimo no eno izmed 4 bered (4 beredna stran)
- Instrukcije 7, 11, 13 in 14 testirajo 3 vhode in lahko določimo no stran dolgo 8 bered
- Instrukcija 15 testira 4 vhode in lahko določimo no 16 bered dolgo stran.

Instukcije relociraneja 2909/2911 z Duhodli (JRP, JSRP, JP in CJS so idealne za ~~288~~ <sup>TEST</sup> OK zamenjati, ki jo daje 29803.

Shema na koliciji, ki je mogoča preko BR0-BR11 di<sup>no</sup> register, ki je v 2909/2911 lahko uraduje odnosa BRANCH tabele.

Branch tabela je relocirana serija brezpopojnih jump instrukcij, katere jump instrukcije po nbo izvršila po je določeno s spodnjimi adresnimi biti. Najmanj polje za 8 way branch mora imeti spodnje 3 bite nula 0. Seveda po zowosti tega to polje niso v ~~spodnje~~ relocotabe honorholi v spominu. Spodnji adresni biti morajo biti vedno 0 po gledi od določene polja.



na prototip sistemu.

R20	R19	R18	R17	SELECT
0	0	0	0	TEST 0
0	0	0	1	TEST 1
0	0	1	0	TEST 2
0	0	1	1	TEST 3
1	1	1	1	TEST 15

R16	OUTPUT
0	COMPLEMENT OF TEST
1	TRUE TEST

R15	R14	R13	R12	FUNCTION
X	X	X	X	NEXT INSTRUCTION

R21	FUNCTION
0	LOAD
1	HOLD

R11-R0	FUNCTION
XXXX...XXX	VALUE

BR11-BR0	FUNCTION
XXXX...XXX	JUMP ADDRESS

R25	R24	R23	R22	FUNCTION
X	X	X	X	TEST INSTRUCTION

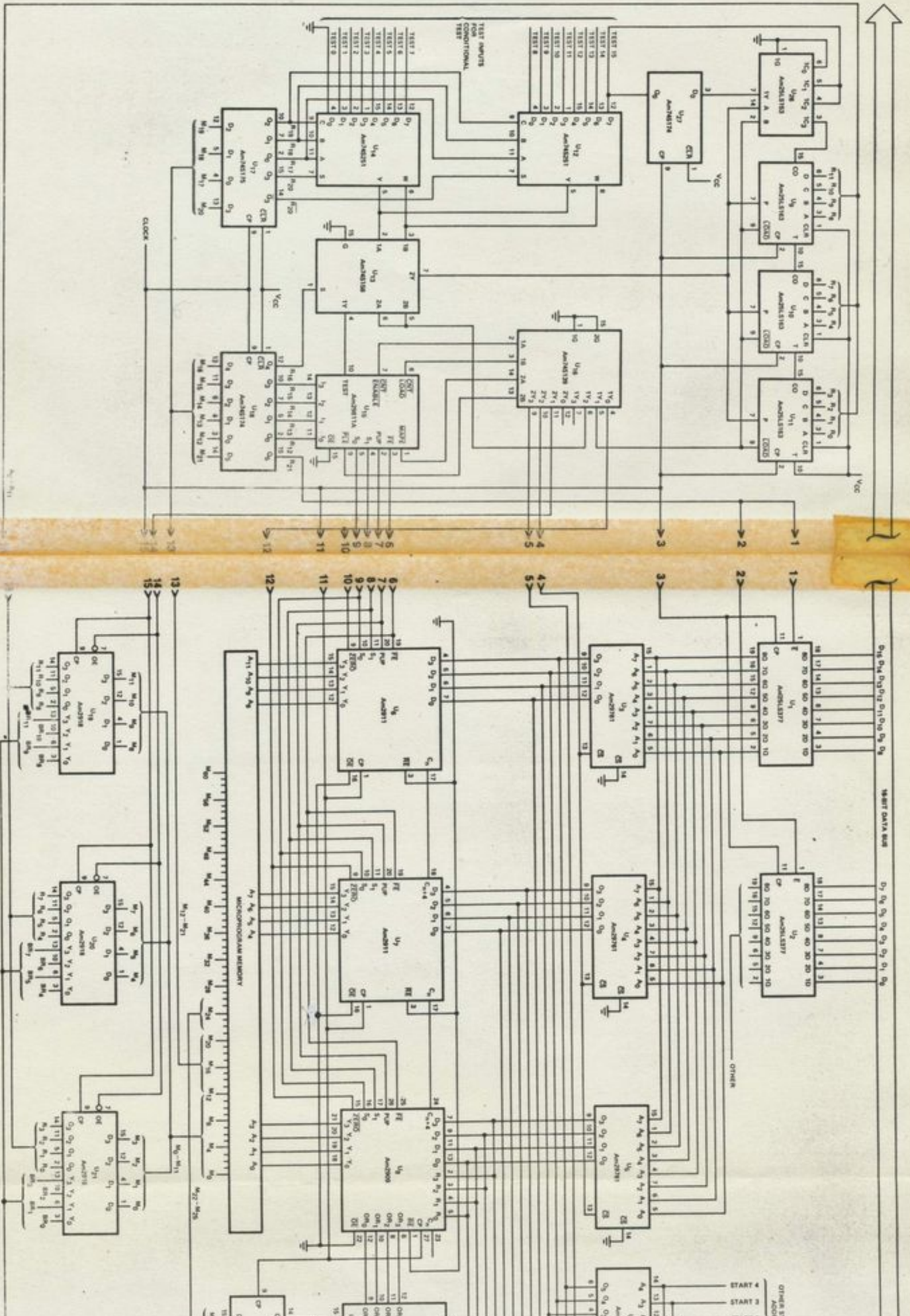


Figure 18. High Performance Computer Control Unit with Am2909/2911.

LECT  
ST 0  
ST 1  
ST 2  
ST 15

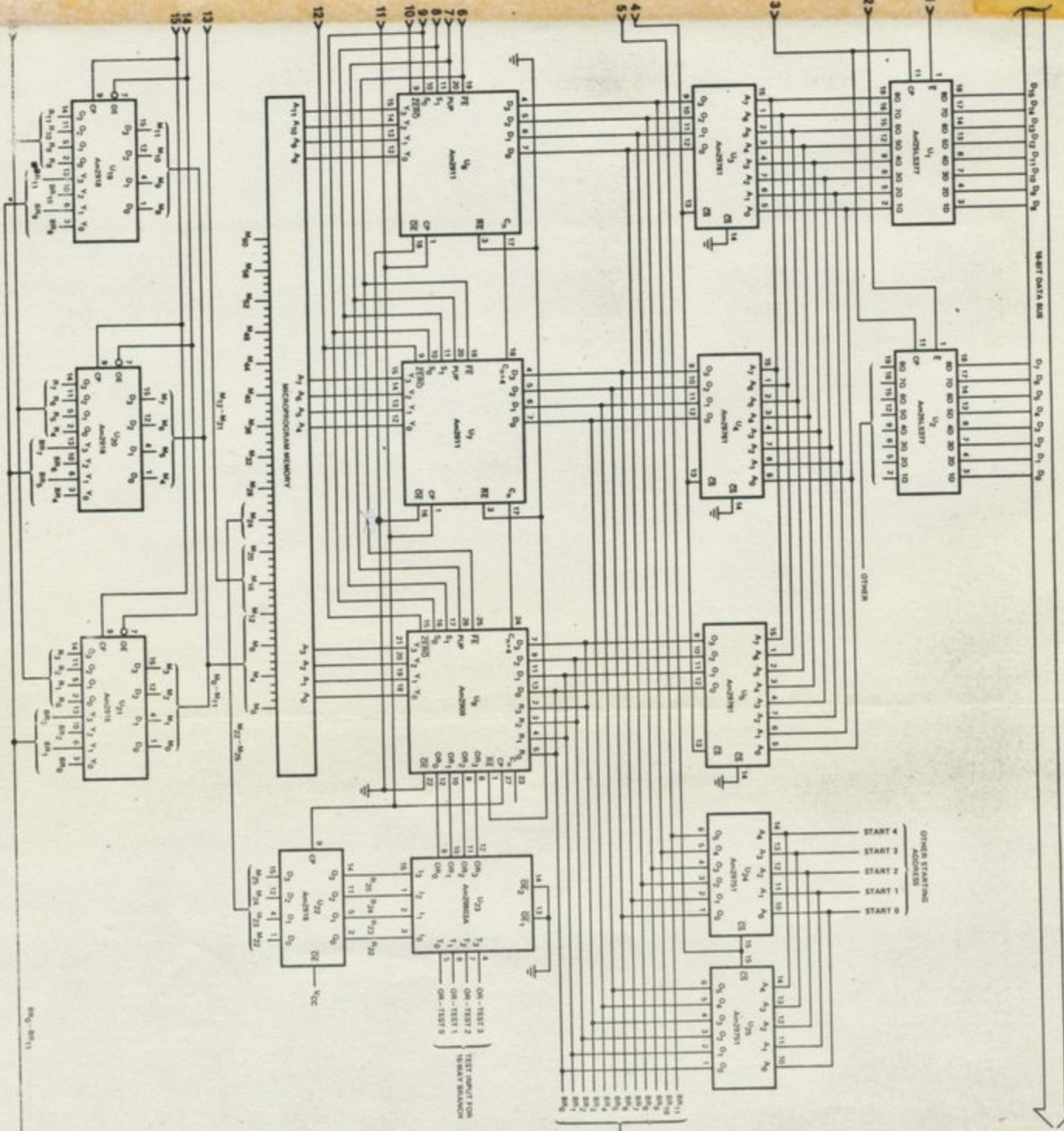
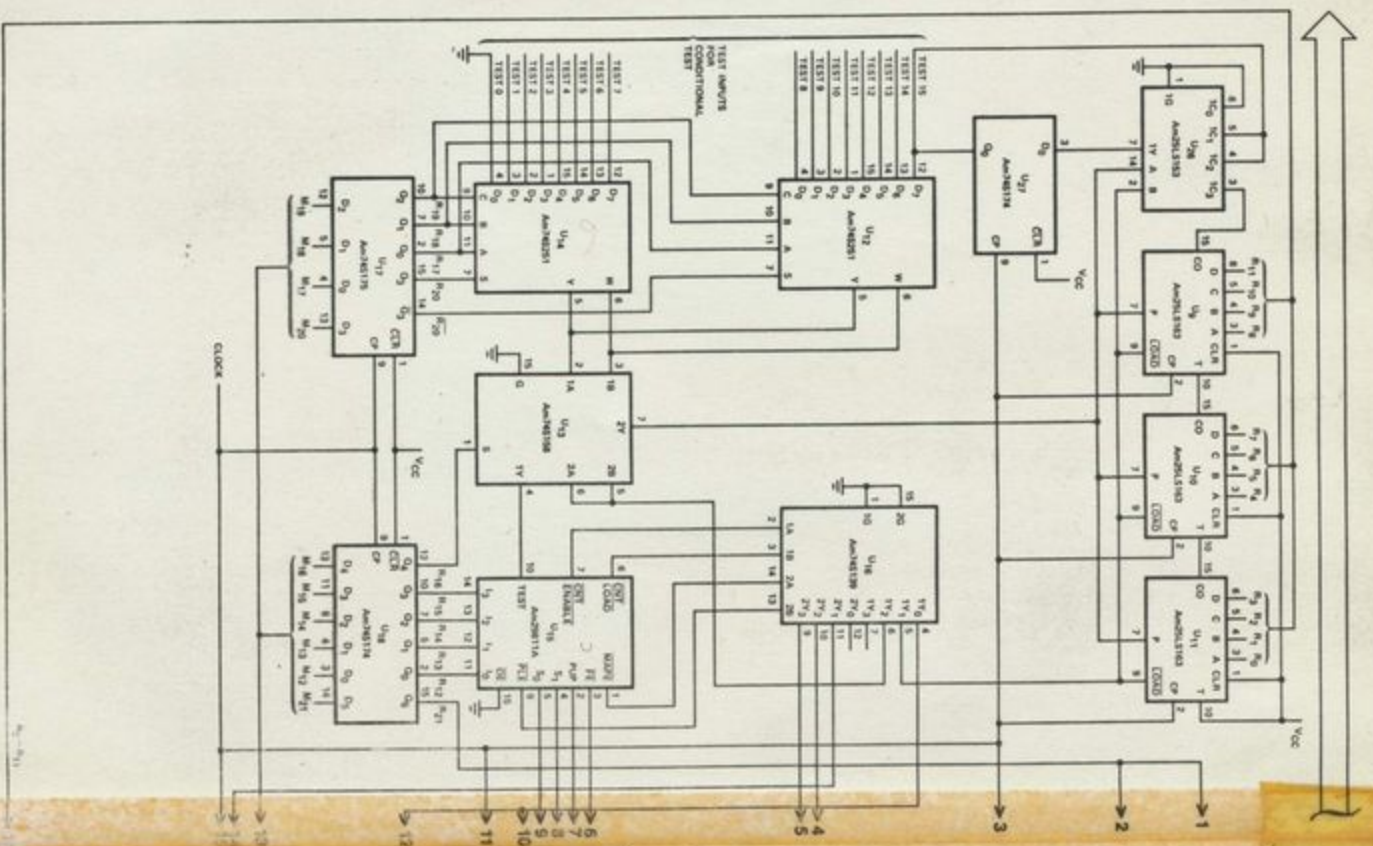


Figure 18. High Performance Computer Control Unit with Am2909/2911.

## DEKODIRANJE INSTRUKCIJ

Instrukcija se učitava no 16-bitnom podatkovnem vodilu (DATA BUS)  
 Od tu se upiše v INSTRUKCIJSKI REGISTER. Pogoj za upis je  
 IR ENABLE, ki pride iz PLR (3 Pipeline register). Upis se  
 izvrti ob urnem impulzu Cp.

Tehtod iz instrukcijskega registra tvori IR BUS, ki vodi podatke  
 no dekoderski ROM in no logiko za izbira dekoderskega ROMa.  
 Instrukcije se z ovrom no oblike ločijo no 7 tipov, ki jih  
 dekodiramo s tremi ROM-i:

IR	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
A	X	X	X	X	SM			S	DM			D				} ROM A, B		
B	0	X	X	X	X	X	X	R			DM		D					
C	X	0	0	0	X	X	X	BRANCH OFFSET										} ROM C, D
D	X	0	0	0	X	X	X	X	X	DM			D					
E	0	0	0	0	0	0	0	0	X	X	X	X	R ZVC					} ROM E, F, G
F	0	0	0	0	0	0	0	0	X	X	X	X	R					
G	0	0	0	0	0	0	0	0	X	X	X	X	X	X	X	X		

P1 000

Logika za izbira dekoderskega ROMa obkretno

P2 0

priznatost pogojev P1, P2 in P3 v instrukciji

P3 000

ROM A, B se izbira pri pogoju  $\overline{P1}$ :

$$D1 = \overline{P1} = 14 \vee 13 \vee 12$$

$$D2 = P1 \& \overline{P3} = (\overline{14} \overline{13} \overline{12})(11 \vee 10 \vee 9)$$

$$D3 = P1 \& P2 \& P3 = \overline{15} \overline{14} \overline{13} \overline{12} \overline{11} \overline{10} \overline{9} \overline{8}$$



Rom za dekodiranje instrukcij imaju format 256 x 8. Neke veljane instrukcije koje dekodiraju u određeno 8 bitno vrednost. Neveljavne instrukcije dekodiraju u vrednost 000<sub>8</sub>.

Zbog iz IR DEC ROM-ov ne veruju u MBUS (MAPPING BUS), ali koriste MAPPING ROM.

Mapping ROM dolazi uzdu 8 bitni kod instrukcije 12 bitno MAP adresu, koji vodi na BRANCH BUS koristeći svaki procesor. Zbog iz Mapping ROM-a u mobilnoj i signalu  $\overline{MAPE}$ , koji iz AM 28811A svaki preko multiplexinga za router.

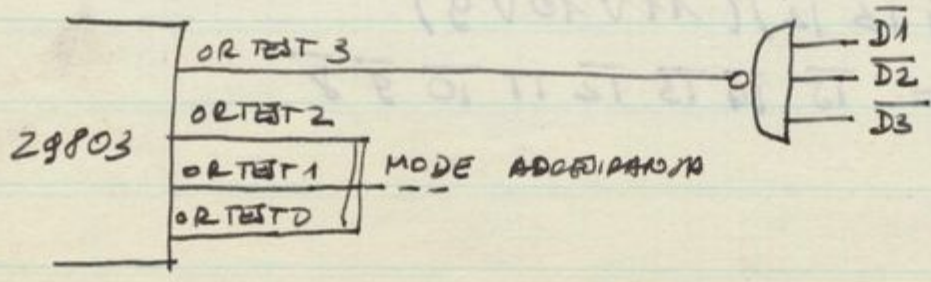
IR BUS vodi u selekcionu mrežu, koji iz instrukcije izdaje Adresu način za SOURCE in DESTINATION. Koristi signali za određuje SM in DM pridaju iz PLR in to to:

SEL1, SEL2 in SEL3. Vreće izdaje podneke linije u koje uključuju u IR linijah 11, 10, 9 za SM in IR 5, 4, 3 za DM.

Te podneke poteri vodi u OR TEST uloze svake AM 28803, koji izdaje multiway branch u određeno lokacij u svaki odredn rutine za izdaje instrukcije u doticnim adresama načinom.

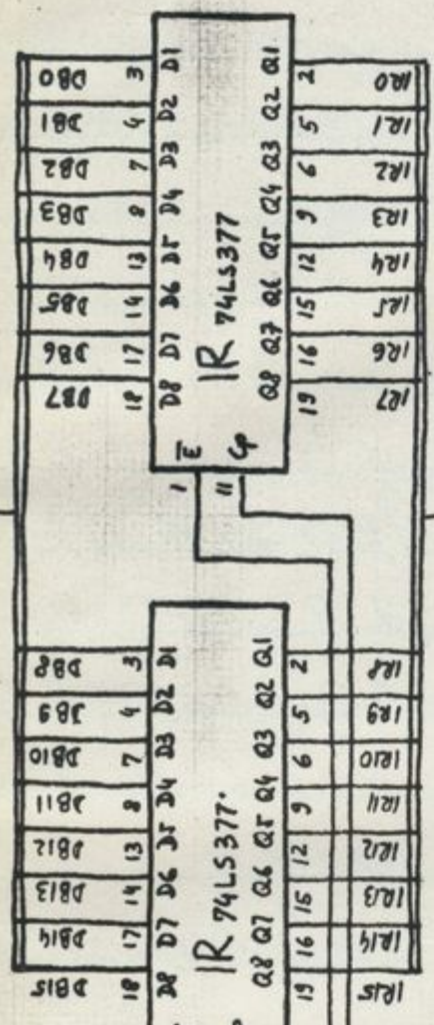
DEKODIRANJE NEVELJAVNIH INSTRUKCIJ

- 1) Rom dekodira neveljavne instrukcije u vrednost 000<sub>8</sub> tj je bit selektivan
- 2)  $\bar{0}$  rom ni selektivan: popoj  $\overline{D1} \& \overline{D2} \& \overline{D3} = 0$  lahko dočimo u svaki rutine preko multiway branch svake (Ulaz OR TEST 3)



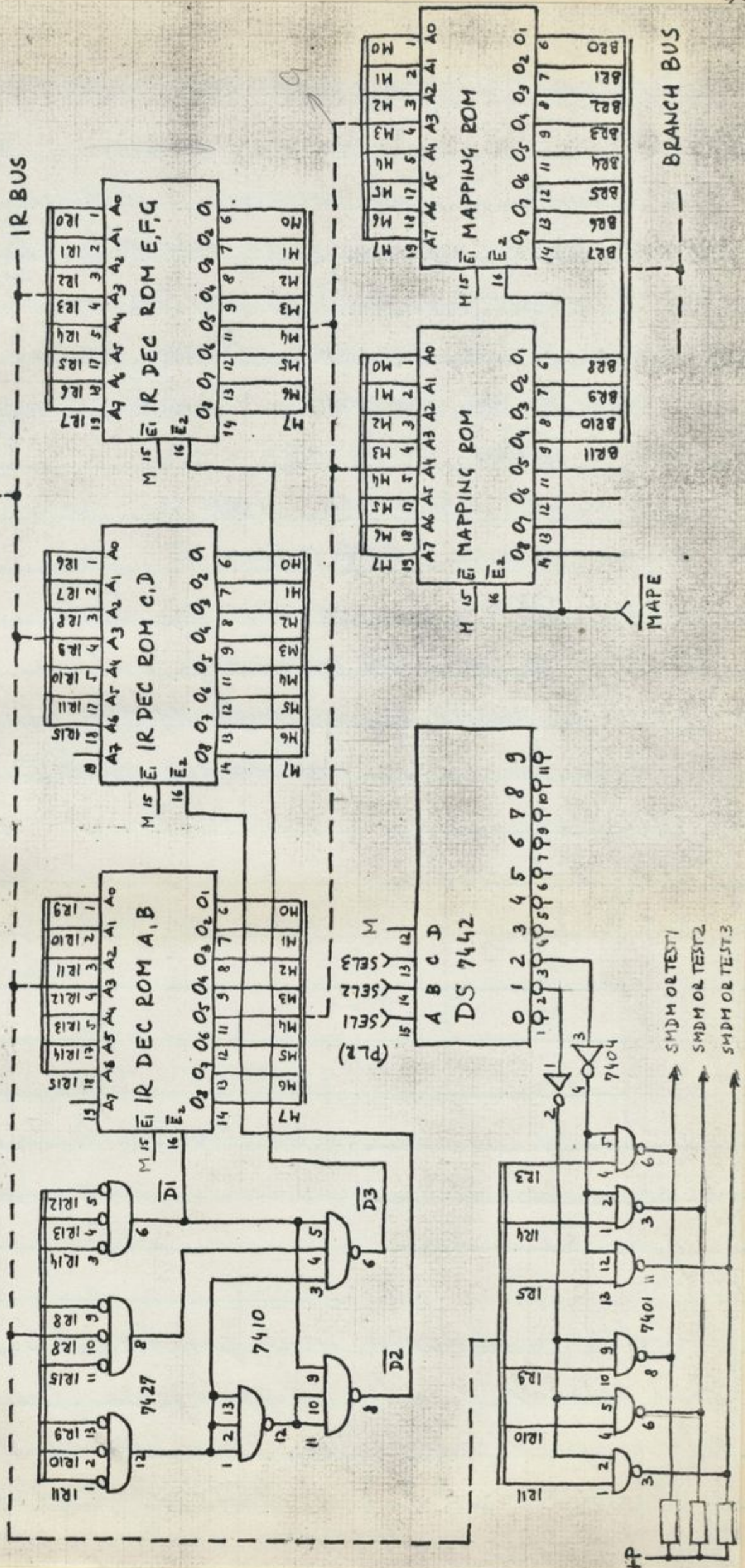
16 BIT DATA BUS

INSTRUKCYSKI DEKODER IN  
MAPPING PROM



(PL2) IR ENABLE  
CP

IR BUS



BRANCH BUS

### ADRESIRANJE SCRATCHPAD POMNILNIKA

Scratchpad adres mux služi za adresiranje scratchpad pomnilnika v AM2903 enoti. Ta ima adresiranje z adresno A, skotno lahko citamo katerikoli register pomnilnika in adresno B, skotno lahko istočasno citamo drug reg v pomnilniku in ovaj tudi vpisujemo.

Na adresno A vodimo adresno SOURCE registra,  
Na adresno B vodimo izhodi iz multiplexerja, ki dobavlja naslednje adresse

- 1) Adresno DESTINATION REGISTRA      biti    IR 2, IR1, IR0
- 2) Adresno SOURCE REGISTRA          biti    IR 8, IR7, IR6
- 3) Adresno PBA                            biti    BA3, BA2, BA1, BA0

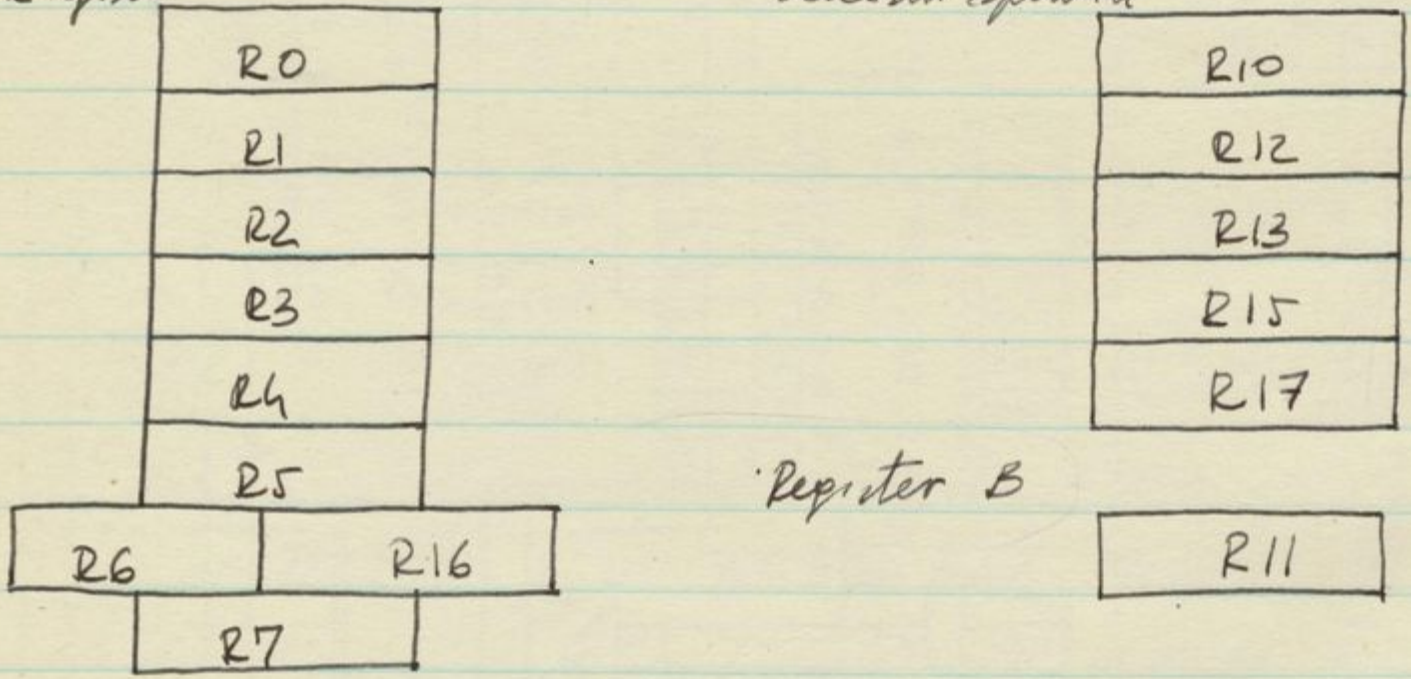
ta služi za direktno vpisovanje vredine preko konzole.

- 4) Adresno iz mikroprogramskega ROM-a    biti: PLRSPA3, 2, 1, 0

Registri so urejeni po naslednjem redu:

SPR registri

Začasnii spomin

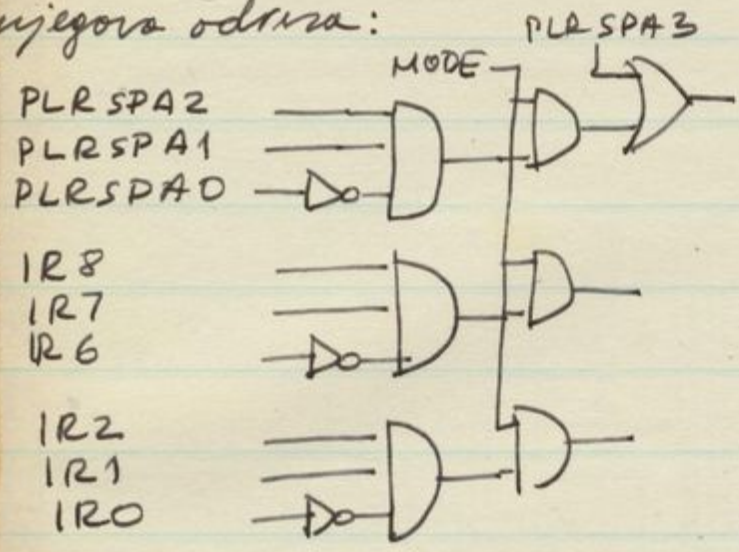


Register B

Kakor koli je adresiran stack pointer R6, R16 se mora najprej določiti v kateri reg vpisujemo glede na mode delovanja računalnika R6 - KERNEL MODE

R16 - USER MODE

Signal PLR MODE SELECT izliva NAČINU DELOVANJA, to je bit 15 in bit 13 iz PSW registra. Kador je bit odbranen stack pointer u detektivu njegova odbrana:



V odbranosti od načina odbranjenja putem ramena R6 do R16. Kador odbranimo SPM iz control store pomislitko lahko direktno odbranimo R6 oz R16 s signalom PLRSPA3.

Pri odbranjenju iz BA po s signalom PBA3 direktno izbranimo register v kolonju vpi zunje.

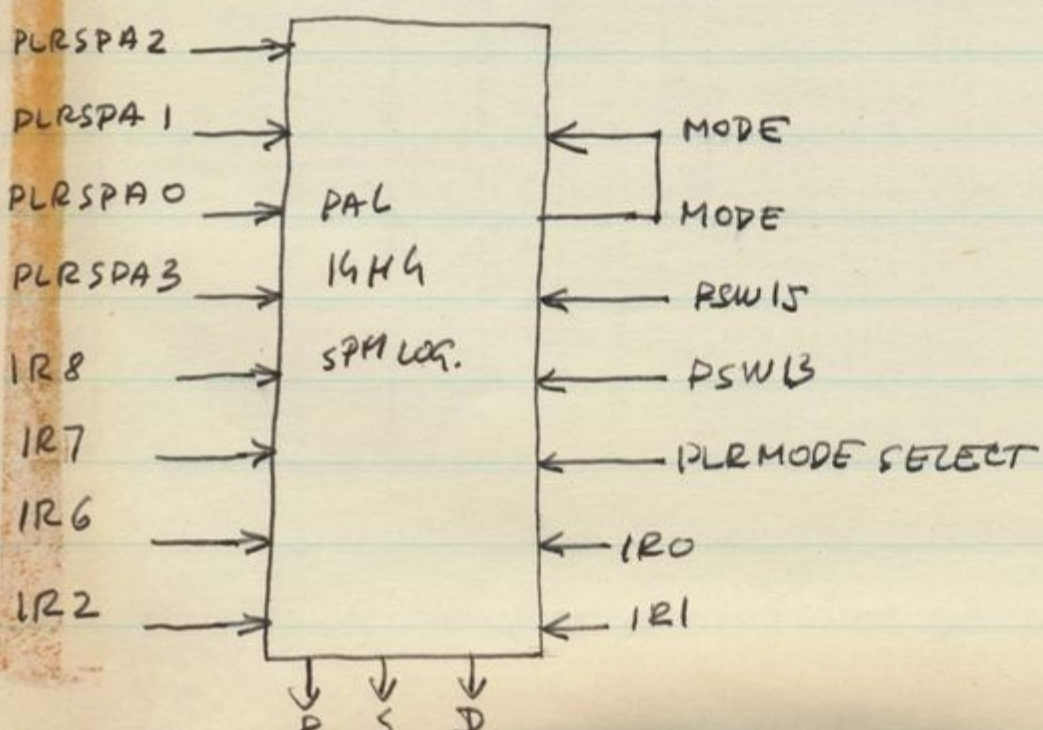
Verje na vodovljenju je R6/R16 je možno izdelati s 5 čipi misle stopnje integracije ali po 2 čipim PAL 16H4 vjelen, ki redziva vodovljenju funkcije:

$$P = \text{PLRSPA2} \cdot \text{PLRSPA1} \cdot /\text{PLRSPA0} \cdot \text{MODE} \vee \text{PLRSPA3}$$

$$S = \text{IR8} \cdot \text{IR7} \cdot /\text{IR6} \cdot \text{MODE}$$

$$D = \text{IR2} \cdot \text{IR1} \cdot /\text{IR0} \cdot \text{MODE}$$

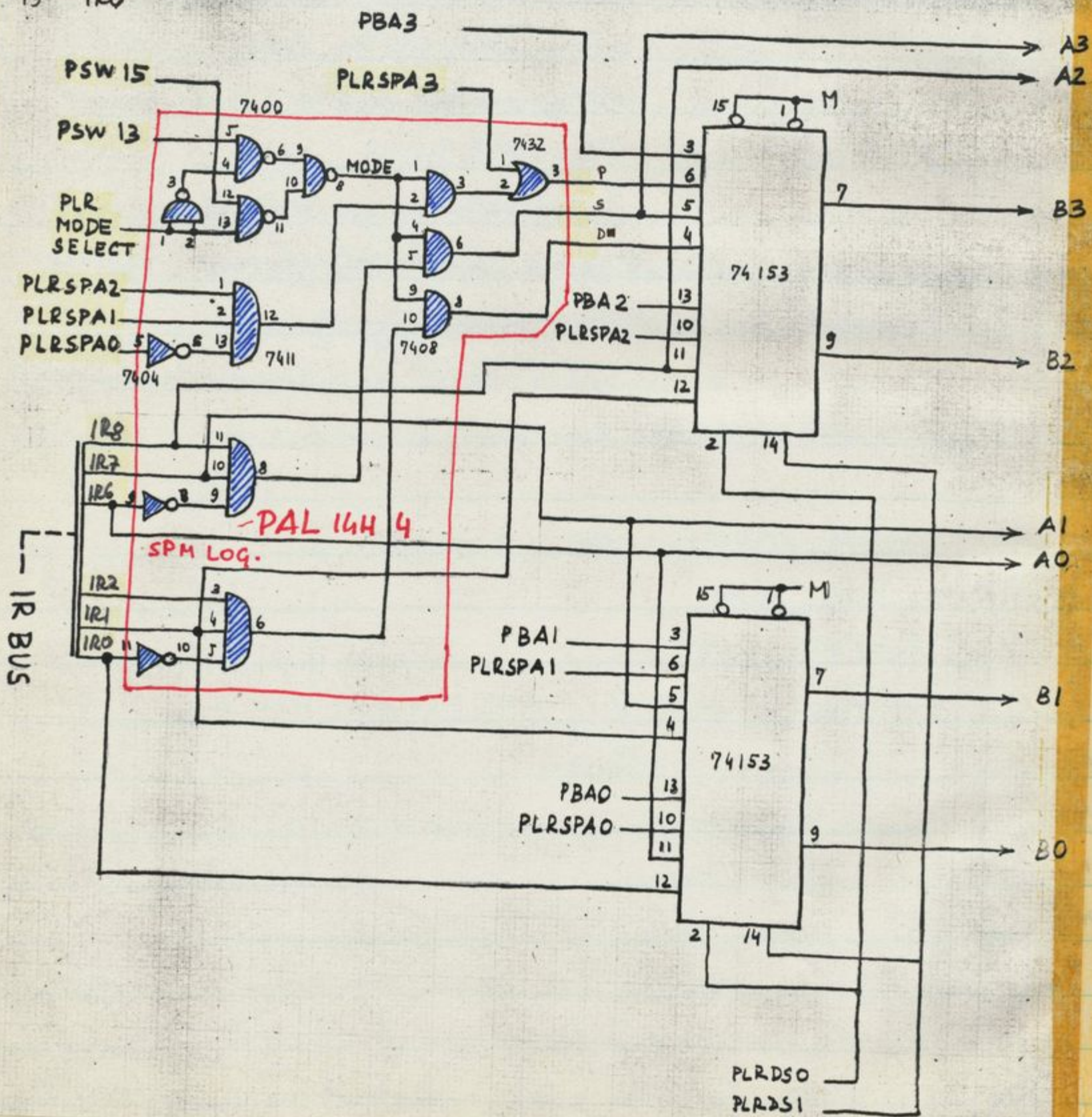
$$\text{MODE} = \text{PLRMODESELECT} \cdot \text{PSW15} \vee /\text{PLRMODESELECT} \cdot \text{PSW13}$$



# SCRATCH PAD ADRES MUX

- VHODI  
 1 PSW15  
 2 PSW13  
 3 PLR MODE SELECT  
 4 PLR SPA2  
 5 PLR SPA1  
 6 PLR SPA0  
 7 PLR SPA3  
 8 IR8  
 9 IR7  
 10 IR6  
 11 IR2  
 12 IR1  
 13 IR0

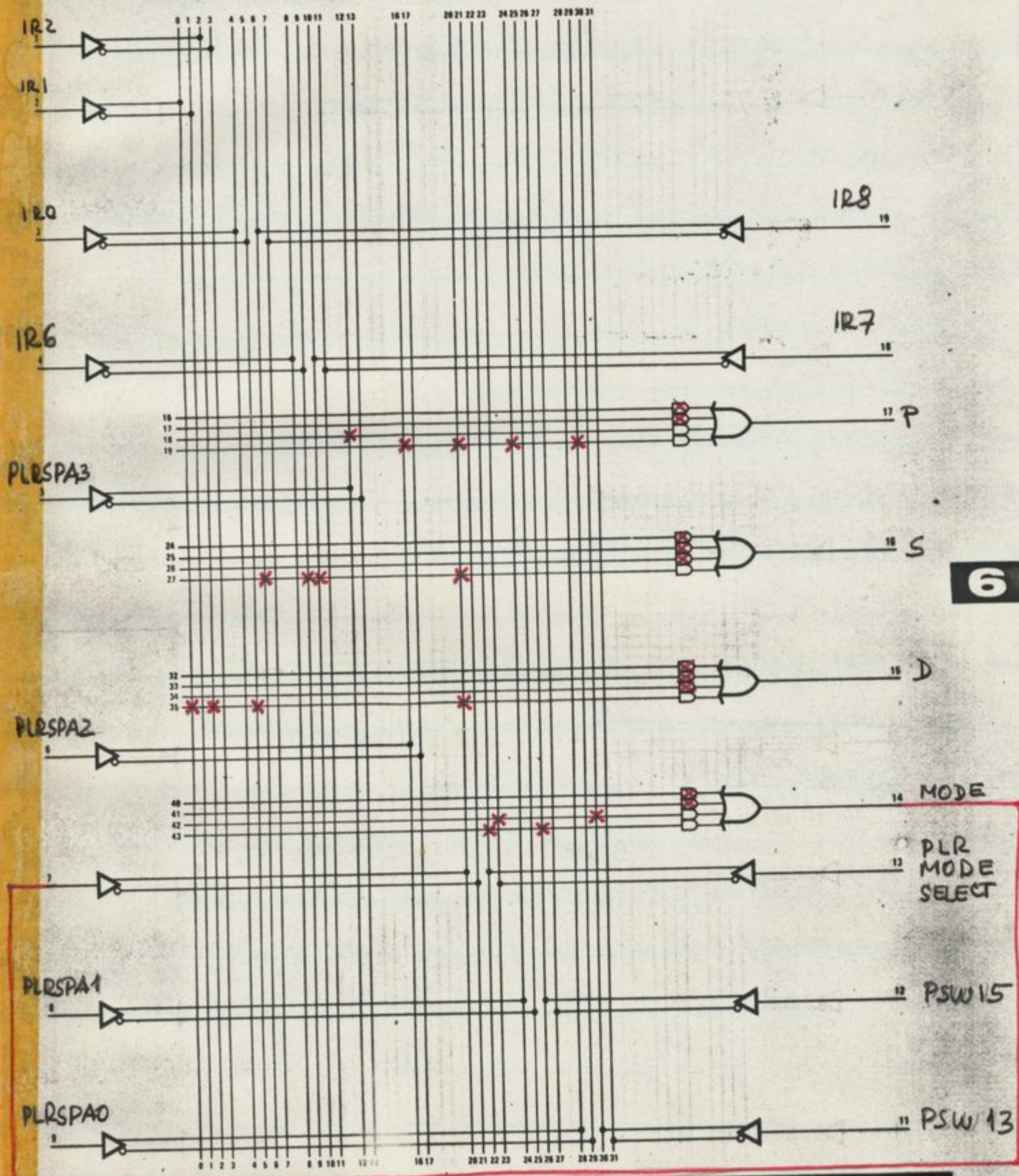
- 12MOD1  
 1 D  
 2 S  
 3 P



PAL Series 20

Logic Diagram PAL14H4

SPMLOG



6

Sestavek na fakulteti za elektrotehniko

FE Virout, Debnikar, Kodar, Uvelko

DETA Zeman, Kogovsek, Trebar

11.8.80 | molitna pogodba v dveh diplomah

rezultati 4 diplome, IC verija ki se uporablja tudi za 2900 serijo.

Mo kaj ulogi je se (21600) elektrotehnik je dostavilo se 1. ovo (1001)

Vogel, 1 laborant, tehnik (začetnik), 1 strojistka (začetnik), 3 diplome (2)

Virout, Pipern, Gustin, Kodar, Debnikar (občasno)

Pocunovo je doba 2 let s temi ljudmi za razhujitvo uologe.

- Delo na delitvenem sistemu zvo vovoj
- Vrovanje na selti
- Vogel, Kodar, + 1/2
- Sistem bo vezjetno 11/23 z dnevno delovanjem
- Prototipni sistem bo v eksperimentu bo en
- Dejnosti sreda julyja - sreda avgusta

Sestavek na FE

FE Virout, Debnikar, Kodar, Uvelko

DETA ~~Zeman~~ <sup>Mlatje</sup>, Kogovsek, Trebar.

Predlog pogodbe

\* Prenos sredstev 100M iz virendovalne skupnosti, ki pripada DETI na elctro fakulteto. z namenom vovojja bit sba poravnaja.

Glede na projekt med EP in Delto bi morali prenesti delov na FE.

Način dela?

- delijo definirano ulogo, ki jo avtonomno rešijo na svoji lozke

### BRANCH BUS

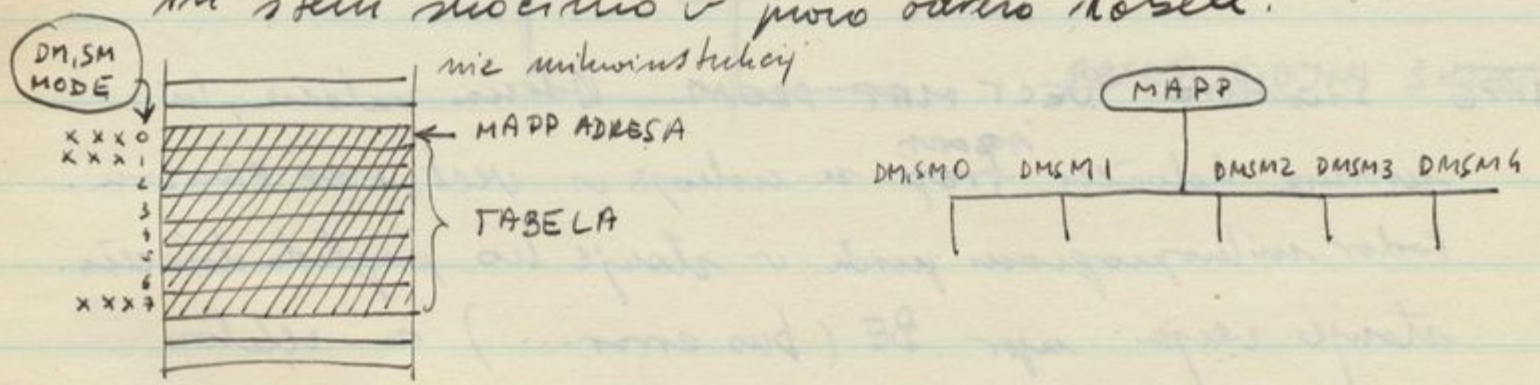
Ustopa v 2911 in 2903 enote in predstavljata možno odnosa za podpiranje u programnega pomnilnika. Kadar je izločena enota 29811:

$S0 = \text{1 (H)}$   
 $S1 = \text{1 (H)}$

Se prenese vrednost na BR bus na u ADDRS BUS. Enota 29811 lahko preko razširjenega multiplexerja 74139 ustvari 4 izvirne adresse na BR BUS:

29811	MAPE	PLE	LOAD	EN	S0 S1	MEMORIJA FUNKCIJA	KRMILNI ENABLER SIGNALI
	H	H	0	0	H H	C/JUMP VECTOR če je PLR 33 = 1 če je PLR 33 = 0	$\overline{\text{OP MAP}}$ $\overline{\text{VECT MAPE}}$
	H	L			H H	JUMP PIPELINE	$\overline{\text{PL BRE}}$
	L	H			H H	JUMP MAP	$\overline{\text{MAPE}}$

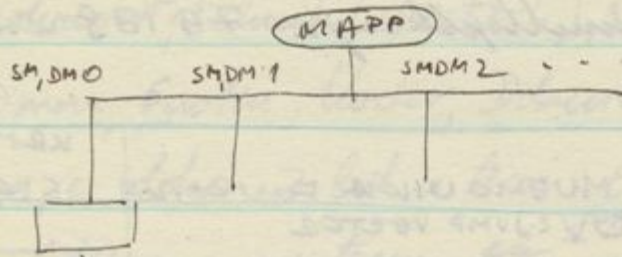
~~MAPE~~ odpirajo (ustvarijo) MAPPING PROM hi nam da startno odnosa rutine hi izvaja instrukcije. Glede na to startno odnosa potem določimo na določeno mesto v tabeli, hi jo določijo multibranch. 2 drugimi besedami je no odnosa hi jo daje MAPPING PROM začetek tabele, hi nam določajo DM (dest mode) in SM (source mode) DM ali SM prejetemo na SM DM OR TEST vhoda enote 29803 in stem določimo v pravo odnosa tabele.





OP MAP

enable OPERATOR DECODE MAP PROM. <sup>mikro</sup> Ko program pride do mesta, kjer se mora izvesti operacija med dvema operandoma: npr. R14(OP) R7, se na ALU mesto prijetje ~~ta~~ instrukcija iz PLR, ki se nahaja na adresi, ki je odredena OP MAP PROM preko BR BUS-a in D vhoda v mux mesto 2911/2909.

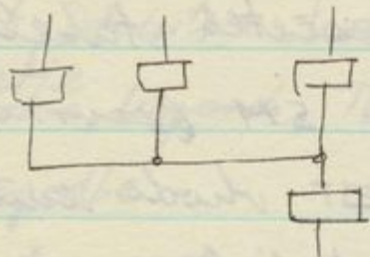


operacija je odvisna od tega katero instrukcija je v LR registru.

Enable OP MAP proma in izvirni ob PLR 33=1 in  $\overline{MAPE}=1$  in  $\overline{PLE}=1$

PL BRE

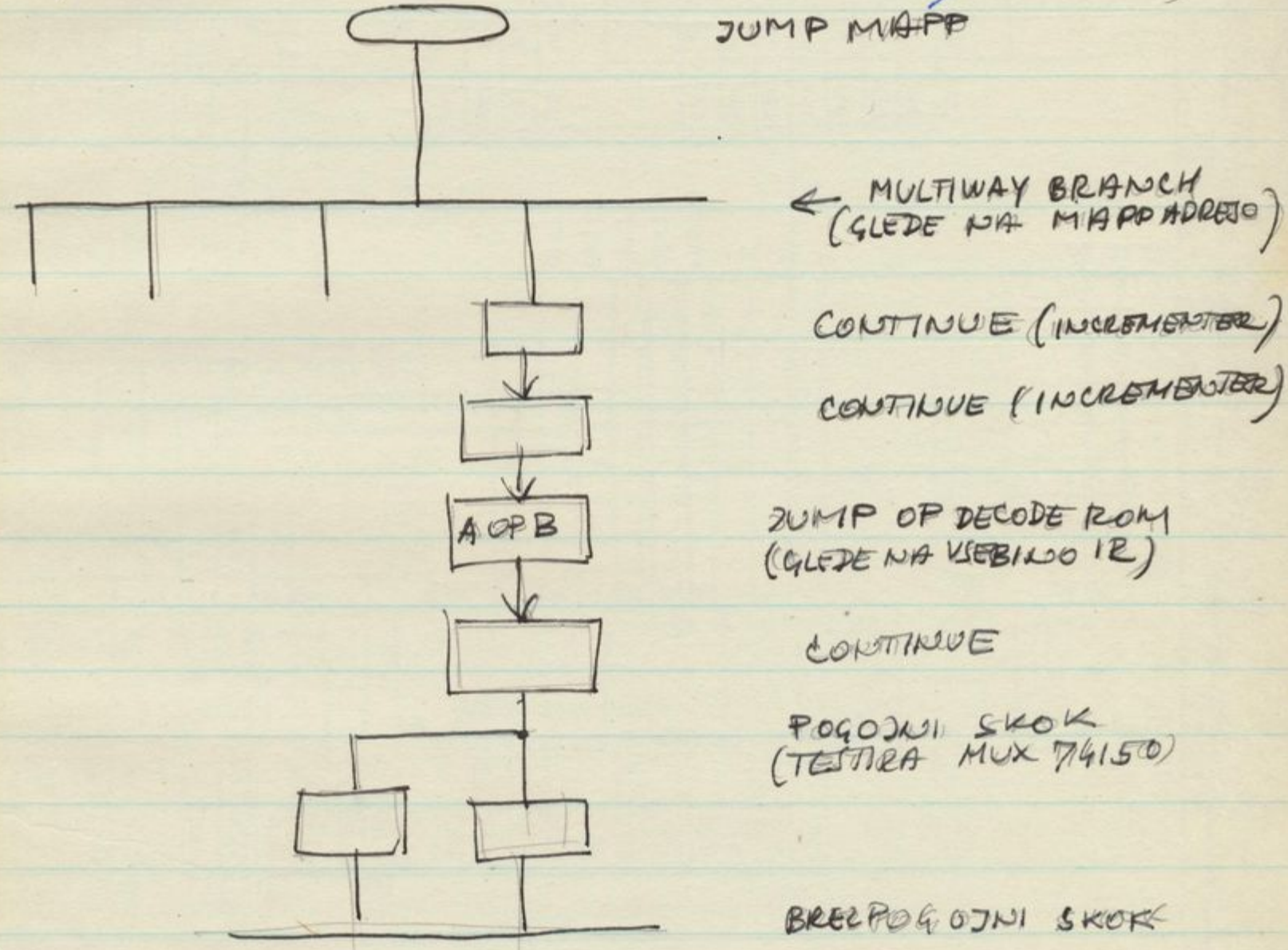
enable izhoda PLR registra PLR0 - PLR1 na BR bus. Mikroprogramni prom se odredno z odredno, ki se nahaja v PLR registru. Uporabljajo se brezprogramski izhodi na določeno lokacijo vseh PLR.

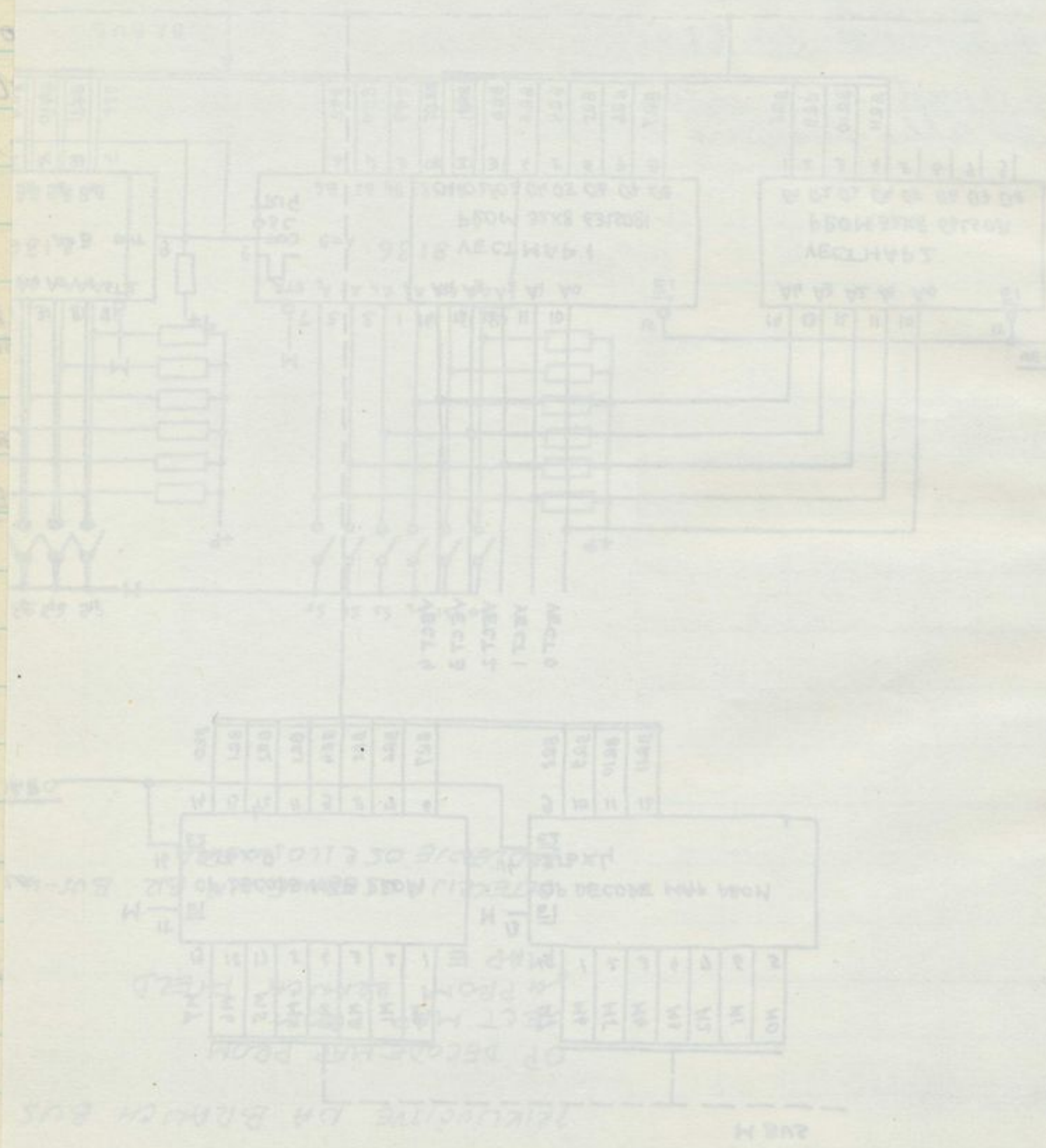


VECT MAPE

Enable VECT MAP PROM. Adresa rutine, ki se izvaja. Določeni <sup>ABORT</sup> flop se nahaja v vect MAP PROM-u. Kodor mikroprogram pride v stanje ko preglada določeno stanje vgraja npr BE (bus error...) in velika rutine so dnost prevase na adresne vhode su proma Abort se detektira preko vhodov v multiplexer 26150 in če je detektirano stanje, ki povzroča abort, se mesto

25811 postavi v stanje  $\overline{MAPF} = 1$  in  $\overline{PLE} = 1$ ,  $\bar{a}/\bar{c}$  PLR33 = 0 in na BR preverijo velikost VECT MAP PROM, hi po odzvojni obrot linije. Signalni vhod v 74150 multiplexer in ~~to~~ potem, po odzvojni obrot brise. (na vhodu 74150 moramo imeti ABORT FLIP FLOP.) (ČE SE POKAŽE DA NI POTREBE PO VEČ KOT 8 VERTONIH ~~VEKTORNIH~~ (ZA {ERROR, FAULT...}) in lahko <sup>VECT</sup> MAP PROM vseje izpusti in se vedno skoči brezpogojno na odzvo iz PLR konor lahko skokimo velitor odkue iz umiranj obrot.)





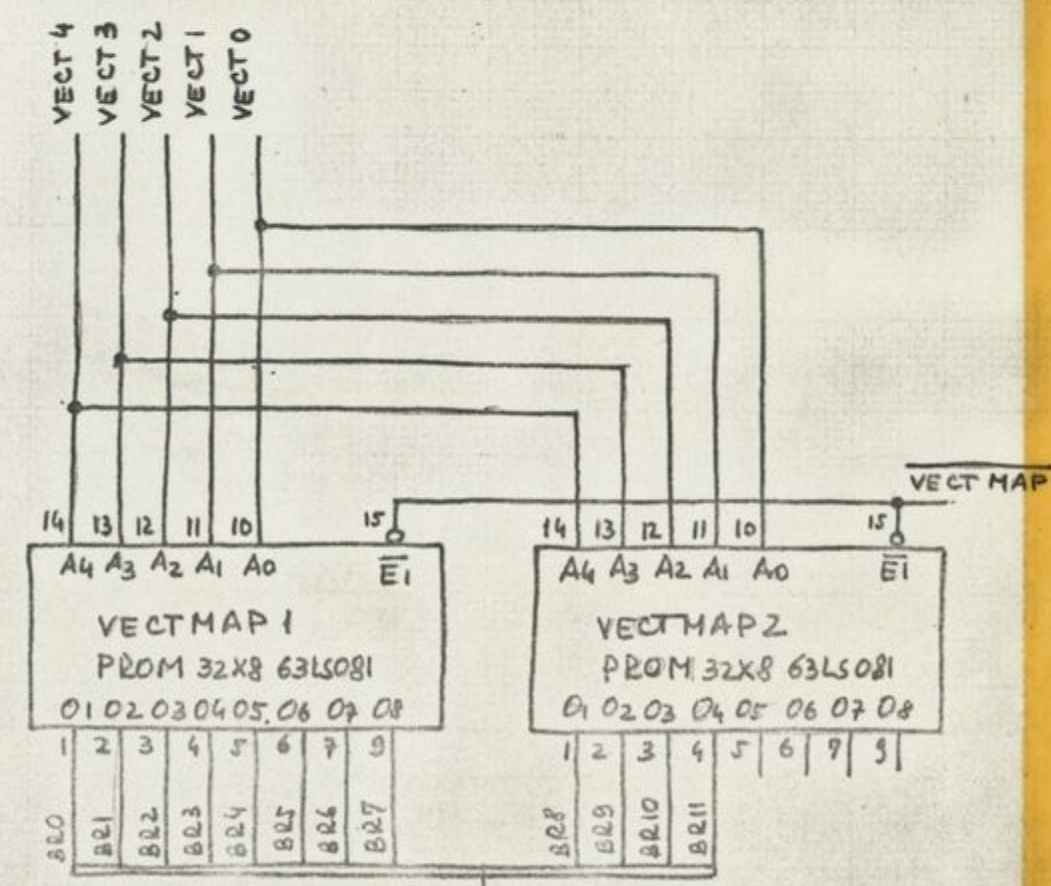
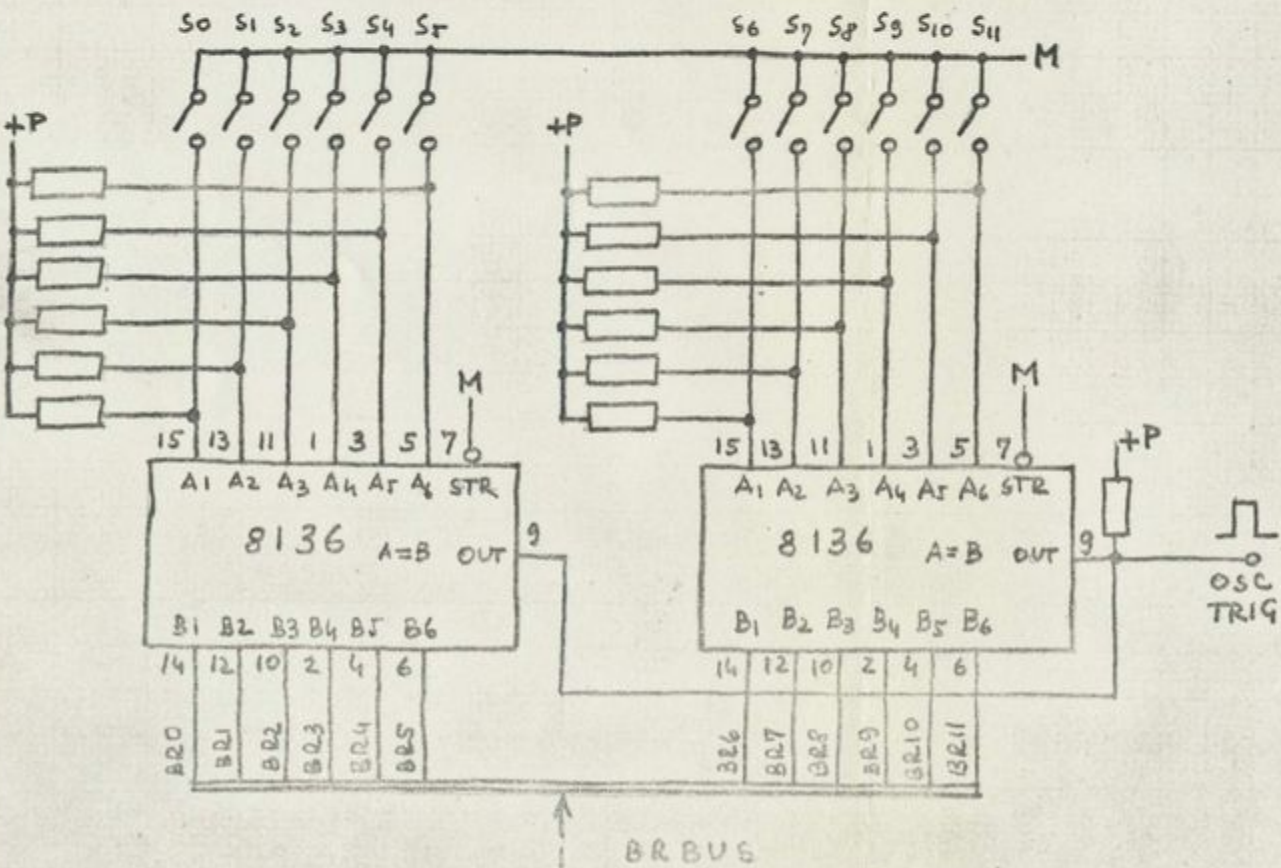
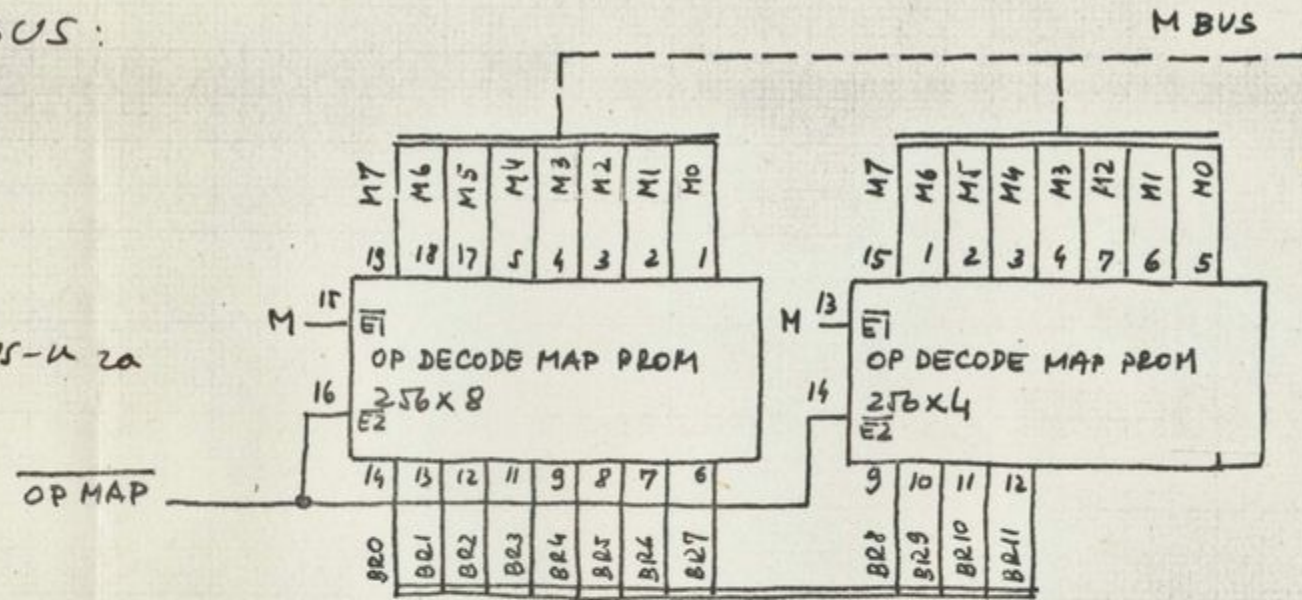
Handwritten notes on the left margin of the paper, including the letters "UN" and some illegible scribbles.

Faint handwritten text at the bottom of the page, possibly bleed-through from the reverse side or a separate note.

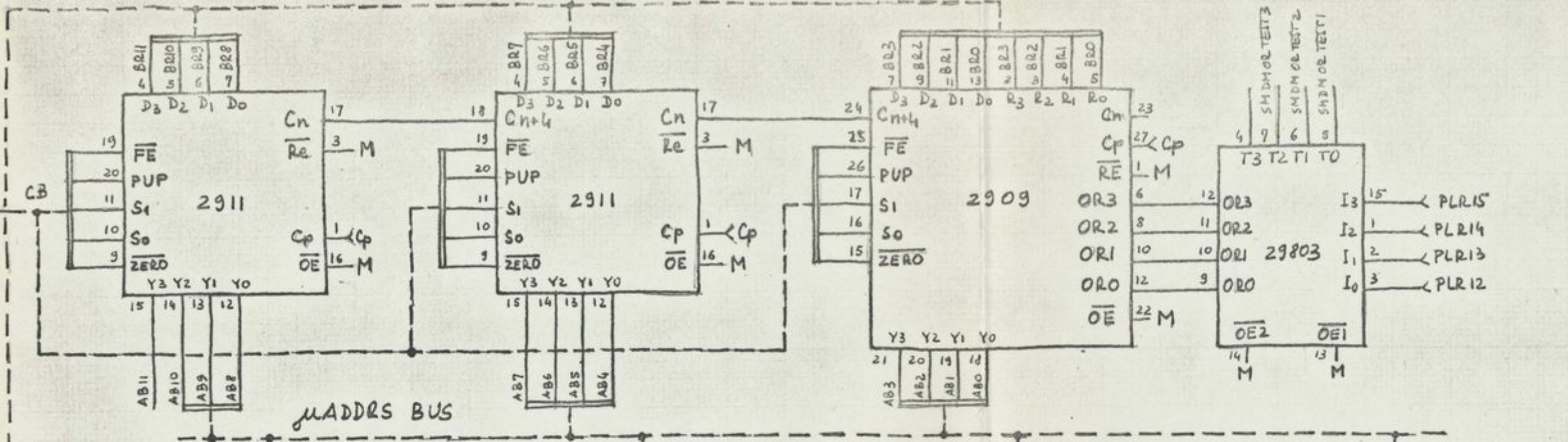
PRIKLJUČITVE NA BRANCH BUS:

OP DECODE MAP PROM  
 VECT MAP PROM  
 u PROM BRANCH FIELD  
 MAP E

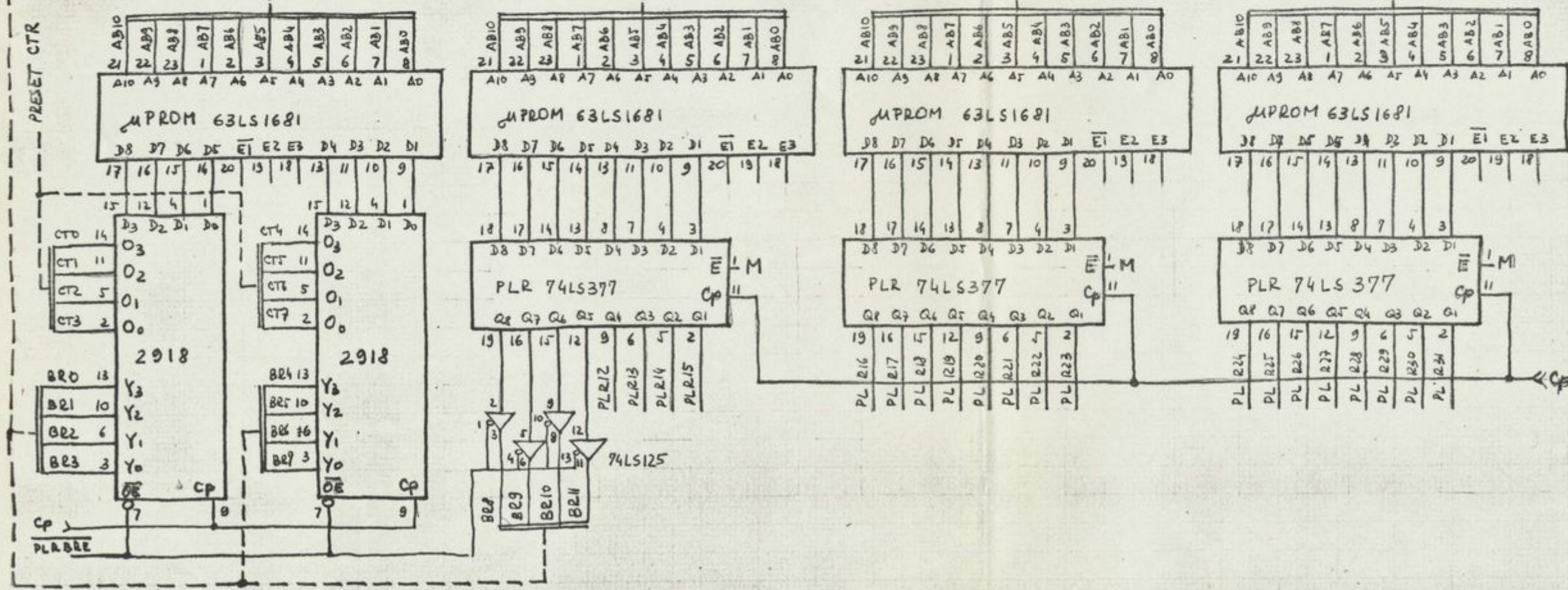
DETEKCIJA VSEBINE NA BR BUS-u za  
 PROJEKCIJE OSR ILOSKOPA



# BRANCH BUS

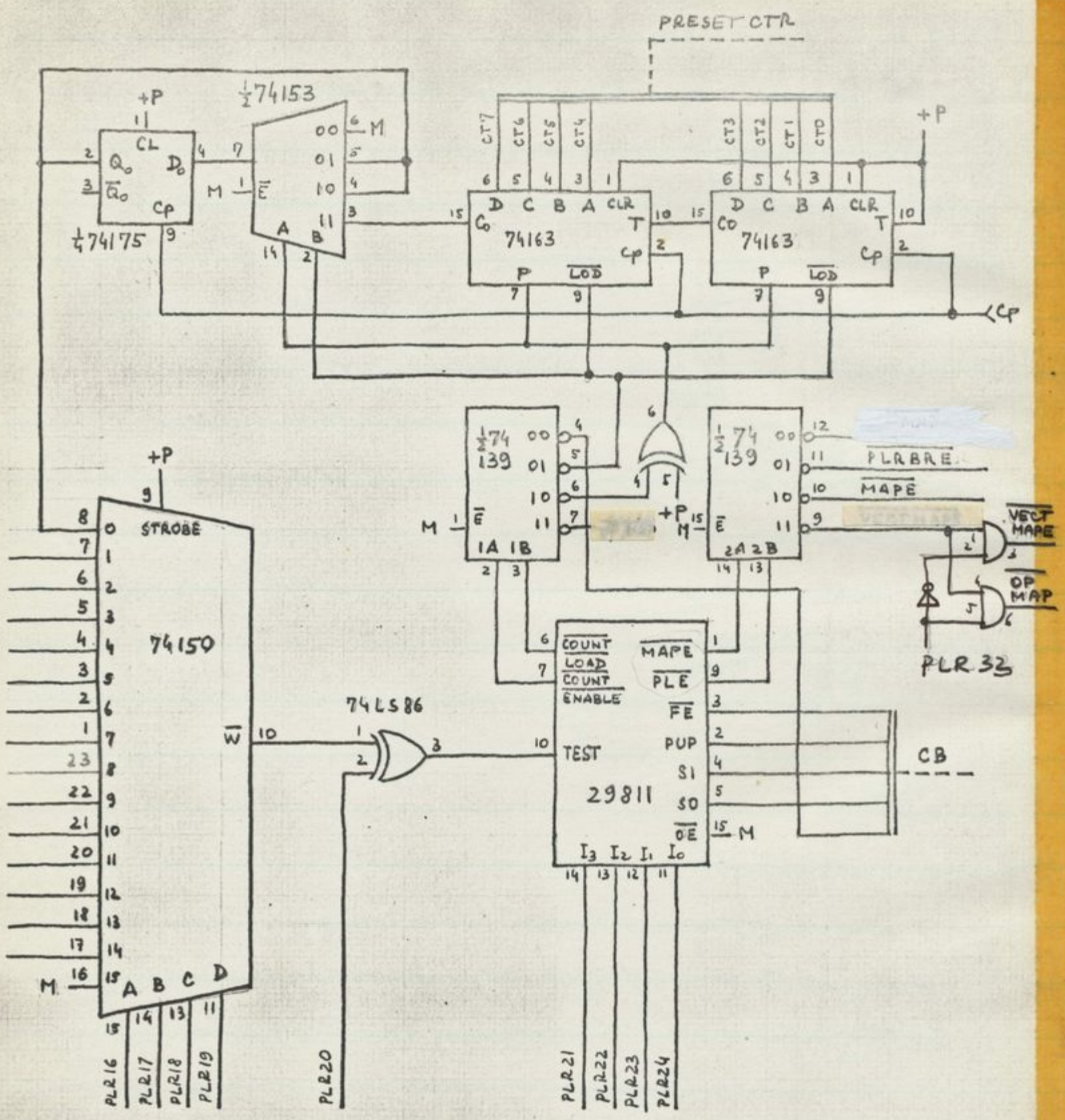


# ADDRESS BUS



PRESET CTR

PLABEE



AUGUST 81

1 - [faint text]  
 2 - [faint text]  
 3 - [faint text]  
 4 - [faint text]  
 5 - [faint text]  
 6 - [faint text]  
 7 - [faint text]  
 8 - [faint text]  
 9 - [faint text]  
 10 - [faint text]  
 11 - [faint text]  
 12 - [faint text]  
 13 - [faint text]  
 14 - [faint text]  
 15 - [faint text]  
 16 - [faint text]  
 17 - [faint text]  
 18 - [faint text]  
 19 - [faint text]  
 20 - [faint text]  
 21 - [faint text]  
 22 - [faint text]  
 23 - [faint text]  
 24 - [faint text]  
 25 - [faint text]  
 26 - [faint text]  
 27 - [faint text]  
 28 - [faint text]  
 29 - [faint text]  
 30 - [faint text]

512.82. Elektrostatika:

STIPENDIST ISKFE

FRANJOVIŠTRA POLZELE

Maceh, Zupancič, (tehnika), Vogel, Kocler, Virent, Anstole, Gustin  
delta: Kopyšeh, Trehor Dobnikar

1) Ureditelj prejšnje pogodbe:

- Delta je ploščo 10M od 30M protusloparovljenega površnika.
- Površnik naloga je v obliki elipsa izdelan in no <sup>izpogled</sup> delo za delta
- Problem zahtevna naloga (Omejenja do je vrednot opravljenih del -) (obim naloge zahtevno)

2) Poleg zgoraj omejenih bi na tem delovali se. Dobnikar, Gustin in delno Pipou.

3) Določitev koordinacijskega sistema o strani Delt  
Kopyšeh, Trehor

4) Elaboret za SBK v splošnem okviru naloge, izdelala bi koristila vredno za delo SBK ~~sko~~ misljeno je skupno naloga. ~~POE~~ Sodelovanje nalogi  
Matjaz, Kopyšeh, Baska, Kucyo, Trehor

Če se delovni red delovni ne da deliti ostane le 10M ~~na~~ vrednoto in izdelava.

5) Material do srede leta 20. FE. Dve prosti plošči za PE <sup>HEX</sup>

6) Material naj bo močta in kakovost od 44 (Hro)

7) Potrdilni listi na dec 10.

8) Upravitelj vzpostavi delo ljudski in dolga 4 leta (za omejen toč).



19.3.82 Elektrofolmfeto

20W

Močeh, Andrcle, Vogel, Virost, Kopravci, Trekor

- 1) do 20M iz prve pogodbe in nabavi material za uvedbo druge naloge.
- 2) Commercial instruction set  
K drugi oblogi pogodbe se doda CIS

15.4.82 Elektrofolmfeto

Močeh, Kodar, Andrcle, Virost, Dobnikor, Gastin Žemro, Kopravci, Trekor

- ① Npreravnje statistor (Andrcle)
- ② **CIS** — Določena vrstov inštrukcije iz CIS v μP instruction set  
(? Kolo je = super 16 ?)  
— Kvaarentor, ki mi konvencionalno lu tra  
(vse male v počke?)
- ③ software support
- ④ zadržano 1. elopu se konvencionalno nabor inštrukcij:
  - [1] — μP do eden int set
  - [2] — Copična shema CPU emote
  - [3] — Realizacija CPU 20
  - [4] — Testroupe
  - [5] — Dokumentacija

WCS

- 1) RAM je lahko hitrejši kot ROM (25ns RAM) (45ns ROM)
- 2) možnost diagnosticiranja. ~~to~~ ~~mo~~ možno je razviti teste ki hierarhično diagnosticirajo hardware.

31.5.82 Elektroplaketa

Virant, Doktor, Kodar, Vogel, Veselko, Maček, Dvornik  
Kogovnik, Matijevič, Treber

- 1) Clock za 3602 44 klocka funkcionalna, lahko je uvo povzročila zmanjšanje

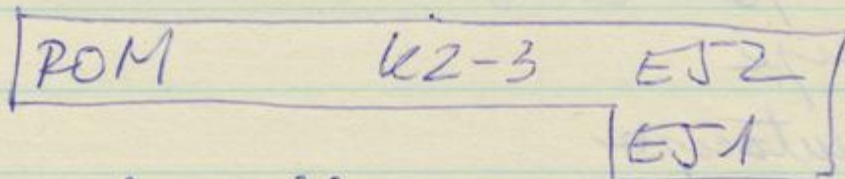
Elektroplaketa

23.06.82

FE Virant, Doktor, Maček, Vogel, Veselko, Dvornik  
Kogovnik, Matijevič, Treber

- 1) Diagnostika za CPU Način testiranja  
Mikrodiagnostika
- 2) CIS instalacijski set (obstoječa nejasnost glede poravnave  
instrukcij) (SHIFT) (STRING CHARACTER <sup>NUMBERS</sup>)
- 3) Signali K2-2 HALT REQUEST

Prečitaj



Prebrano kodno  
dne 9.7.82

- 4) External data path - zahtoj vlogo v SS max

E 53  
E 59

9. 7. 82

Elektrotehnika

Vivout Koder, Gustav, Koprarich, Mlotjiz, Trehor

① Maslinski sistem 6. AUGUSTA

17. 9. 82 Elektrotehnika

FE. Inderle, Maeh. Vogel, Dobulcor, Koder, Uedlko, Vivout

Δ Koprarich, Mlotjiz, Trehor

1) Zunayranje HW na 50%

2) μ Program je do delo dati na disk z METU 855.!

3) 275 μ m² z 90 bitov širine

$48 + 98 \times 1K$

4) 4K x 4 m² eom cca DM 100

$126 \times 2K$

275 4 1

1) μ Store ho WCS naj ho poseben modul znanj teh plošč programje. Osnovni instrukcijki set po svojo lasti - PROM-ih na procesorskih modalih z svojnovo poroelbuje paketyrko WCS.

1. 10. 82. Elektrotehnika

FE Vivout Dobulcor Duvrle Vogel Maeh Gustav

Δ Trehor, Mlotjiz

1) Dopovor o opravljenem delu na FE

ⓐ Strojje delno opravljeni μ Program z standardni set } METU 855 - RAKO  
" " " " " " " " " " " } z cis

ⓑ Blok sheme programje hi se uro v stoupi z uropouji

ⓒ Ponovljen cos 30 dnovh meseu

4 ESh

12.11.82 Eliklojehullito

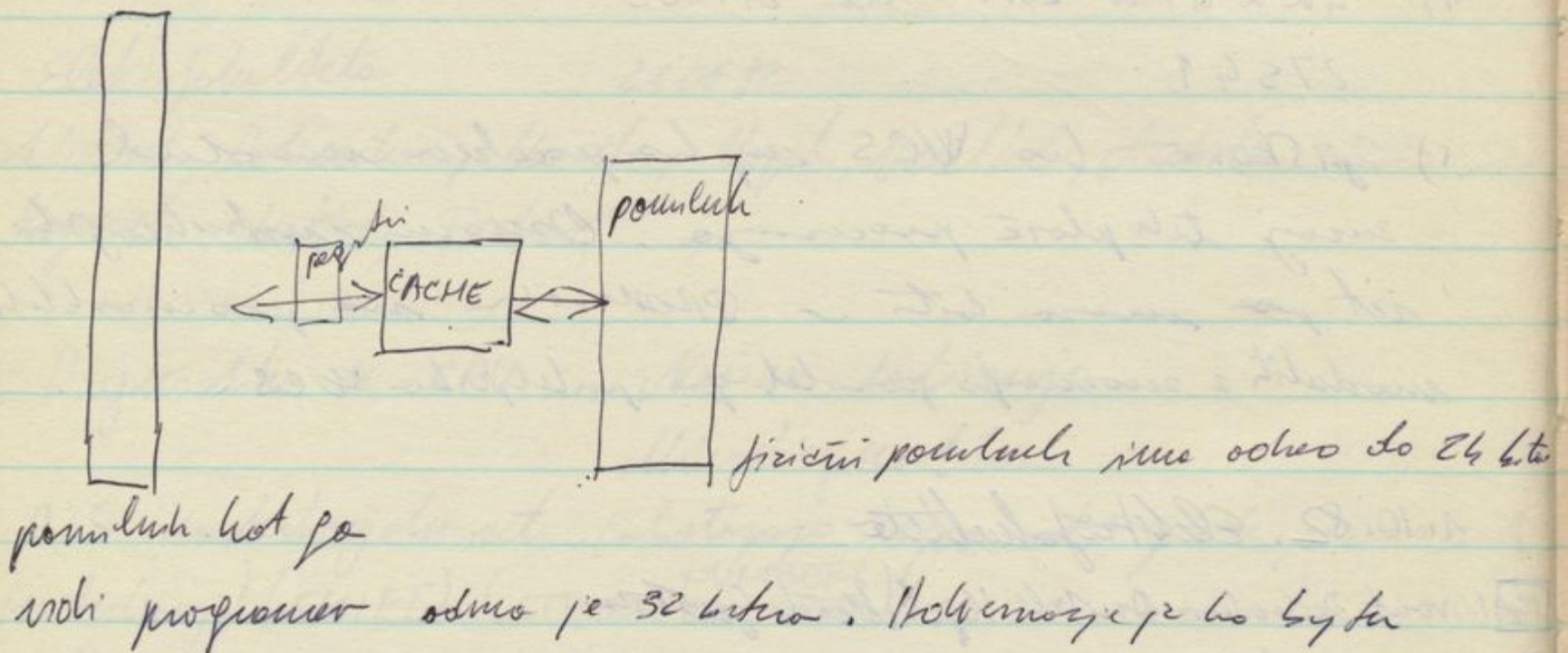
FE Delunhor, Andrič, Kodar, Vogel, Verulho  
Δ Trehur, Koprivsek

# ~~W~~ ~~X~~ VAX PREDAVANJA (VTS MARIBOR)

adresi  $0 \sim 2^{32} - 1$

Instukcij 20 račne jezike  
Virtualno pomulsko adresiranje  
Multiproc. org

## virtuelno adresiranje



prejeto viti.  
v fizični adres

TRANSLACIJSKI  
BUFFER

prvoje vrhoke  
adresi - fizične

podotel

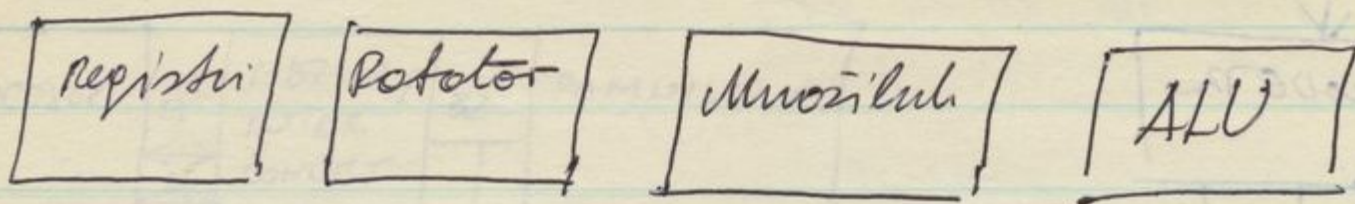
CACHE

instukcija

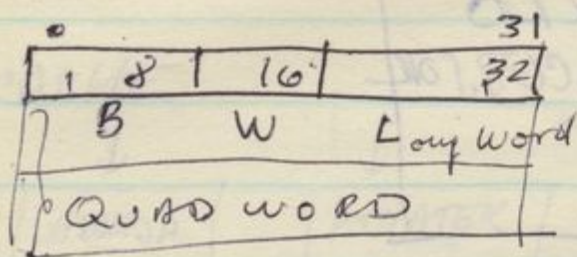
EXECUTION BUFFER

Programski niz

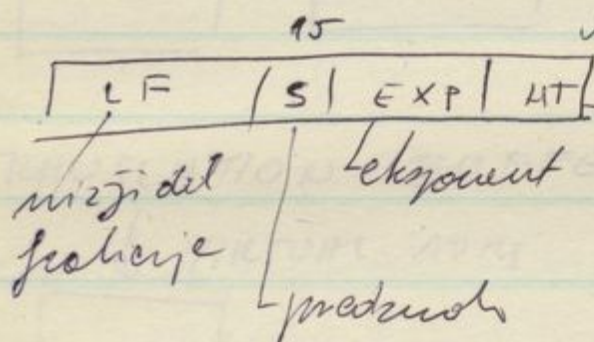
Osnovni moduli mikroprocesorja (data path)



Va to je povezano preko brehi vodil.



Operacije so na 32 bitih  
 lahko po dolo dirženi na 8 do 16 bitih.  
 Prenos med komputacijami in presojenji -  
 vedno 32 bitov



niži del funkcije

niži del funkcije

eksponent  
 predznak

0 - +  
 1 - -

normalizirani je v dvojiškem komplementu

0 1 1 ... 1 +127  
 ...  
 0 0 0  
 1 1 1 1 -1  
 ...  
 1 0 0 0 -128

Oblika mantice

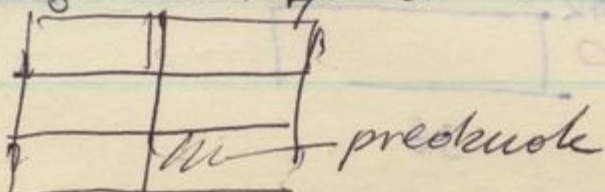
0.5 < 1

0.1

0.1 1 ... 1 1 1

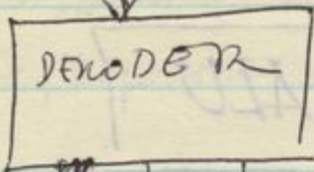
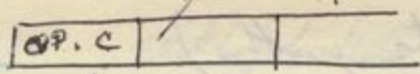
Double precision zovemo 2 words LW

Polovica decimalni niz



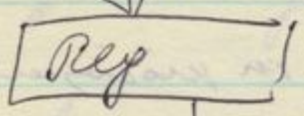
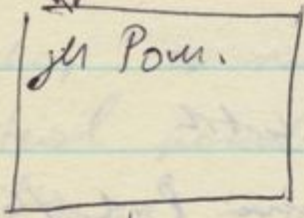
specifični operator

XB



Spec. op.

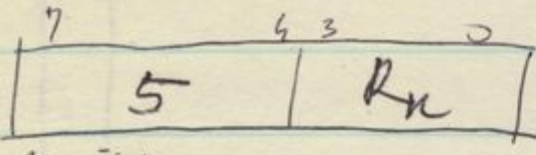
Startna adresa  
JIP or JI Pom.



ADRESE  
ZV.  
INTERNE  
REGISTRE



krmljenje



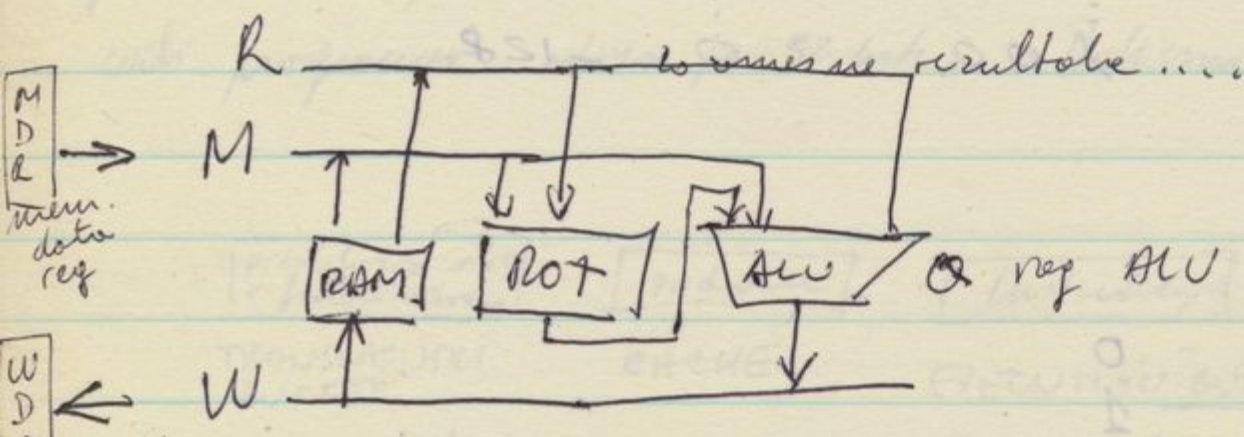
Adresna  
odrežitev

ADD B

dvoooperandna instrukcija

$$A + B \rightarrow A$$

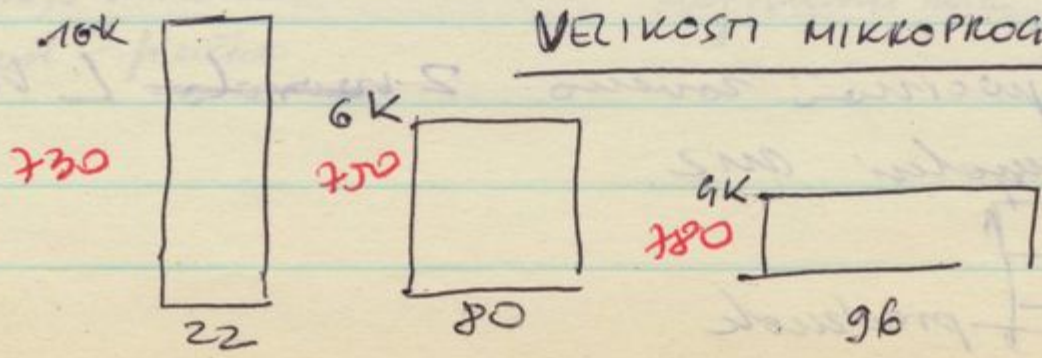
Voolila



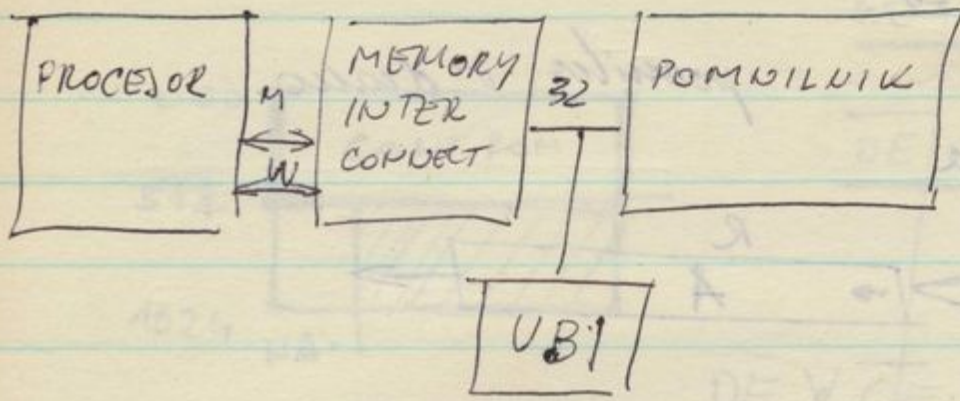
$$R + Q \rightarrow W$$

VELIKOSTI MIKROPROGRAMSKEGA POM.

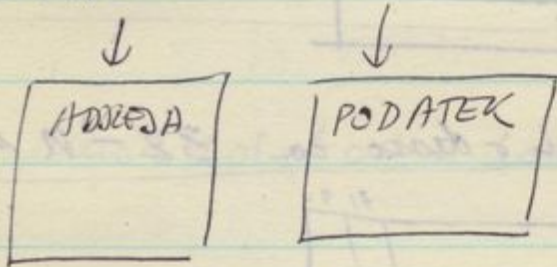
za VAX  
MINIPAC.



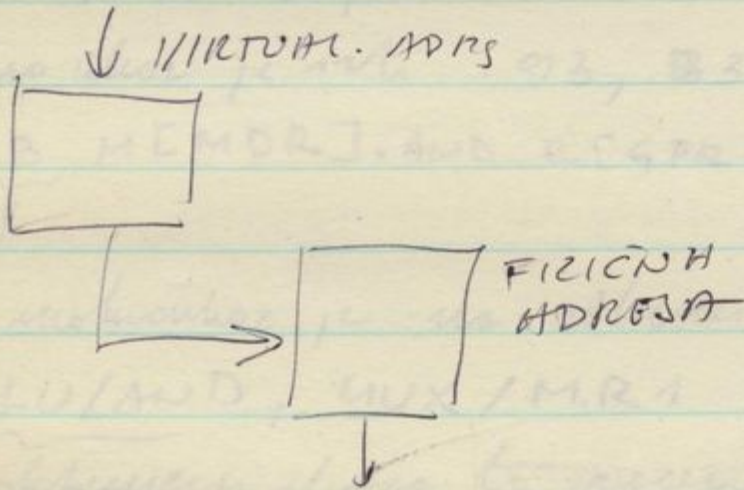
# SPOMINSKI UMESNIK



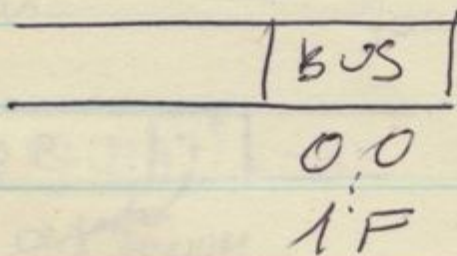
## CACHE



## TRANSLATION BUFFER



Mikroinštrukcia musí pojsť BUS so kľučkou  
spomínajúceho umesníka



10 READ

18 WRITE

# Dvomi vojjski rotator

0, 4, 8, 12, 16, 20, 24, 28

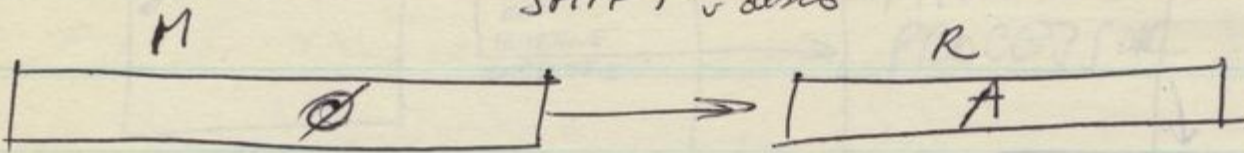
0, 1, 2, 3

promika v desno

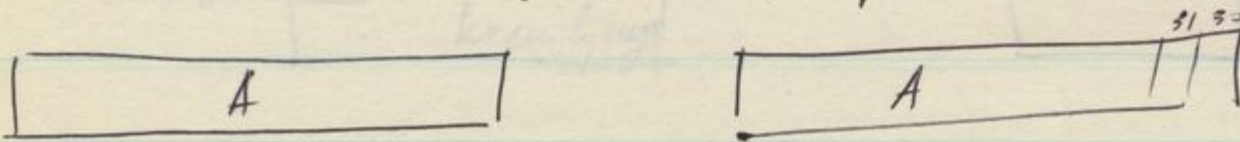
Rotacija



SHIFT v desno



$N$  SHIFT v levo = promika v desno za  $32 - N$  mest



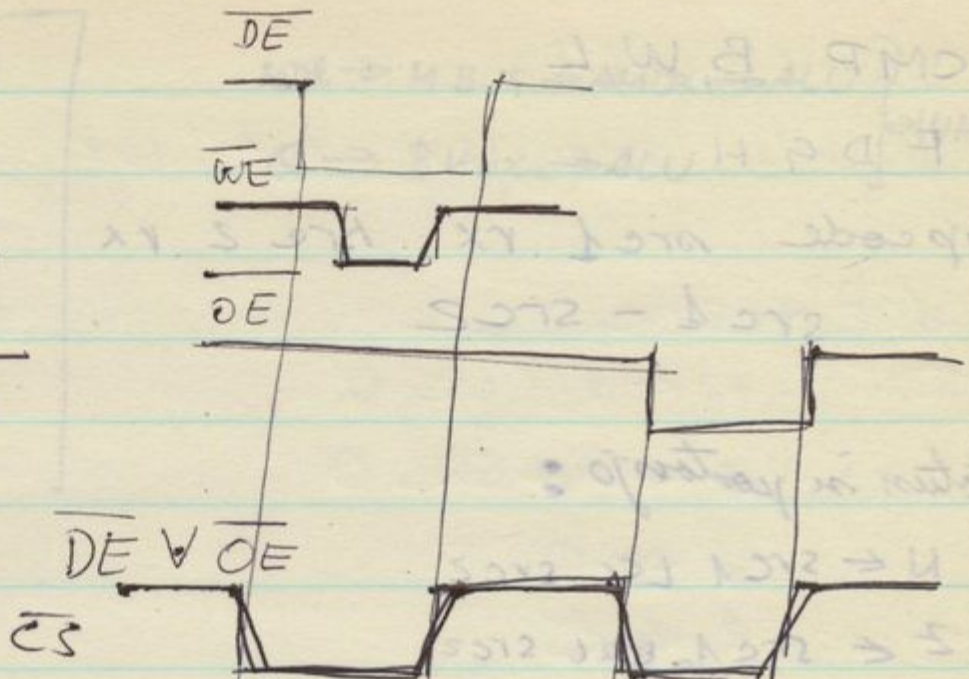
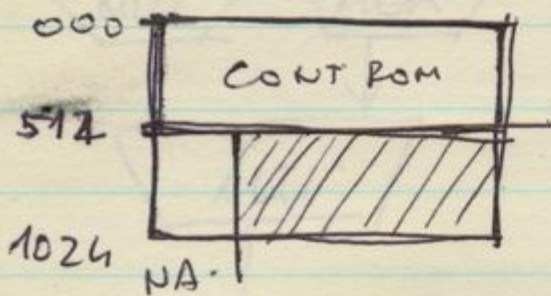
FE 24. 11. 82

KOGOUJEK, TROBIDR, MITJAS

VIRANT, DOBNIKAR, KODER, VOGL, PIREK,

Problematika FP procesorja





W:  $\overline{w}$  ; R:  $\overline{r}$  ; R:  $\textcircled{r}$

Vrsta Adresa doloz odvoz doloz  $\overline{cs}$  adrs 4 do 1

IL.BIT.B.W.L.REG

Mohno uloz je izti . 93, **B3**, D3

WB MEMDR].AND R[GP.R.R], ccop2

ke mohuonhoz je no sluvence 173

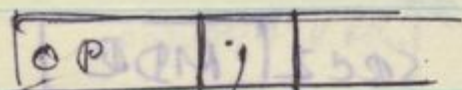
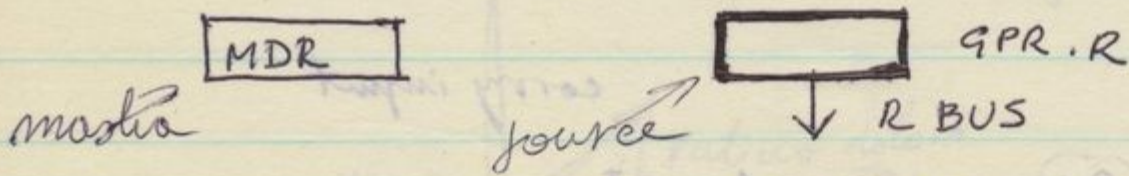
ALU/AND, MUX/M.R.1

ko definiciji slozjo to poevni definicija ALP  $\overline{c}$  jco (12k/107)

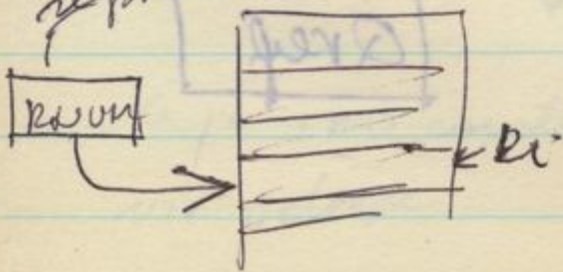
A = AND.B

A ← MBUS

B ← RBUS



odvrsu registora za prevrsu v RNUM

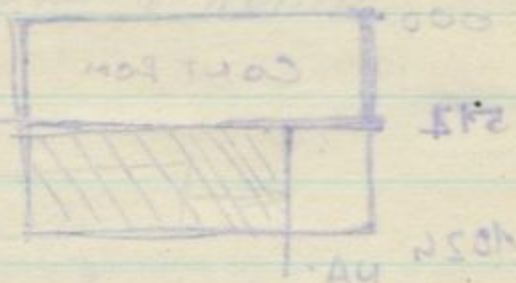


Sredaj imovno R, MDR relacij

CMP B, W, L

F D G H

opcode src1, rx, src2, rx  
src1 - src2



statun in postotnjo :

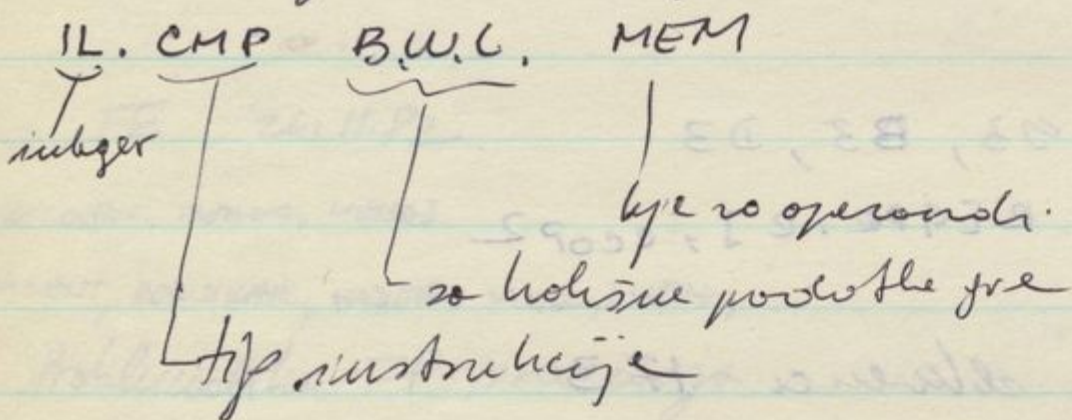
$N \leftarrow \text{src1} \text{ LSS } \text{src2}$

$Z \leftarrow \text{src1} \text{ EQ } \text{src2}$

$C \leftarrow \text{src1} \text{ LSS } \text{src2}$

$V \leftarrow 0$

Če no operandi v spominu



WB\_Q - M[MDE], CCOP1, SIZE [IDEP],

Seg 175  
4829 vrstica

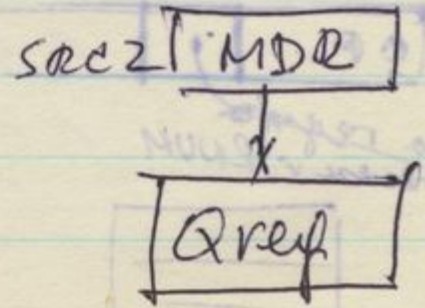
to je MUX/M, Q1, SRC(01), ALU/B-A-CI

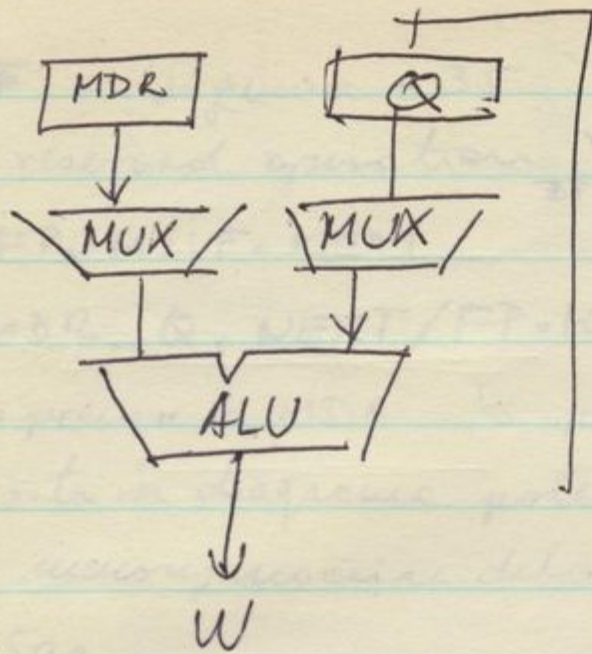
počlona no branje  
MUX v ALP ojač

to je MDE register

$A \leftarrow \text{MBUS}$   $B \leftarrow \text{Q register}$

carry input



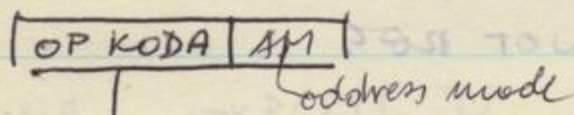


MDR → MBUS → AMUX → ALU }  
 Q → BMUX → ALU } WMUX

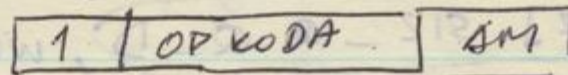
IRD 1 tip

IRDX tip

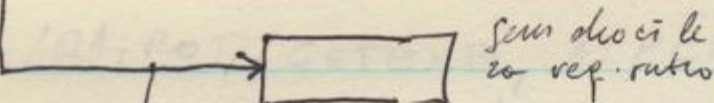
PRVI OP. SPEC.



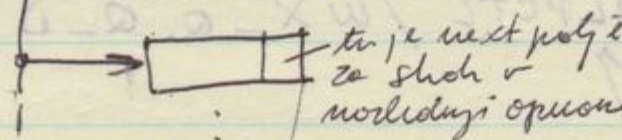
IRD CTR.



IRDX

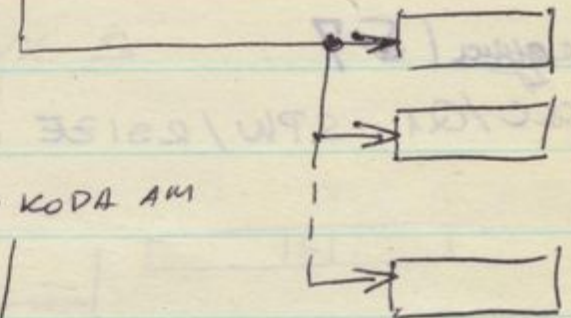


sema določi le za ref. rutino



to je next polje za slednji naslednji operand OP KODA AM

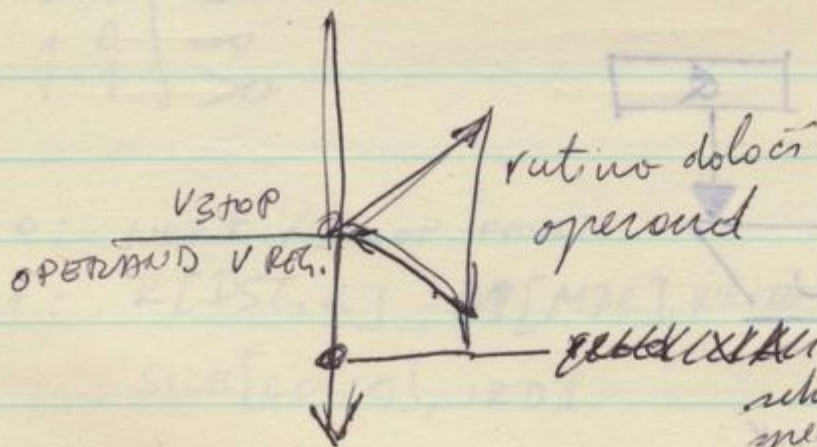
IRD CTR=0 IRD 1



OB VSTOPU IGNOIRAJ OP KODO

ti biti določajo rutino ali določajo operand

NEXT OP V REG



specifira IRDX specifikacija naslednjega naslednjega spec. operanda

Če je IRD counter > 2 potem je vedno naslednjemu vir odrese.

MOV (B,W,L)

Opcode, src, <sup>READ</sup>rx, dst, <sup>WRITE</sup>wx

dst ← src

Status: N ← dst < 0

Z ← dst EQL 0

V ← 0

C ← C

IL. MOV B, W, L. MEM

R [dst.r]. size - Q Q - D, WRITE NO REG  
size [IDEP], CCOP2, IRD1

sequencia 57

→ RSEC/Q1, SPW/Rsize, ALPCTL / WX - Q. Q - D

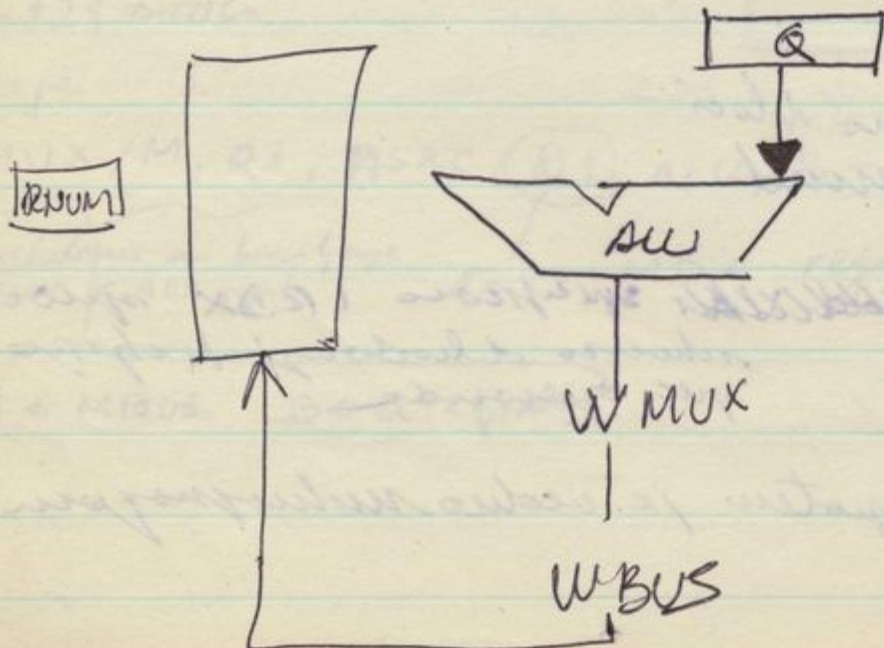
sequencia 96

definicija spec. funkcij  
ACP CPA

WRITE. NO REG

WRITE UNLESS REGISTER

MODE ASSERTED



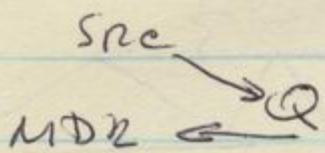
**MOVF** sequenca 335

EX: reserved operation

FP. MOVF. MEM

MDR\_Q, NEXT/FP.MOVF.REG,

Q rep. prenes v MDR to ponem SRC prenes v Q  
 Zdruzita u diogramu poteka za registrski nacini dela  
 in memory nacini dela



FP.MOVF.REG: WB\_EXP[MDR] *rotator napravi ekstrahirajo*  
*disponento*

WB\_EXP[M[]]

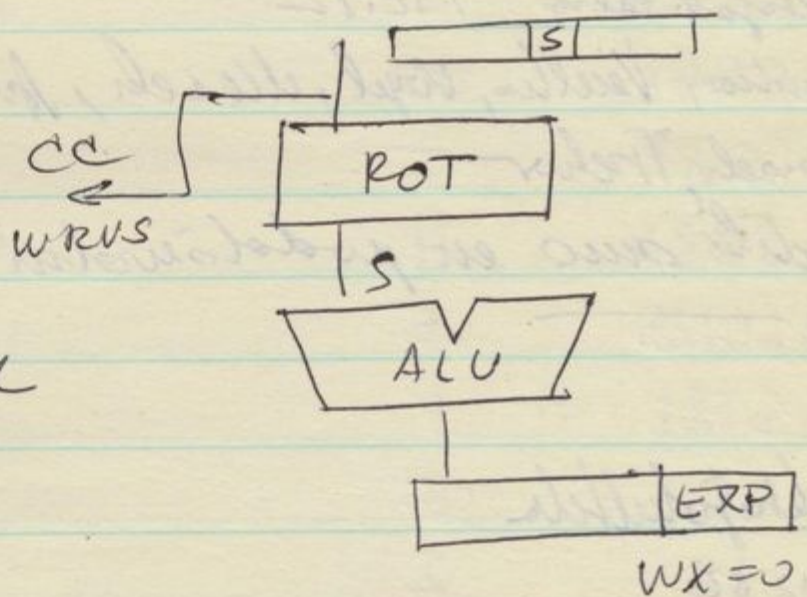
MSRC/Q1, ROT/GETEXP, ALPCTL/WX S

BUT

FLO FLT2

NOT M<15>

NOT W<31:0>

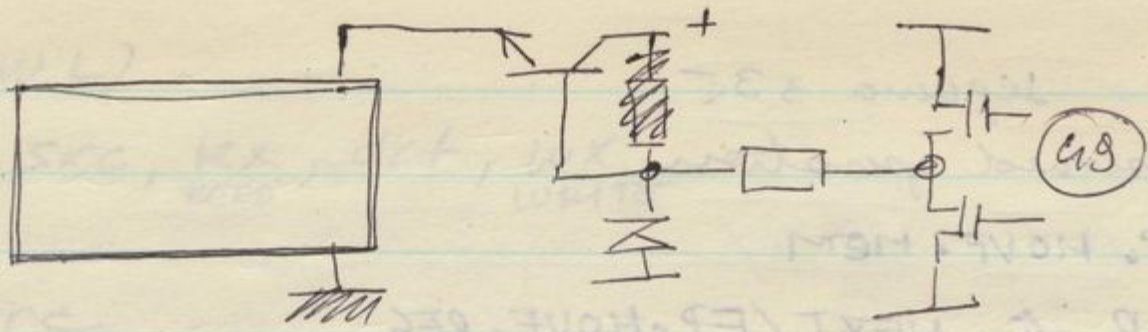


MW	
0 0	rezervirana operacija
0 1	<0
1 0	=0
1 1	>0

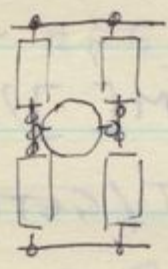
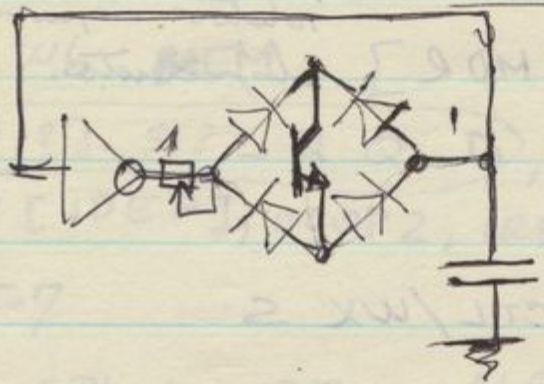
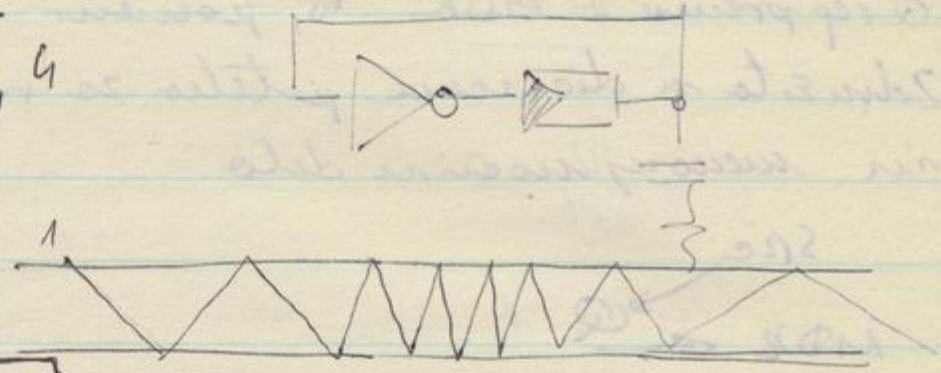
00: NEXT / IE.OP. FAULT

01: R[DST.R]\_M[MDR]. WRITE NOT REG

SIZE[LONG], IRD1



- 32
- 22
- 16
- 8
- 506
- 302
- 204
- 202
- 104



Elektronikseto 7.12.82

Dobrotvor, Vecellin, Vogel, Mosech, Anshin  
Kopovskii, Trekhov

Porodit' nuo en podotivudni board W900  
PR0M E80 control board

Elektronikseto

25.12.82

Vivout, Dobrotvor, Koder, Vogel, Mosech, Anshin, Vezelkov  
Kopovskii, Mosh'ev, Trekhov

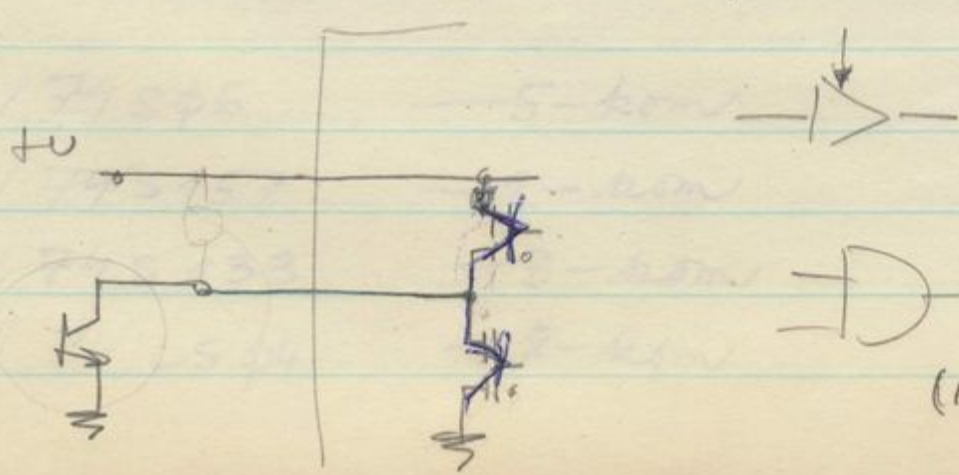
10x8

4801-5J

МОСТЕК

90

№	КОМАНДОВ	ОЗНАЧКА	№	КОМАНДОВ	ОЗНАЧКА	№
1	5	74LS127	2	5	74LS127	3
2	16	74LS04	3	16	8228C	4
3	10	74LS14	4	10	74LS14	5
4	2	74LS174	5	2	74LS174	6
5	2	74LS30	6	2	74LS30	7
6	2	74LS88	7	2	74LS88	8
7	10	8286A1	8	10	8286A1	9
8	2	74LS83	9	2	74LS83	10
9	4	74LS88	10	4	74LS88	11
10	2	74LS153	11	2	74LS153	12
11	1	74LS181	12	1	74LS181	13
12	2	74LS02	13	2	74LS02	14
13	2	74LS135	14	2	74LS135	15
14	2	8640	15	2	8640	16
15	2	74LS37	16	2	74LS37	17
16	2	74LS27	17	2	74LS27	18
17	15	6341	18	15	6341	19



7 AVT 6 (12) 6  
 (12V) 1ml (Power)  
 7

18. JAN 1983

Predali na fakulteto

Proto Board HEX AUGAT 1 komad

36 pinna konektor EXTRA - CHANNEL 2 komada

Prevel ANĐELJE Anderle Janes

ZAP ST.	OZNAKA	KOMADOV	ZAP ST.	OZNAKA	KOMADOV
1	74S157	2	20	74LS373	3
2	74LS04	16	21	85586	10
3	74LS14	10			
4	74LS174	5			
5	74LS30	5			
6	74LS85	5			
7	<del>74</del> 8641	10			
8	74LS83	5			
9	<del>74</del> 8881	4			
10	74LS123	5			
11	74S181	1			
12	74S05	5			
13	74132	2			
14	8640	5			
15	7437	2			
16	74LS27	5			
17	6341	12			
18	74LS03	5			
19	74S133	3			



8.6.83 FE

Koder, Virout, Dobunhor, ~~Česka~~<sup>Ferkoj</sup>, Verelko  
Kogovsek, Trešor

Noroplazji glede izdelovanj med FE in Dalto  
Obruteli odpravno no dops FE (Aco)

15. april 1983

✓ 74 S φφ	— 15   kom
✓ 74 S φ2	— 5   kom
✓ 74 S φ8	— 1   kom
✓ 74 S 11	— 5   kom
✓ 74 S 2φ	— 1φ   kom
✓ 74 S 3φ	— 5   kom
✓ 74 S 138	— 5   kom
✓ 74 S 153	— 5   kom
✓ 74 S 175	— 5   kom
✓ 74 φ8	— 5   kom
✓ 74 LS φ2	— 1   kom
✓ PAL 16 H 2	— 5   kom

IC-ji za emulacijo PDP 11/34

✓ 74 S φ5	— 5 - kom
✓ 74 S 157	— 2 - kom
✓ 74 S 133	— 3 - kom
✓ 74 LS φ4	— 8 - kom

- ✓ 74 LS 14 — 2 — kom
- ✓ 74 LS 27 — 5 — kom
- ✓ 74 LS 3φ — 5 — kom
- ✓ 74 LS 83 — 5 — kom
- ✓ 74 LS 85 — 5 — kom
- ✓ 74 LS 123 — 5 — kom
- ✓ 74 LS 174 — 5 — kom
- ✓ 74 LS 373 — 3 — kom

~~63φ~~

- ✓ 8641 — 1φ — kom
- ✓ 8881 — 4 — kom
- ✓ 6341-1 — 12 — kom
- ✓ 85 S68 — 1φ — kom
- ✓ 864φ — 5 — kom
- ✓ 74 Sφ3 — 4 — kom
- ✓ 74 132 — 2 — kom

IC-je vzal : Janez Anderle za FE

cont

- 74 LS 244 — 5 — kom
- 74 157 — 15 — kom
- 74 Sφ8 — 3 — kom
- 74 S 1φ — 1φ — kom
- 74 LS 139 — 5 — kom
- 63φ6-1 — 2 — kom

physical memory. In an unmapped RMA 11M system, any location in physical memory can be accessed.

In an RSX-11M-PLUS system, an attempt to specify an illegal location or access beyond the limits of the partition causes an error. The user is limited to the specified memory region (TASK, REG, or PAR). However, if the command omits a keyword, it can access any address of physical memory. In a multiprocessing system, to access an address in a device page (the processor's I/O page), you must treat the location as absolute and append the /AFF keyword. Likewise, to access an address in a CPU partition, you must treat the location as absolute and append the /CPU keyword.

The line terminator directs the subsequent action of Open. Any value that is entered replaces the contents of the word whose address and current value are shown in the first part of the line-2 display. The slash (/) is part of the line-2 display.

Line Terminator Options: *OPE* *XXXXXX* *address register*

**ESC** - ESCape or ALTMODE: **ESC** (ALTMODE) terminates acceptance of further input and is the only means of exit from the MCR Open function.

**RET** - Carriage return: the next sequential location is opened.

4-166

**RET** - Up-arrow (circumflex on some terminals) carriage return: the previous location is opened.

\***RET** or @**RET** - Asterisk carriage return, or at-sign carriage return: the location pointed to by the contents of the opened location is opened.

>**RET** - Right angle bracket carriage return; the contents of the open location is treated as a branch instruction. The effective address of the branch is the next opened location. It is not permissible to supply a value (modify the current open location) and close the location with a >.

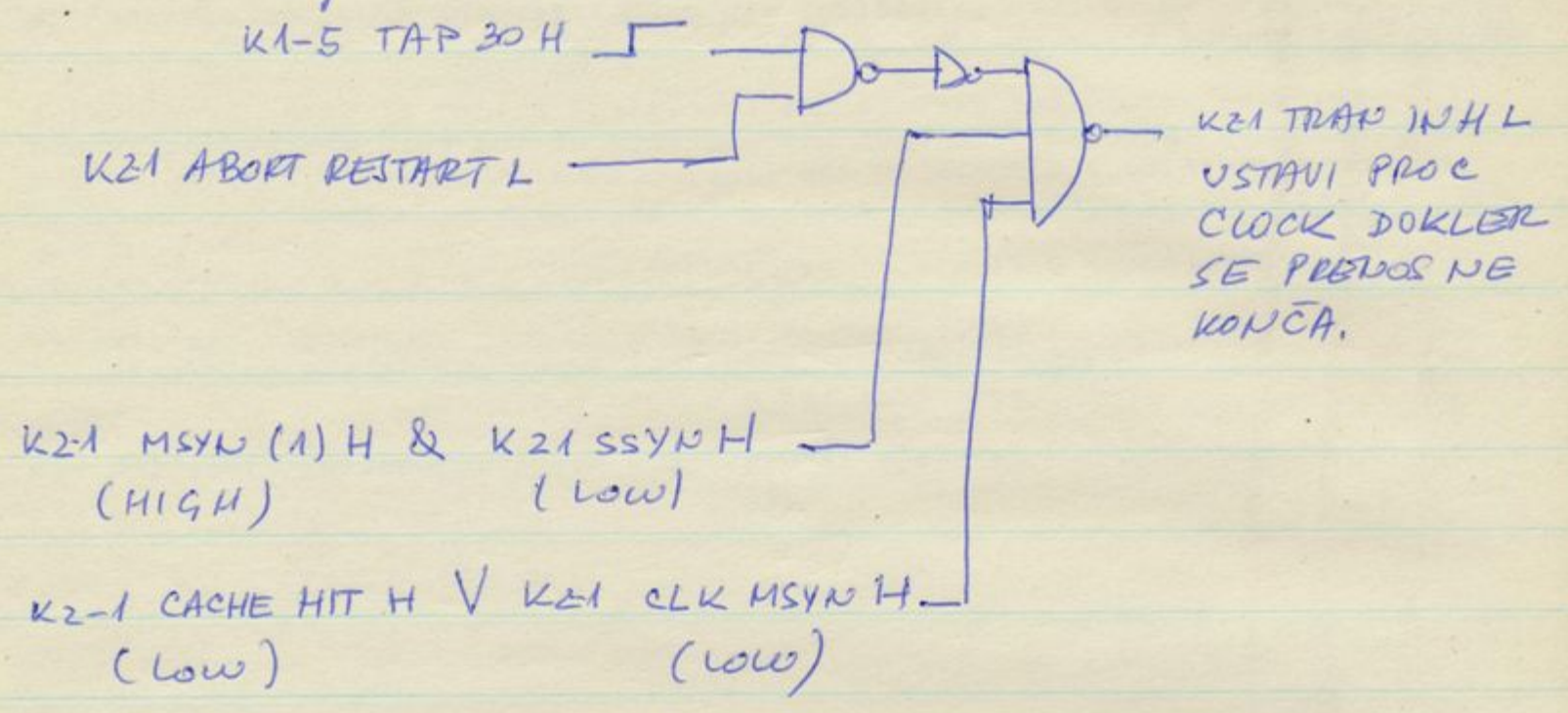
Example:

```
→OPE 4+10/TASK=CYLE
000014 060014/350ESC
```

In this example, the specified virtual address in the task CYLE is 4, while n equals 10, yielding the effective task-relative address of 000014. The system responds by printing the virtual address (000014) and the contents of the address (060014).

# DATA TRANSFER VEZJE

K2-8 BUF DAT TRAN (1) H. Vsi memori podsklopi se začnejo s tem signalom.



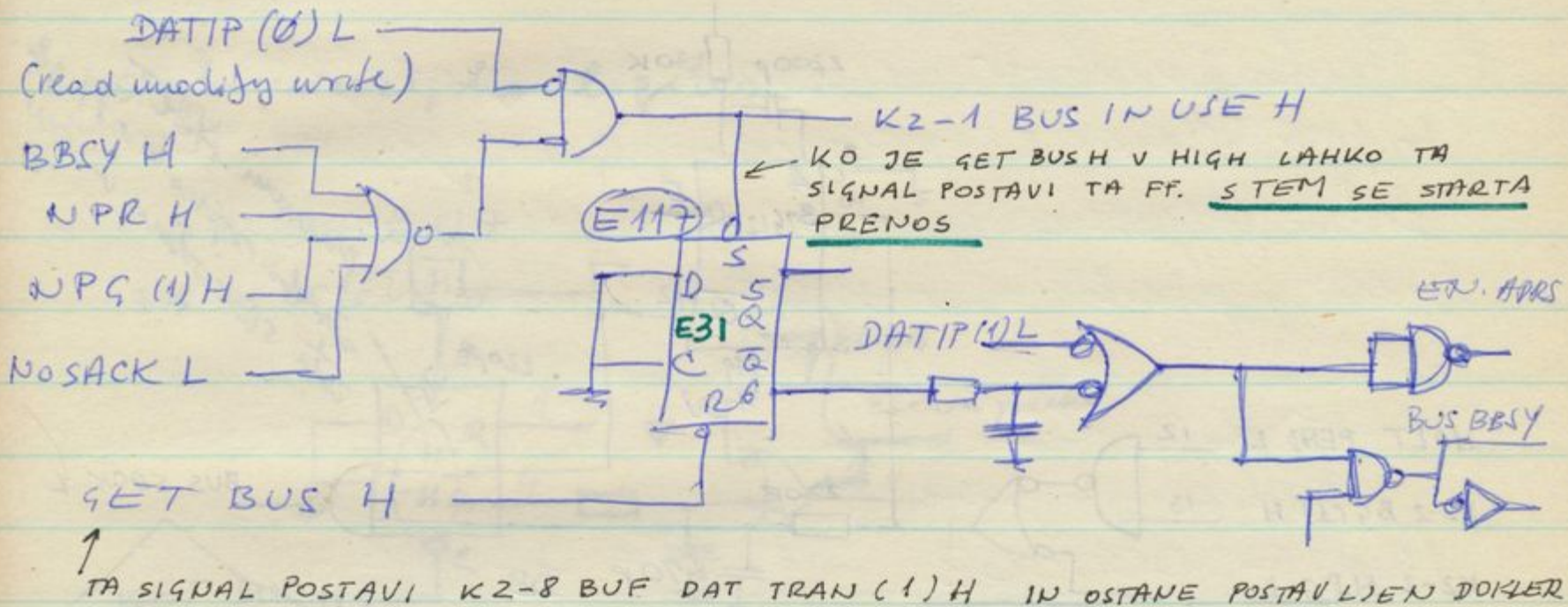
Če se generira CACHE HIT L pomeni, da ima coche podsklopi 10 BUS ADDRESS linij že na kometor J1.

Ko je med DAT1 ciklom postavljen signal K2-1 START TRAN L coche uporablja BA linije, da ugotovi, če ima podsklopi. Če so podsklopi v coche pomembni in postavijo signala CACHE HIT L in TRI STATE AMUX L. Ti signali abortirajo prenos ~~po~~ po UNIBUSU in podsklopi se prenosujejo preko EXT DATA PATH.

CACHE HIT L generira signal K2-1 CACHE HIT to signal potem ponovno sproži procesorski clock, ko je K2-1 CLK MSYN H v high (to je 150 ns po signalu K2-1 START TRAN L)

## SINHRONIZACIJA NA UNIBUS

Logično vezje orbitra ali je procesor ali kateri druge naprave obkrožene na unibusu.



TA SIGNAL POSTAVI K2-8 BUF DAT TRAN (1) H IN OSTANE POSTAVLJEN DO KLER  
 K2-1 GET BUS H NE GRE V LOW S POSTAVITVIJO TAP 30 H

## BUS CONTROL

Ko je (E117) celica retirana v pričnem prenos s signalom K2-1 ENAB ADDRESS L kot sproži naslednjo sekvenco

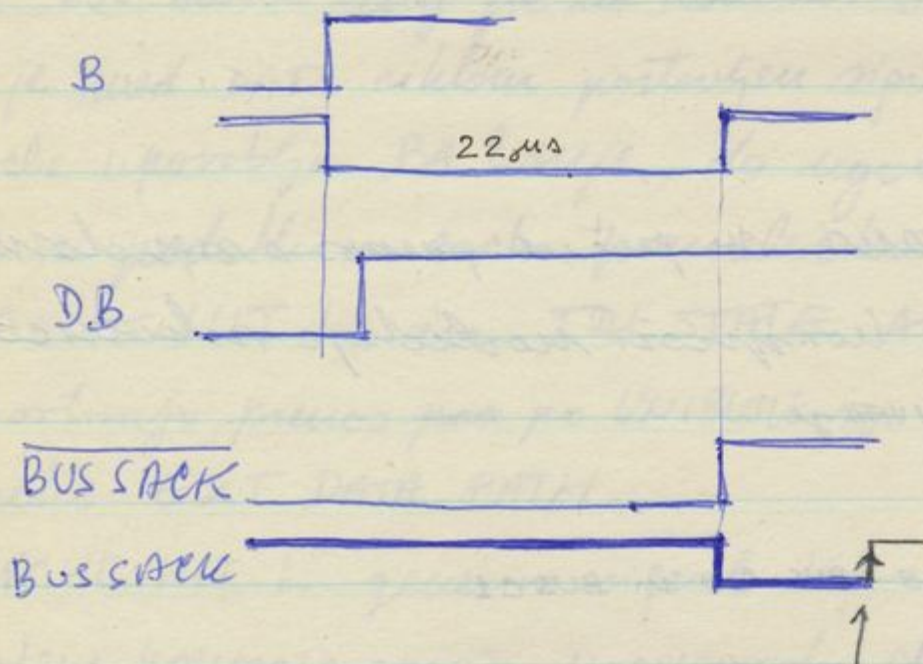
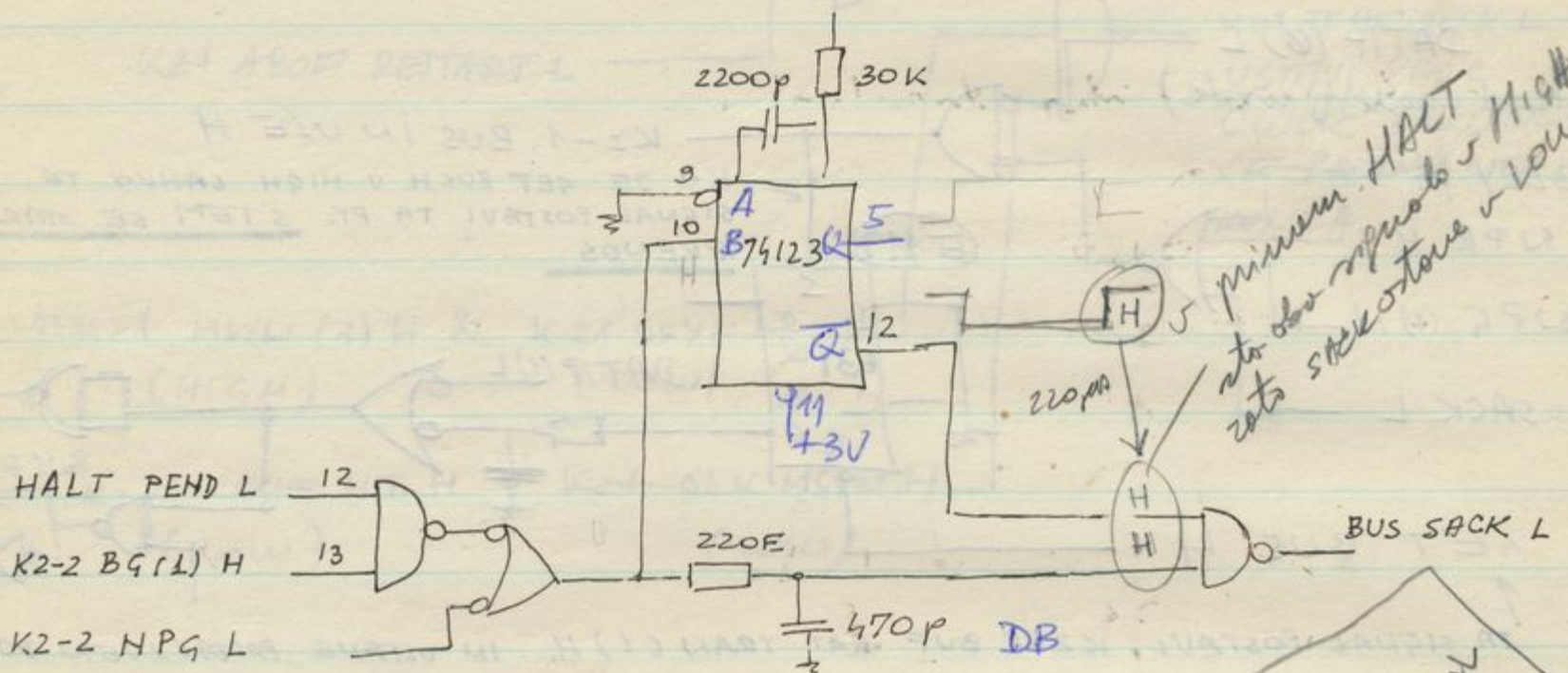
- 1) ENABLIRA BUS ADDRESS DRIVER-je
- 2) — " — BBSY driver
- 3) ENABLIRA KONTROLNE SIGNALE BUS C0 in BUS C1

CS	CO	Op
0	0	DATI read
0	1	DATIP read modify write
1	0	DATO write
1	1	DATOB

4) ENABLIRA PODATKOVNE DRIVER-je ČE GRE ZA DATO OPERACIJO

# TIMEOUT NO SACK I VEJE

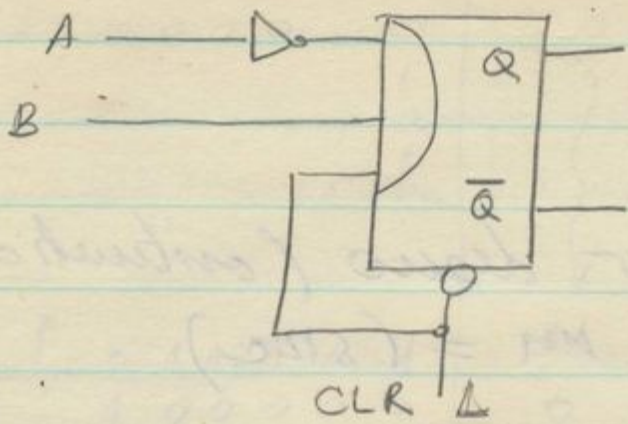
Veje portari SACK L NO UNIBUS se naprava, ki je zahtevala unibus na portari SACK v 22µs potem, ko je bila esoblivous grant linijs.



~~PO SHEMI LA ID I EA  
NE VIDIM ~~SACK~~ VAKO  
JE TO MOZNO  
(HALT REQUEST IE EXT SIGNAL  
PGM MPG PA SE BALSAS  
SACK SIGNALOM)~~

SACK PRIDE  
IZ KONTROLE

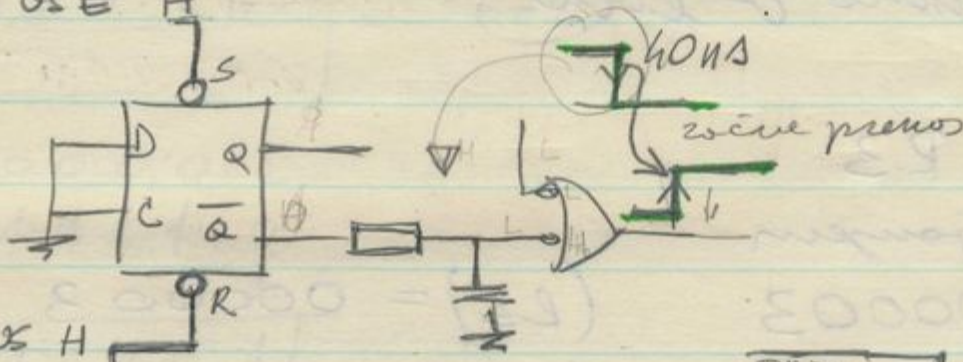
ko procesor sprejme SACK umakne grant linijs



simulacionijska logika:

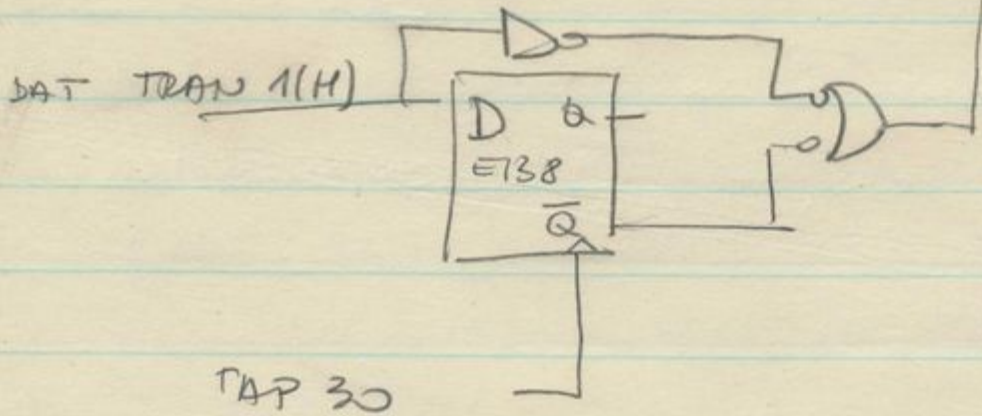
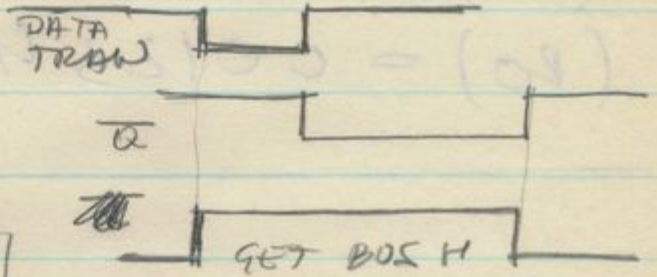
E117

BUS IN USE H



GET BUS H

omogućiti, da se preuzme  
BUS, kada je BUS IN USE H  
preklopu L



30.06

ASH

072RSS

2 u pomembne un svet v desno (aritmetični  
pomiki) di levo, kjer je un = (snc)

Število pomikov u ravne kot spodnjih 6  
bitov source operando, v razpon od -32 do  
+31. Negativno u vedno pomika pomika  
desno, pozitivno v levo

ASH R0, R3

pred izvajanjem

(R3) = 000003

(R3) = 000003

(R0) = 001234

(R0) = 012340

---



UAPA 0000000 0  
 ① PT 11 0  
 ② XBS 11 0  
 ③ E6/1 0  
 ④ 0 1 1 1 1 1 1 3L  
 ⑤ 1 0 0 0 0 0 0 0  
 ⑥ -1 1 1 1 1 1 1 -3<sup>-</sup> (-1)

H = 000001  
 000000  
 1  
 111111

- ③ E141/6 (4700) NI VEZAN NA GND
- ④ E24/5 (6300) NI VEZAN NA GND
- ⑤ E72/2 NI VEZAN NA GND
- ⑥ R255, R209, R201, R226, R227
- ⑦ E192/7 (R215) NI VEZAN NA GND
- ⑧ E199/1 (R211 200101) POVEZANA NA LAGE 5 SE STIKA NA GND
- ⑨ E159/4, 5, 6 (7402) INPUT 7E6 OUTPUT 7E4
- ⑩ E145/8 (7473) NI VEZAN NA Vcc
- ⑪ E143/16 (6331) NI SPOLJEN NA Vcc
- ⑫ E142/17 (6333) NI VEZAN NA Vcc

⑬ R 9

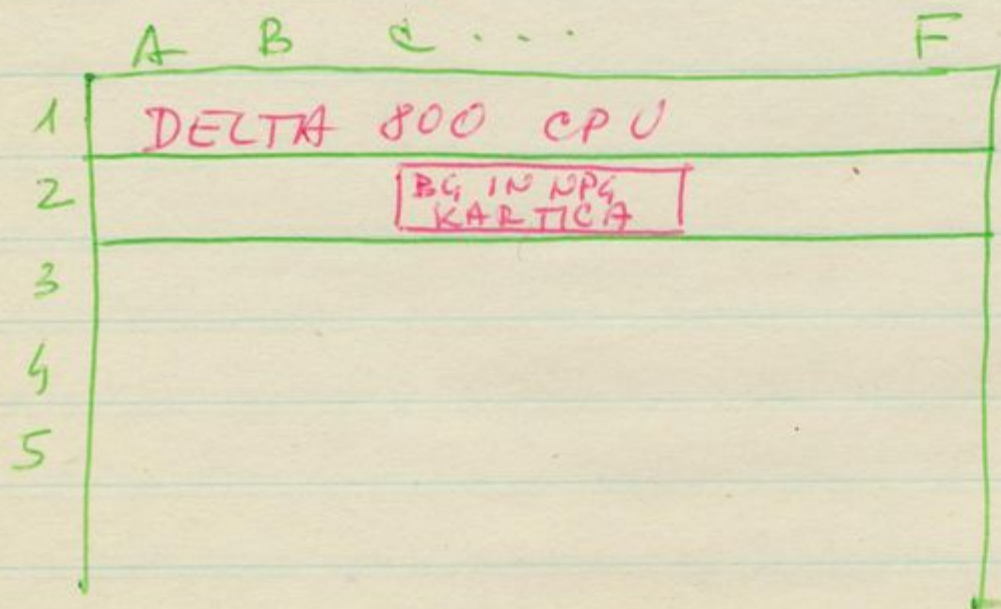
TRAP VECTORS 1

D15	14	13	12	11	10	9	8	7	6	5	K2-3			AMUX
0	0	0	0	0	0	0	0	KTE 1H	PE 1H	KTE 1H	C4	C3	C2	0 0
											0	0	1	0 0
											0	1	0 0 0	TIMEOUT, ODD ADDR STACK, OFL ERR
											0	1	1 0 0	ILLEGAL INSTR. & RESERVED -1-
											0	1	1 0 0	BPT (TBIT TRAP)
											1	0 0 0 0		I/O TRAP
											1	0 1 0 0		POWER FAIL
											1	1 0 0 0		EMT
											1	1 1 0 0		TRAP INSTR
										1	0 0 1 1 0 0			MEM PE
										1	0 1 0 1 0 0 0			MMU ERROR

22.11.85


Testiranje D800 CPU pod operacijskim sistemom  
net PDP 11-34

BACKPLANE DD.11-PK



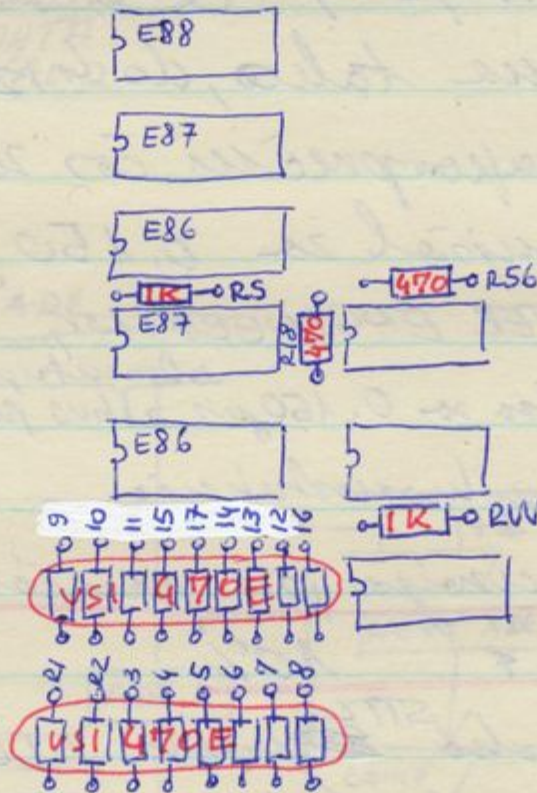
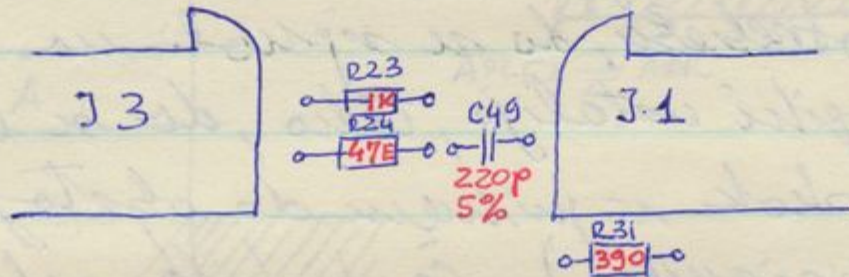
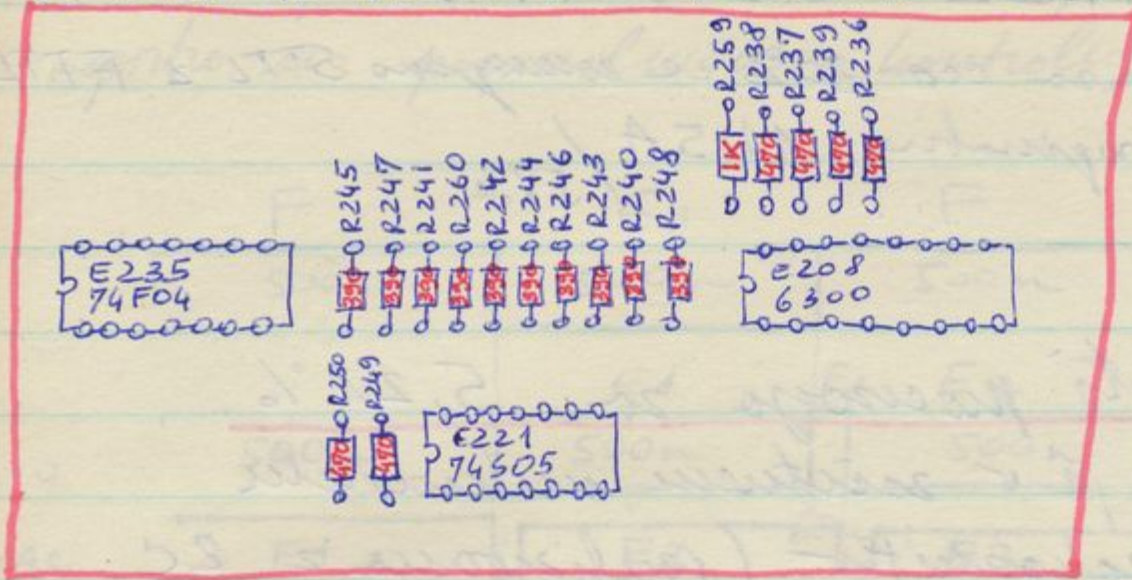
Operacijski sistem deluje OK.

## NAPAKE

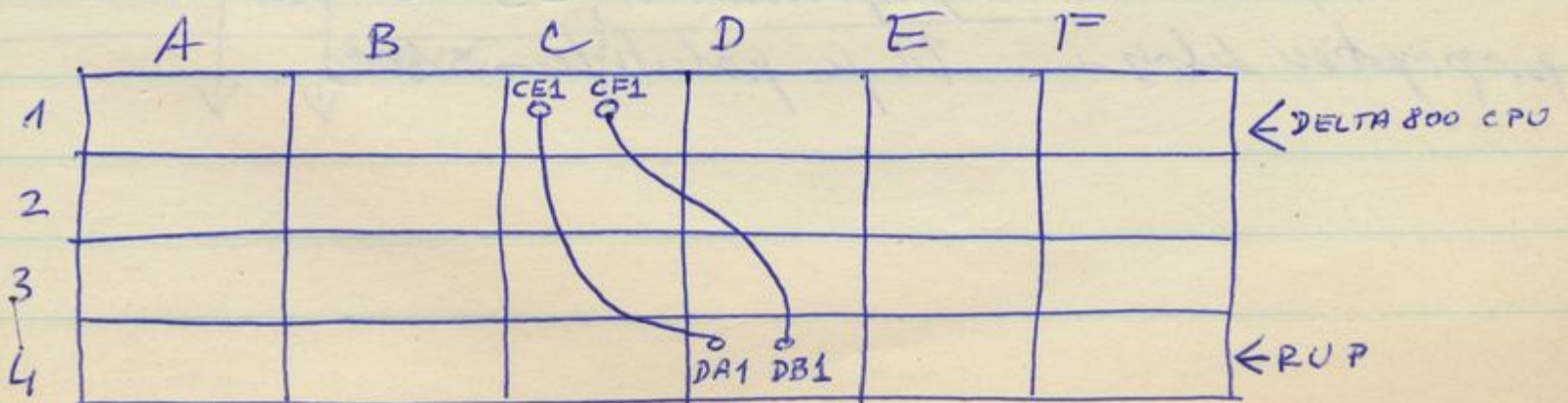
- ① RT NI SPOJEN NA VCC
  - ② E8/12 (74S181) NI SPOJEN NA GND
  - ③ E6/1 (74S153) NI SPOJEN NA GND
  - ④ E71/15 (74S157) NI SPOJEN NA GND
  - ⑤ J3 PIN 3 NI POVEZAN NA E54/1 (74S00) <K1-8 RELOCATE H>
  - ⑥ J3 PIN 37 NE SME BITI VEZAN NA GND  
J3 PIN 37 MORA BITI POVEZAN NA E37/9 (8641) <K21 ENAB ADRS L>
  - ⑦ E109/12 (85S68) NI VEZAN NA GND
  - ⑧ E95/3, 21 (7485) NI VEZAN NA GND
  - ⑨ E95/16 (7485) NI VEZAN NA VCC
  - ⑩ E93/11 (74LS253) : <K1-7 PAR  $\phi 6(1)H$  JE SPOJEN S SIGNALOM <K1-10 ENAB KT MUX L>
- 
- ⑪ E66/8 (74174) NIMA GND
  - ⑫ E91/11 (74F11) JE VEZAN NA GND, MORAL PA BI BITI SIGNAL
  - ⑬ E141/6 (74F00) JE SPOJEN NA GND
  - ⑭ E74/1, 4, 13, 15 (74153) NI SPOJEN NA GND
  - ⑮ E84/5 (6300) NI VEZAN NA GND
  - ⑯ E72/2 NI VEZAN NA GND (74F74)
  - ⑰ NA MESTU C268 STA SPOJENA VCC IN GND
  - ⑱ R255, R209, R261, R226, R227, R228, R229 NISO VEZANI NA VCC
  - ⑲ E172/7 (8815) NI VEZAN NA GND
  - ⑳ E199/1 (8881 ali 96101) POVEZANA NA LAGE 5 SE STIKA Z GND
  - ㉑ E159/4, 5, 6 (7402) INPUT JE 6 OUTPUT JE 4
  - ㉒ E145/4 (7473) NI VEZAN NA VCC
  - ㉓ E143/16 (6331) NI SPOJEN NA VCC
  - ㉔ E142/18 (6353) NI VEZAN NA VCC
  - ㉕ C317 NI SPOJEN NA GND

- 26) E166/15 (6301) IN E165/6 (6300) STA VEZANA NA SIGNAL K1-1 ZBIT (1)H MORAL PA BI BITI SIGNAL K1-1-CBIT (1)H
- 27) NA R231 JE POVEZAVA MED GND IN VCC
- 28) E217/4,5 <sup>(745138)</sup> NI VEZAN NA GND
- 29) J3 PIN 2 MORA BITI VEZAN NA GND
- 30) E58/3 (74F04) NI POVEZAN NA E72/3 (74F74)
- 31) SPOJ MED <sup>GND</sup> IZVRTINO IN SIGNALOM BQ4 <LAGE 5>
- 32) W10 PREVELIKE LUKNE <PARITY ENABLE>
- 33) SIGNAL INT MSYN (K2-1) MORA BITI SPELIAN NA <CE1> IN NE SME BITI POVEZAN NA OSTALE ELEMENTE (POMOR) POMOR SIGNALA INT MSYN (K1-10) mpr E140/11 MORA BITI SPELIAN NA <CF1>
- 34) KONEKTORSI J2/1 IN J2/2 NISO KOMPATIBILNI S KONEKTORSI U4 M3266 ZATO NI MOGOCE UPORABLJATI PROGRAMER'S KONZOLE IN FP-11KOPROCESORJA

# POZICIJA IN VREDNOST PASIVNIH ELEMENTOV



## DK BACKPLANE: DODATNI WRAP



6.1.1986

MERJENJE PORABE CPU DELTA 800

Poraba in zmernjso so cca 30% ce vzamemo STTL & FTTL.

$I_{cpu} = 7,7A$  /  $5V_{Vcc}$  originalno  $11,5A$  /  $5V_{ole}$

Povečanje hitrosti procesorja so 5,21%.

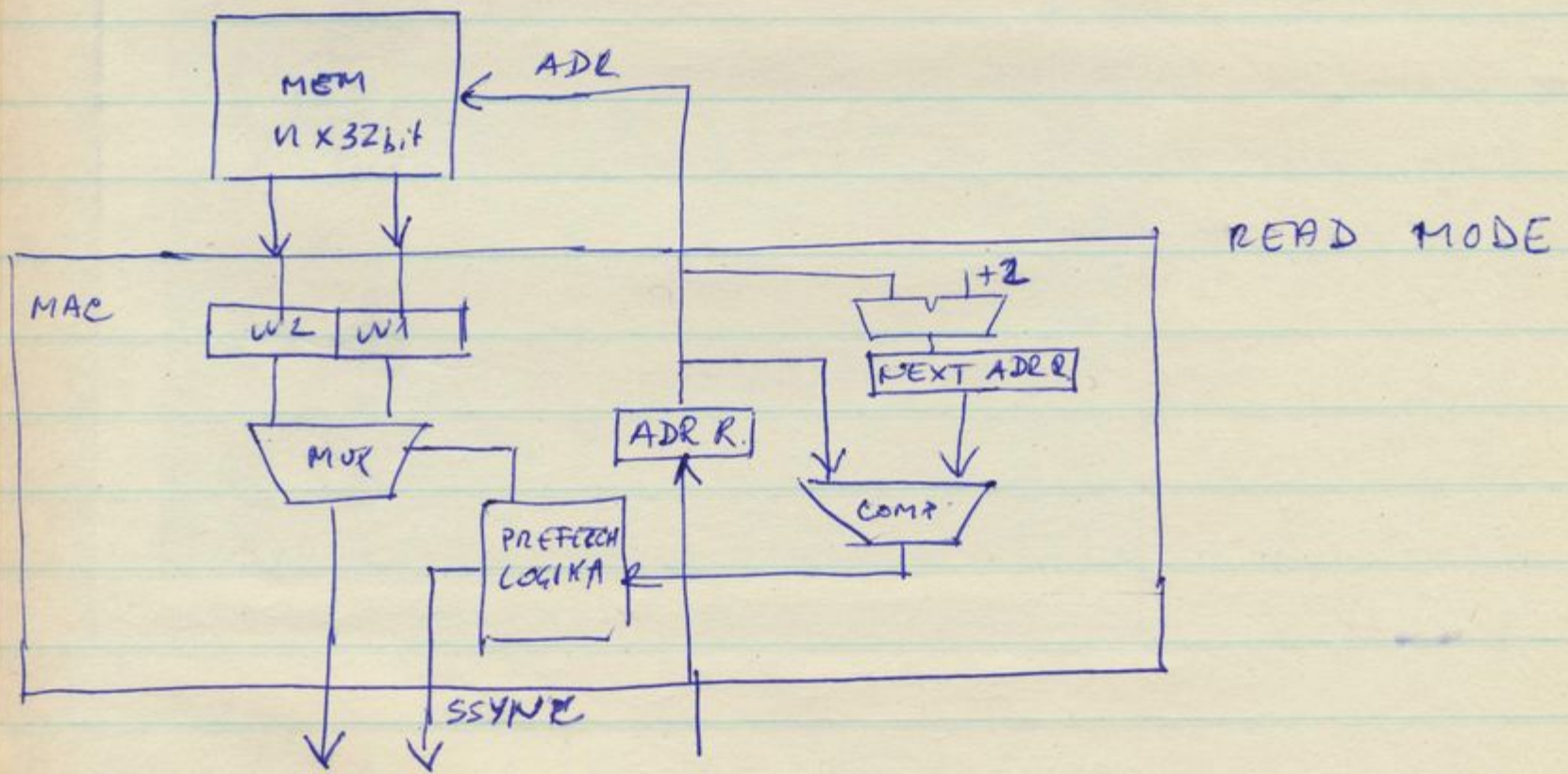
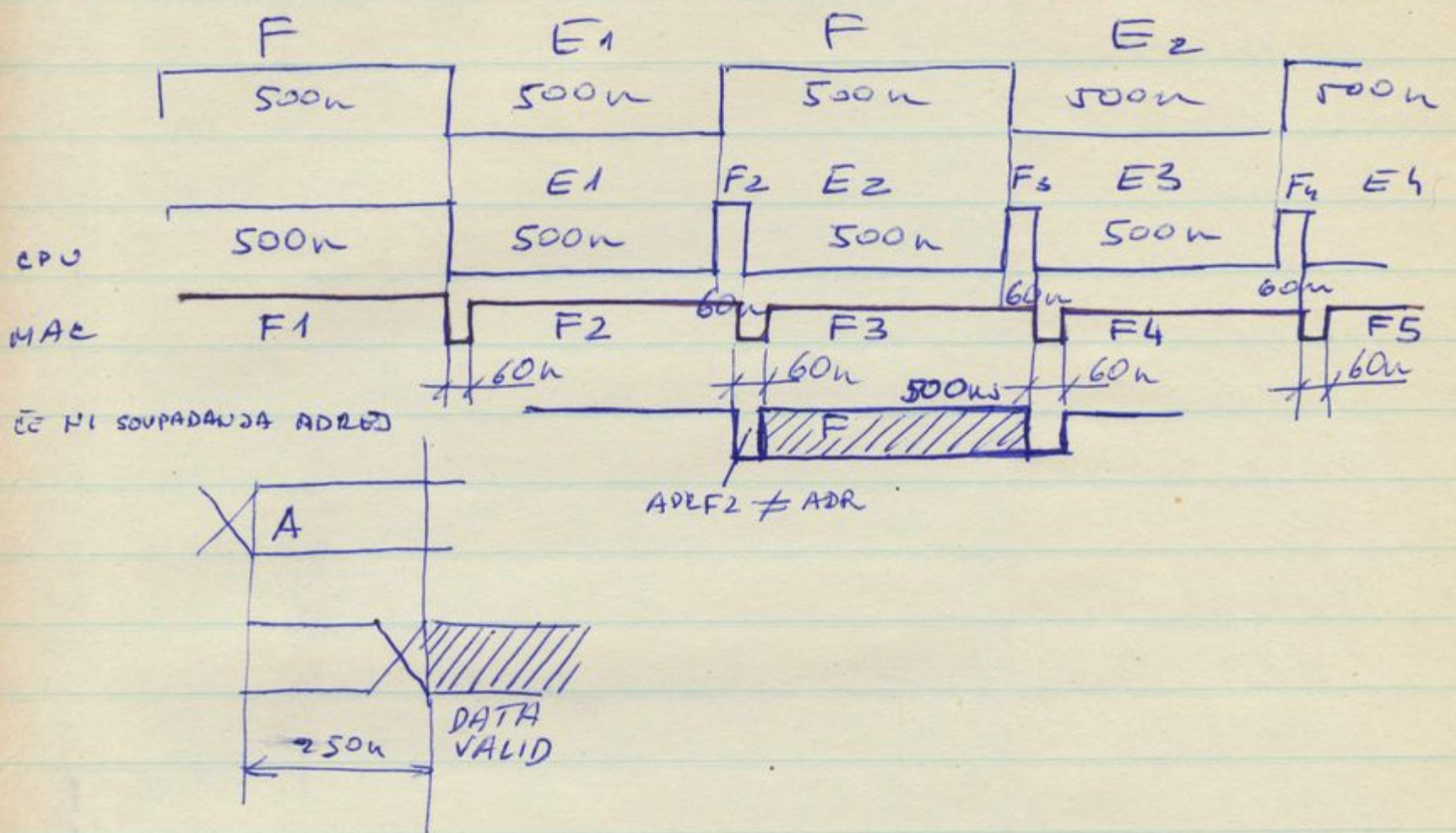
V FETCH fazi je v začetnem mikrociklu fetch faze zehosuter (realizirano z RC konstanto), ki omogoča BUS GRANT arbitracijo. (čas, ki je potreben, da se signal na bus arbitracijske logiki ustali, tako, da se lahko zgodi mikroskok v primeru do obstoja popoj za urisovanje). Če bi bila arbitracijska logika spremenjena tako, da bi izločili to zehosuter, kar povprečni čas za izvršitev instrukcije zmanjšal za 0,150  $\mu s$ .

< za poznavalec gley COMPUTER ENGINEERING str 353 >  
Čas mikrocikla se podaljša za 0,150  $\mu s$  z bus purn tetue arbitracije med BUT SERVICE mikroinstrukcijo.

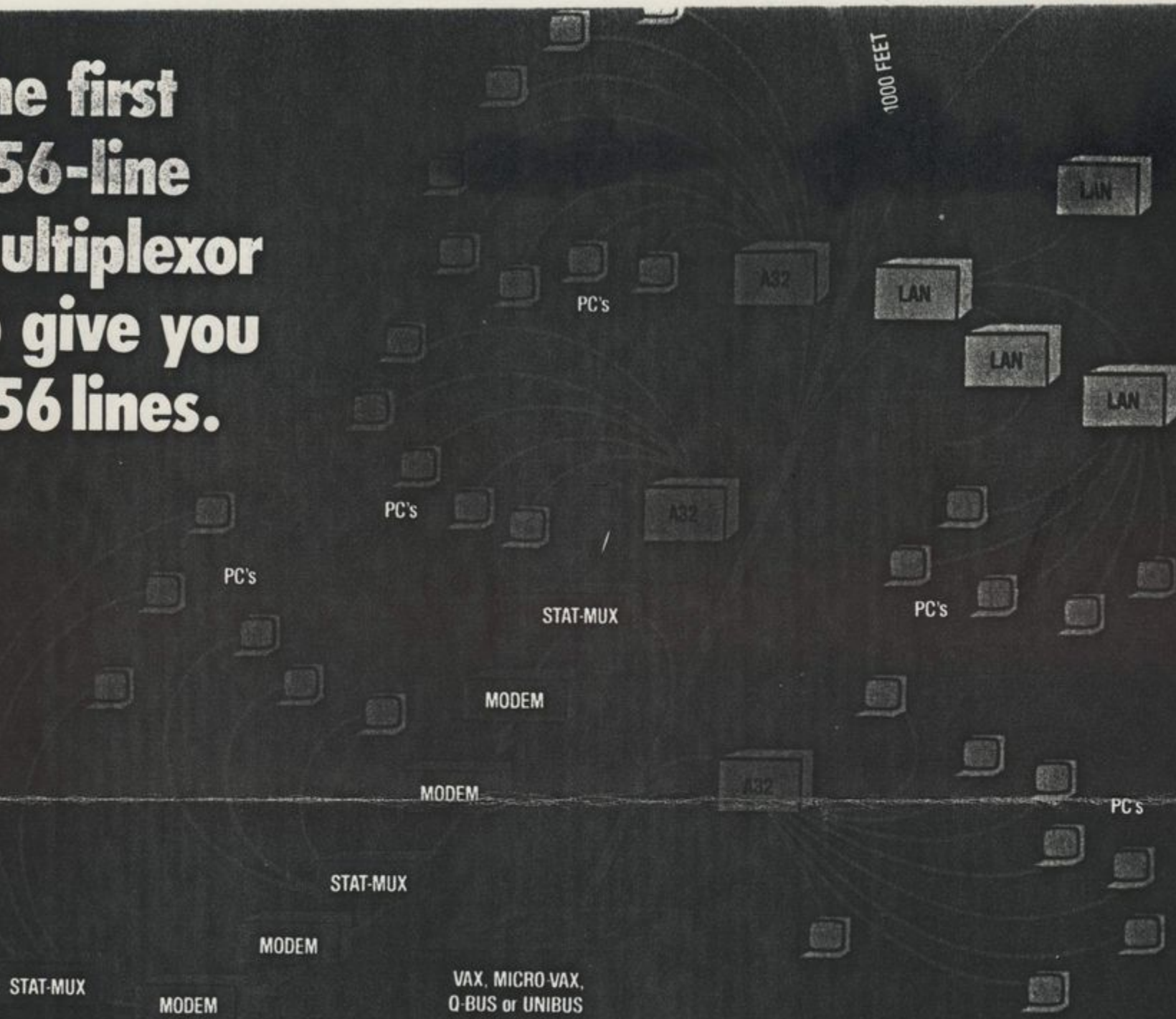
Povečanje hitrosti procesorja pri povečanju hitrosti podoflovnih poti:  $\approx 10\%$

Pri 11/34 so z uporabo ~~STTL~~ <sup>STTL</sup> ~~STTL~~ TTL no podoflovnih poteh dno si kotve signoli čerto pukojojo (npr ALU & in carry look AHEAD) mikroskli 35 ns pri propozitni delog-u. Tu je polutiter 10%.

Povečanje hitrosti sistema z uporabo PREFETCH  
 registera na posamičnem kontrolerju (MAC).



# The first 256-line multiplexor to give you 256 lines.



Now up to 256 asynchronous serial communication devices, from as far away as 1,000 feet, can communicate with DEC's VAX, Micro-VAX, Q-bus and Unibus computers.

MDB's new Digital Multiplexor Family (DMF) of high performance multiplexors was developed because we know systems designers need more efficient ways to control statistical multiplexors, LAN's, dynamic asynchronous terminal line switches and interprocessor links, in the new data communications world.

So from configuration to operation and installation, we've thought of everything to make it possible. Featuring a space-saving quad sized format, our DMF series is comprised of a host processor board, one or more asynchronous channel boards (A32), a selection of connector panels, and the best emulation, such as DHU11.

DHU11 emulation means true 256 line capability, and terminals can be placed in up to 16 separate clusters anywhere along the 1,000 foot bus with each channel board controlling

DEC, VAX, Micro-VAX, Q-bus, Unibus are trademarks of Digital Equipment Corporation.

DMF- MP  
(ONE QUAD SLOT)

up to a maximum of 32 RS-232 terminals running up to 38.4K baud.

The host board uses the 29116 ultra-high performance 16-bit, 100 nanosecond microprocessor, 104-bit wide microword, and special logic structures, for the fastest DMF emulation available.

User versatility counts too. The MDB DMF Multiplexors offer such unequalled through-put advantages as the capability to select silo or DMA transfers automatically; and with 16, 18 or 22-bit addressing, the designer can choose the best configuration for the DEC system used. Also, it operates with fully supported DEC software.

Now that you can finally use all 256 lines, you may find that you like the bottom line best. All of those technological advances make it the lowest per line cost possible.

Extending the capability of DEC systems is what we're all about. That's why you should contact us immediately with your questions on our new DMF Series Multiplexors or our complete line of DEC compatible system enhancers.

Visit us at DEXPO/West '85 - Booth #364 and  
AECEA/Anaheim in January. Booth #612



# UBA ADAPTER

## MOŽNI PRISTOPI

### 1) MDRIBORSKA REGISTEV:

ne vsebuje dovolj dovoljene varjante, tako; da po nastanku ni mogoče izdelati delujočega modula, ki vsoj modula, hi bi lahko bil v določenem času operativen. Definicija interrupe vodila ni dovolj ustrezno podana.

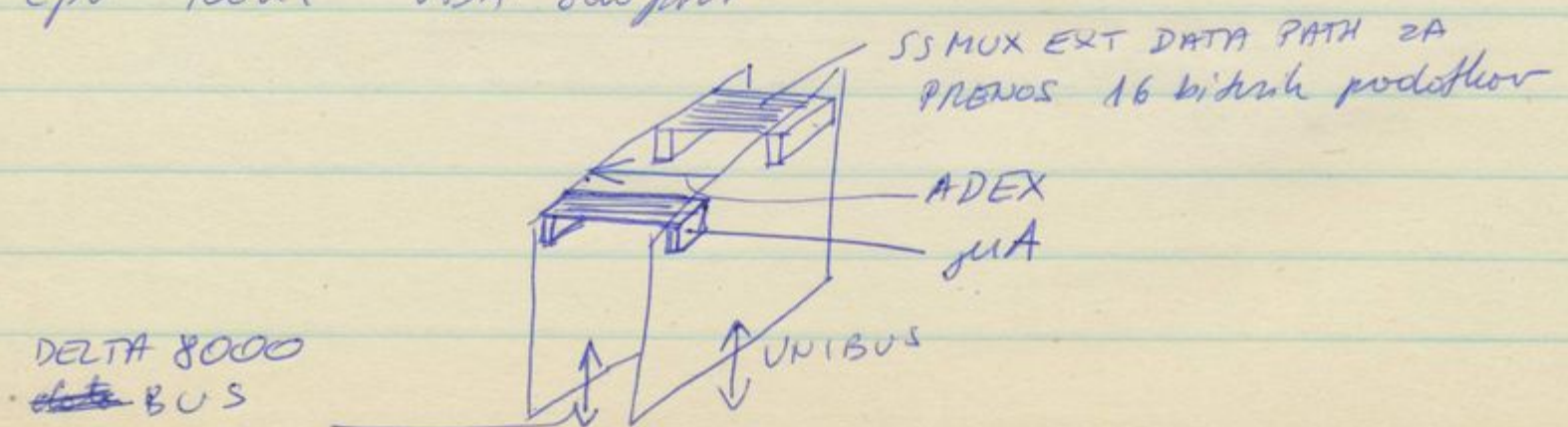
2) Direktna kopija za modula DW 780 ne pride v postelje ker je interrupe vodilo v Delta 8000 drugačno.

3) Kopiranje DV 780, ~~to~~ razije po se interface za vodilo za Delta 8000. Tak način bi verjetno bil dovolj hitro do se v določenem roku razije, naprave. Zopet se lahko komprimira v uporabi PAMOV. Si EPROM verzij če se to isto pokazal kot uporabo. Možna je uporaba originalne DEC diagnostike

4) Razvoj modula ne omori bit sliča elementov 74 AS 883 rep file, 74 AS 8838 BAREZ. SH.

5) Razvoj modula ne omori bit sliča elementov 74 S 116

6) Razvoj Delta 8000 odlopterje, hi bi skupaj z Delta 8000 CPU tvoril UBA odlopter

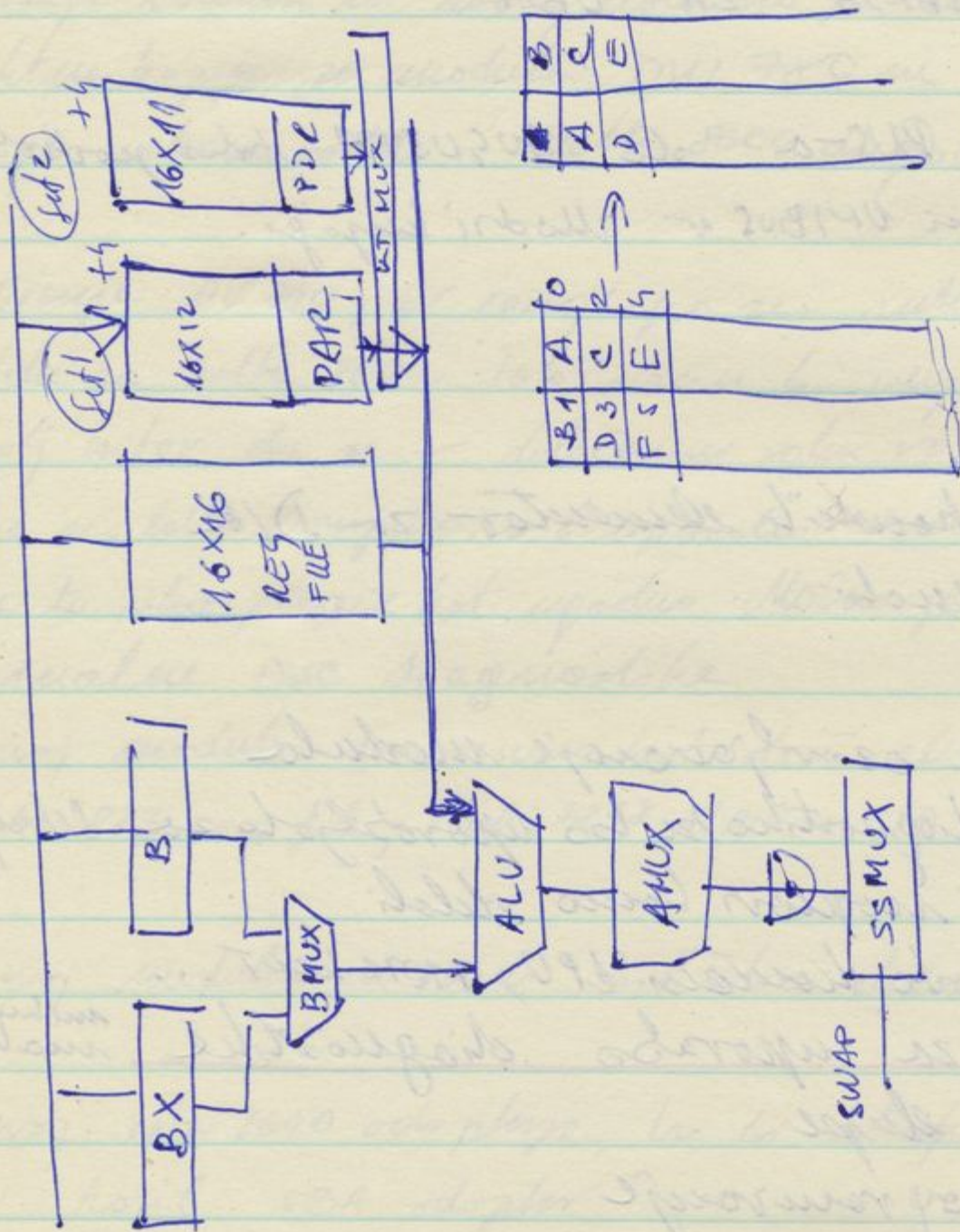


to verjeto, ki negetno razluka najbolj vlogovja. Mi shenier  
no DETA 8000 CPU ploči bi lahko služil za dva modula,  
za Program bi bila potreba spremembi. Deto 8000 odopter  
bi uporabljal isti mikroshenier, vedovd po bi svoj  
mikroprogram. Zasnova predvenceno v 0800 CPU ploči je točno  
do to omogoča.

## NEREJENA VPRAJANJA ZA UBA

- 1) Definicija BUS-a do AUGUSTA točno podrobno,  
kot je podan UMIBUS v Modri knjigi.
  - signali
  - timing
  - protokol
  - tip elektonični elementov za DIR
  - vrsti signali
- 2) Nočim za oživljanje modula
  - katera diagnostika se bo uporabljala za debugging
  - na kateri nivoji bomo delali
  - do kadej bodo kovčani CPU, MEM KONT...
  - solauje za uporabo diagnostike <sup>mikro</sup> <sub>makro</sub>
- 3) formiranje elipe
  - mikroprogramiranje
- 4) število odoptujer v sistemu  
prioriteto fiksno / variabilno

5) Napajanje enovito / ločeno  
 (to je potrebno za PWR / UP / DOWN logiko)



# UPORABA SERIJSKIH SHADOW REGISTROV (SSR) V DIAGNOSTIČNE NAMENE PRI BIPOLARNIH MIKRORAČUNALNI KIH.

## DIAGNOSTIKA

Diagnostika je sklop funkcionalnih in sistematičnih metod za detektiranje in lociranje napak (olov) v sistemu. Nodrivati more določeno testno točko v sistemu in, ker je podoben in odvisni poti, statusnih signalov tako do ~~zajete~~<sup>obruca</sup> kompletno djernost sistema. KONTROLABILNOST in OBSERVABILNOST

Tabela barv za prvi in drugi obroč:

črna	0
rjava	1
rdeča	2
oranžna	3
rumena	4
zelena	5
plava	6
vijoličasta	7
siva	8
bela	9

Tabela barv za multiplikator (pika)

siva	x 0.01
bela	x 0.1
črna	x 1.0
rjava	x 10

Tabela barv za tretji obroč: (napetost)

bela	3V
rumena	6.3V
črna	10V
zelena	16V
plava	20V
siva	25V
roza	35V

Primer: 1.obroč rjav ..... 1

2.obroč zelen ..... 5

3.barva pike bela ..... x 0.1

4.tretji obroč črn .... 10V

Tako označen kondenzator ima vrednost

1.5  $\mu$ F/10V

