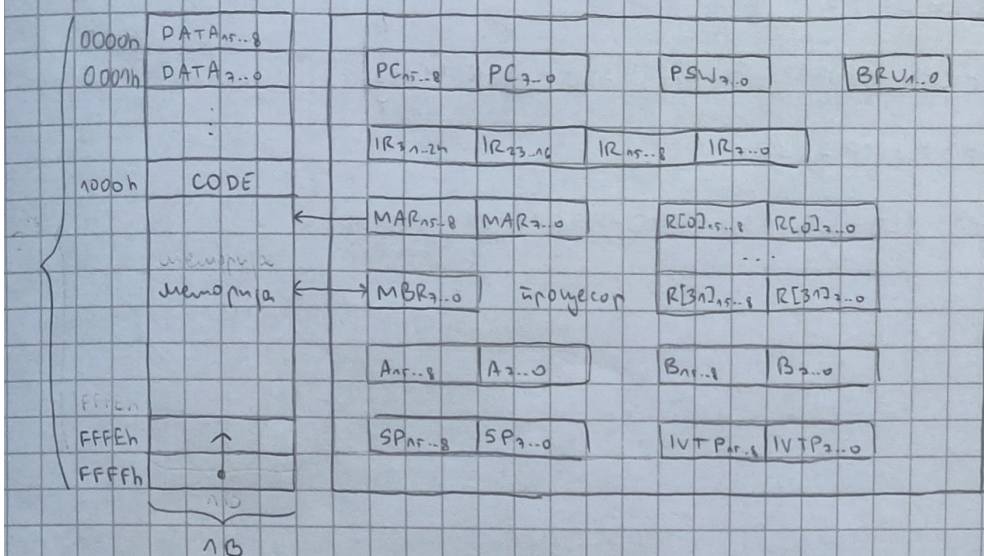


OSNOVI RAČUNARSKE TEHNIKE 2



Pripremila: Lena Joković

- Архитектура рачунара - хардвер



Величина меморије $2^{16}B$, $\frac{2^{16}B}{1B} \rightarrow 2^{16} \rightarrow 16$ бита за адресу

систем расте на нижим локацијама, указује на прву слободну локацију

IVTP - адреса, Interrupt Vector Table (прекидач → програм за обраду)

- инструкције (из IR_{15..0}):

условни скок BZ: 00 00 00 00 | DISP

безусловни скок JMP: 01 00 00 00 | HIGH | LOW

JSR: 01 00 00 01 | HIGH | LOW

безадресне PUSH: 10 00 00 00

POP: 10 00 00 01

RTS: 10 00 00 10

RTI: 10 00 00 11

адресне:

LD: 11 00 00 00

ST: 11 00 00 01

ADD: 11 00 00 10

AND: 11 00 00 11

ASR: 11 00 01 00

JADR: 11 00 01 01

regdir 000 | REG

regind 001 | REG

postdecr 010 | REG

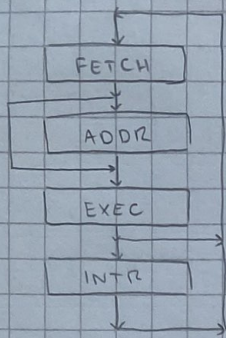
preincr 011 | REG

memdir 100 | ... | HIGH | LOW

memind 101 | ... | HIGH | LOW

regindpwm 110 | REG | DISP

immed 111 | ... | HIGH | LOW



bez adr1 = PUSH + POP + RTS + RTI

branch2 = BZ

adr2 = (LD + ST + ADD + AND + ASR + JADR) · (regdir + regind + postdecr + preincr)

jmp3 = JMP + JSR

adr3 = (LD + ST + ADD + AND + ASR + JADR) · regind · pom

2. Задача (Архитектура процессора-железо)

- меморія $\frac{2^{16}B}{1B} = 2^{16} \rightarrow 16 \text{ біт} \rightarrow 2B$ глибина адреси

- поділ 2B $\rightarrow \begin{array}{|c|c|} \hline 0 & \text{low } 34h \\ \hline 1 & \text{high } 12h \\ \hline \end{array} \rightarrow 1234h$

- інструкції

BNZ: $\begin{array}{|c|} \hline 00000000 \\ \hline \end{array}$ 1. disp 2. 3.

JMP: $\begin{array}{|c|} \hline 00001000 \\ \hline \end{array}$ low high

JSR: $\begin{array}{|c|} \hline 00001001 \\ \hline \end{array}$ low high

PUSH: $\begin{array}{|c|} \hline 11110000 \\ \hline \end{array}$

POP: $\begin{array}{|c|} \hline 11110001 \\ \hline \end{array}$

RTS: $\begin{array}{|c|} \hline 11110010 \\ \hline \end{array}$

RTI: $\begin{array}{|c|} \hline 11110011 \\ \hline \end{array}$

LD: $\begin{array}{|c|} \hline 0001 \\ \hline \end{array}$ 1. 2. 3.

ST $\begin{array}{|c|} \hline 0010 \\ \hline \end{array}$ imm $\begin{array}{|c|} \hline 00xx \\ \hline \end{array}$ low high

SUB $\begin{array}{|c|} \hline 0011 \\ \hline \end{array}$ memdir $\begin{array}{|c|} \hline 01xx \\ \hline \end{array}$ low high

OR $\begin{array}{|c|} \hline 0100 \\ \hline \end{array}$ regindpom: no reg disp

ASL $\begin{array}{|c|} \hline 0101 \\ \hline \end{array}$ regdir: 11 reg

JADR $\begin{array}{|c|} \hline 0110 \\ \hline \end{array}$

- сфера роботи на чипі, утворення на базі байтів $\rightarrow \begin{array}{|c|} \hline 0 \\ \hline \end{array} \begin{array}{|c|} \hline \dots \\ \hline \end{array} \begin{array}{|c|} \hline 1 \\ \hline \end{array}$

без зарп = PUSH + POP + RTS + RTI

зарп = (LD + ST + SUB + OR + ASL + JADR) · regdir

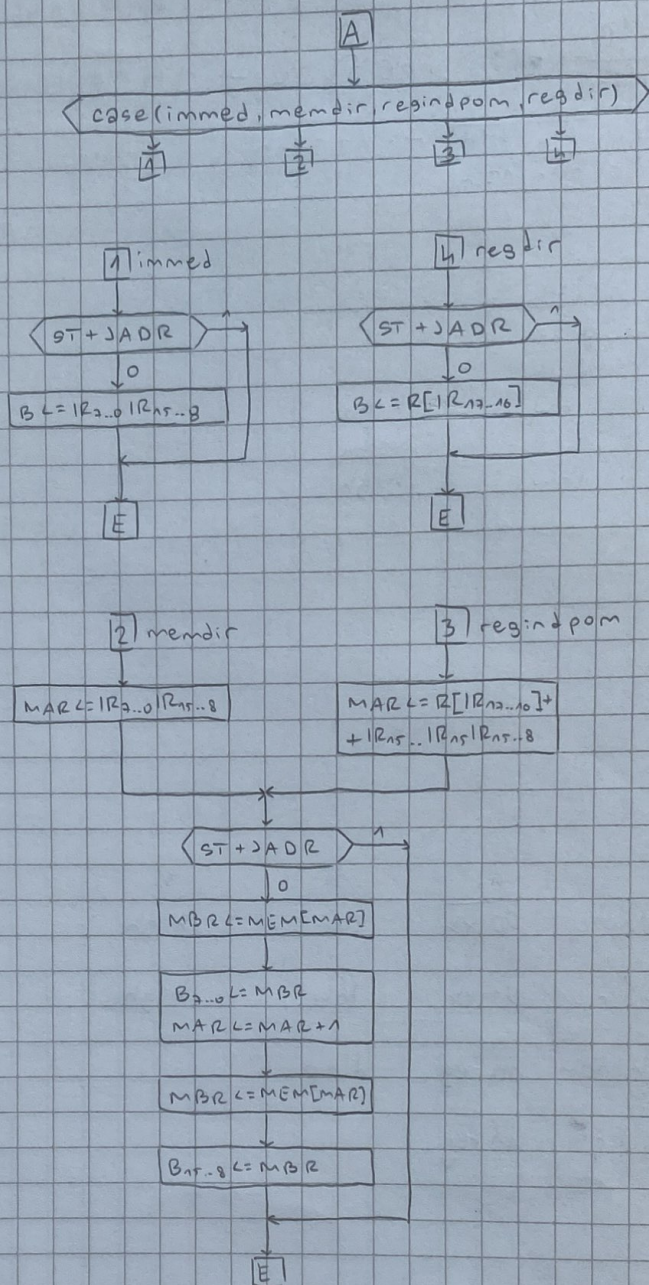
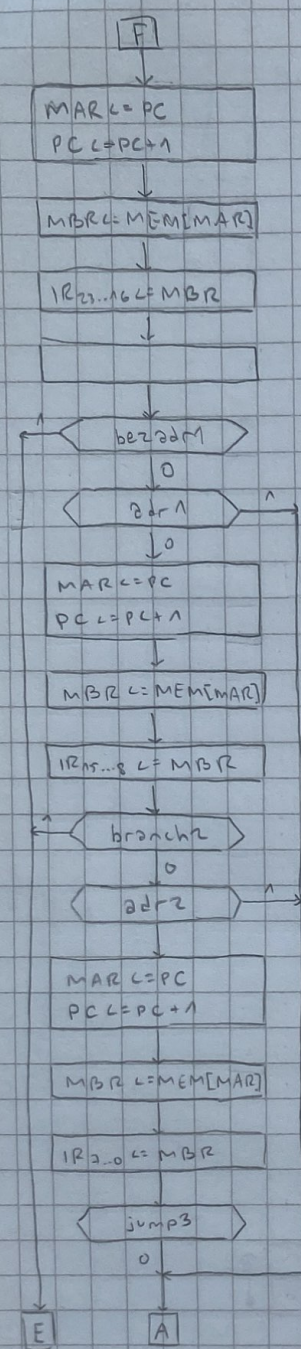
branch2 = BNZ

зарп2 = (LD + ST + SUB + OR + ASL + JADR) · regindpom

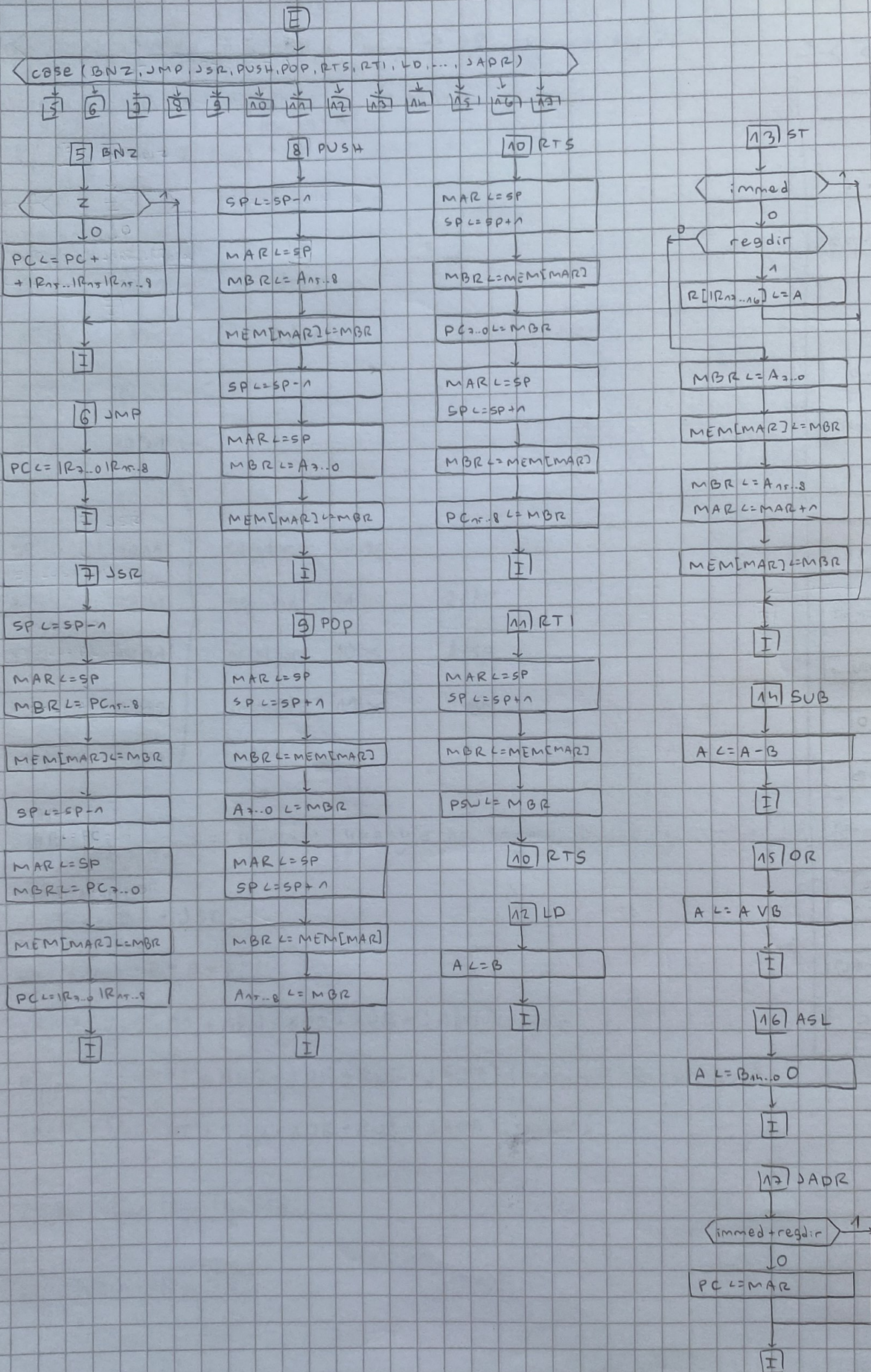
jump3 = JMP + JSR

FETCH:

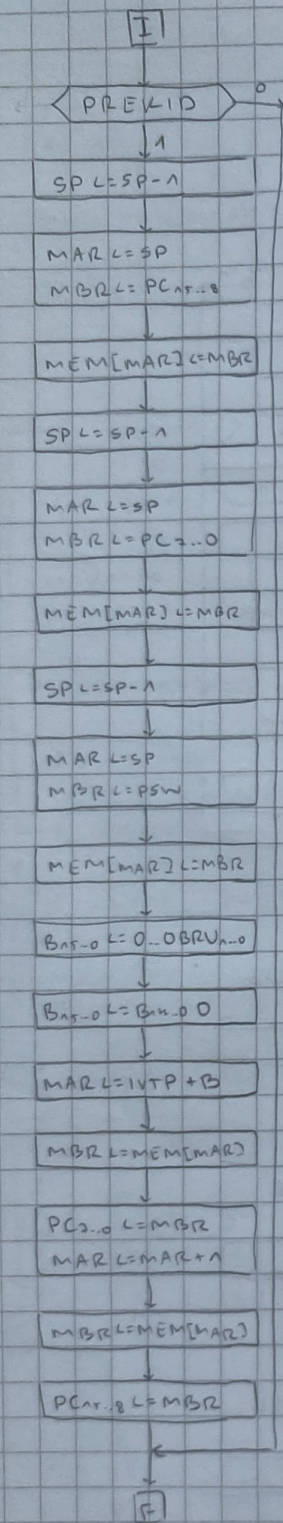
ADDR:



EXEC:



INT 2:



Пример кода на K2:

- меморија: $\frac{2^8 \text{B}}{1 \text{B}} = 2^8 \rightarrow 8 \text{ бита} \rightarrow 1 \text{B}$ дужина адресе

- подаци 2B $\rightarrow$

0	high	12h
1	low	34h

 $\rightarrow$ 1234h

- број улаза 2^4 (16)

- инструкције:

	1.	2.	3.
BC:	0000	disp	
JSR:	0100	addr	
RTS:	1000		
RTI:	1001		
LD:	1010		
ST:	1011	regdir: 00 reg	
ADD:	1100	memind: 01 $\times$	addr
XOR:	1101	pcrel pom: 10 $\times$	disp
ROR:	1110	immed: 11 $\times$	high low
JADR:	1111		

- стил раскривања индикатора указује на адресу заузетости $\rightarrow$

0
...

 $\updownarrow$

$$1) \text{ skok2} = \text{BC} + \text{JSR}$$

$$\text{bez adr1} = \text{RTS} + \text{RTI}$$

$$\text{adr1} = (\text{LD} + \text{ST} + \text{ADD} + \text{XOR} + \text{ROR} + \text{JADR}) \cdot \text{regdir}$$

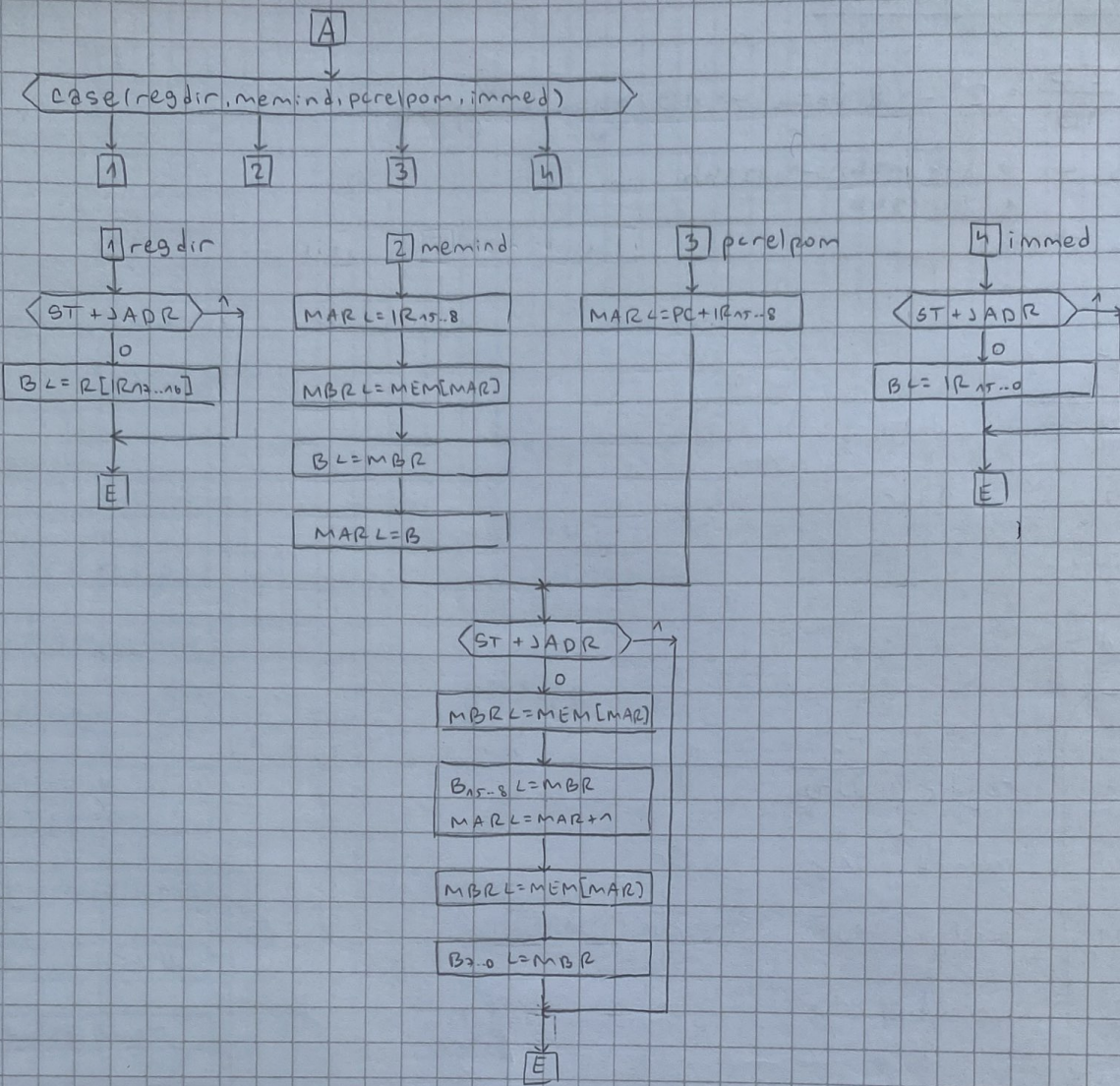
$$\text{adr2} = (\text{LD} + \text{ST} + \text{ADD} + \text{XOR} + \text{ROR} + \text{JADR}) \cdot (\text{memind} + \text{pcrel pom})$$

$$\text{adr3} = (\text{LD} + \text{ST} + \text{ADD} + \text{XOR} + \text{ROR} + \text{JADR}) \cdot \text{immed}$$

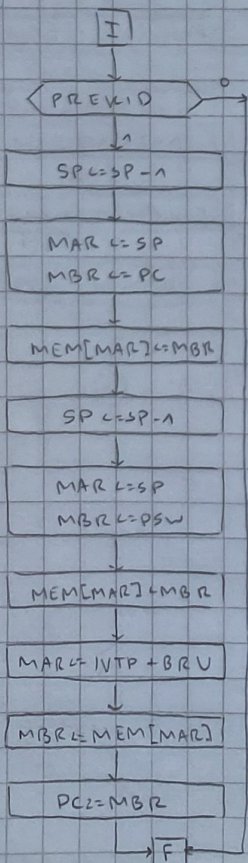
$$\text{XOR} = \overline{R_{23}} \cdot \overline{R_{22}} \cdot \overline{R_{21}} \cdot \overline{R_{20}}$$

$$\text{pcrel pom} = \overline{R_{19}} \cdot \overline{R_{18}}$$

2)



3)



8)

E

Case (BC, JSR, RTS, RTI, LD, ST, ADD, XOR, ROR, JADR)

5

6

7

8

9

10

11

12

13

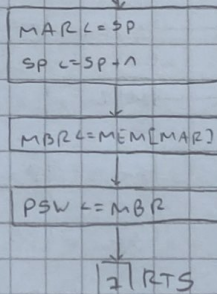
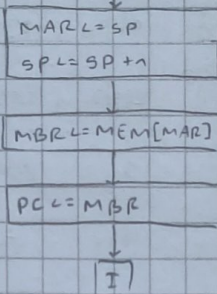
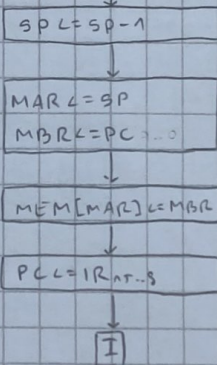
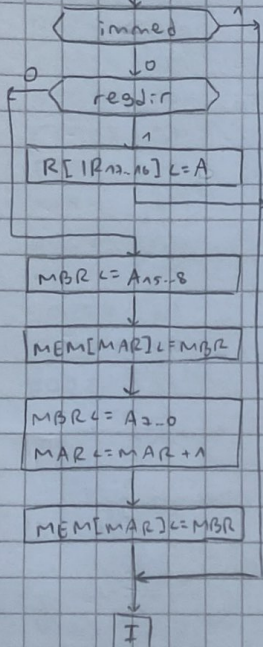
14

10 ST

6 JSR

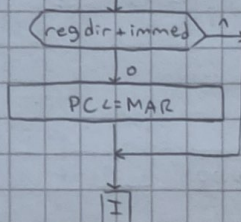
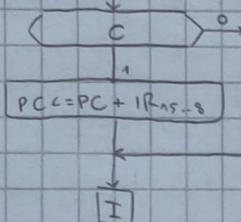
7 RTS

8 RTI



5 BC

14 JADR

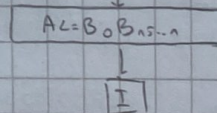
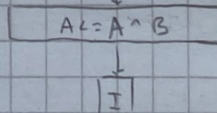
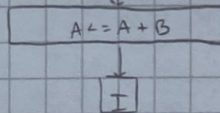
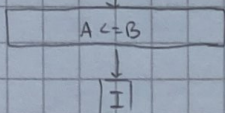


9 LD

11 ADD

12 XOR

13 ROR



3. задаток (Архитектура рачунара-хардвер)

- меморија $\frac{2^{16}}{16} = 2^6 \rightarrow 16 \text{ бита} \rightarrow 2^6 \text{ јединица адреса}$

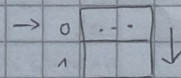
- подаци 16

- број улаза $2^2(4)$

- инструкције:

	1.	2.	3.
RTS:	00000000		
RTI:	00000001		
PUSH:	00000010		
POP:	00000011		
BZ:	01000000	disp	
JMP:	01100000	high	low
JSR:	01100001	high	low
	1.	2.	3.
LD:	10000	regdir	00r
ST:	10001	regind	01r
ADD:	10010	immed	100 data
XOR:	10011	perelpom	101 disp
LSR:	10100	memdir	110 high low
JADR:	10101	memind	111 high low

- сачекање на брину, гласује на засну зауставу



$$\text{no ad-1} = \text{RTS} + \text{RTI} + \text{PUSH} + \text{POP}$$

$$\text{branch 2} = \text{BZ}$$

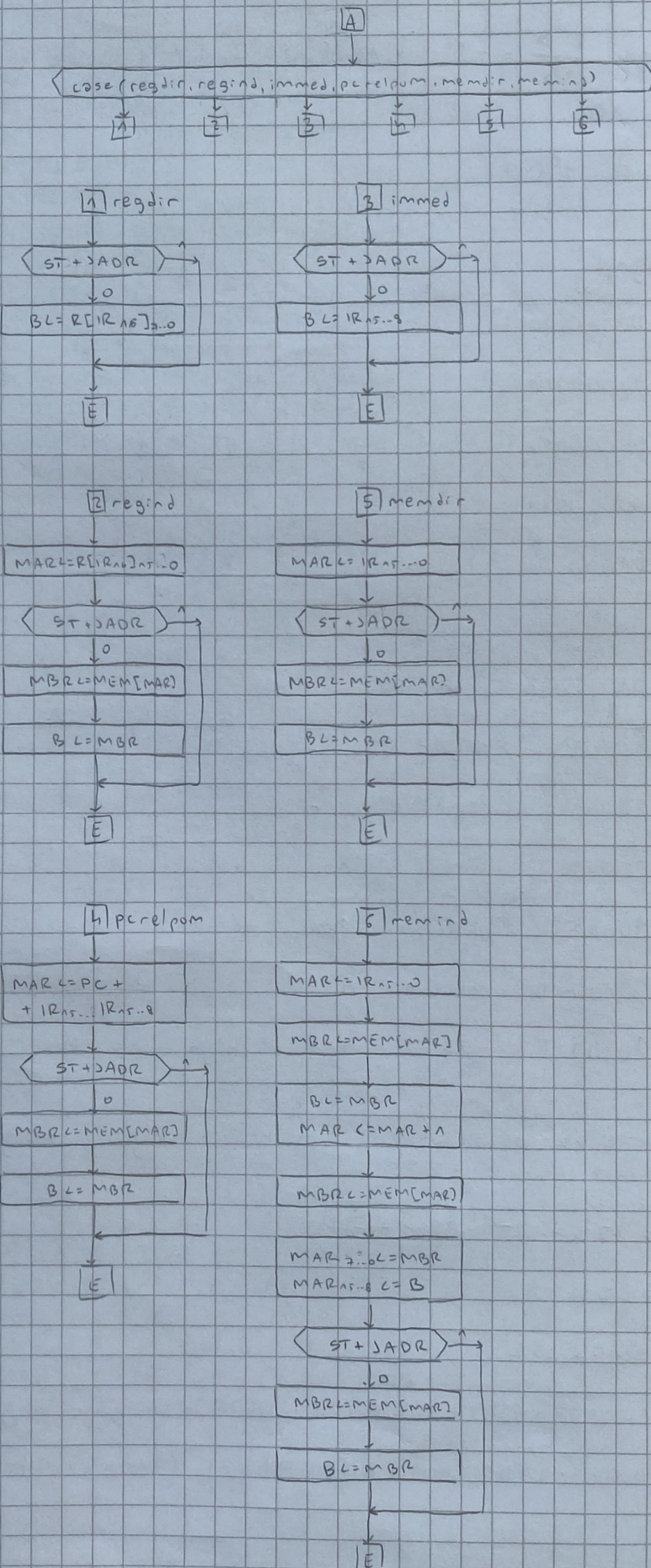
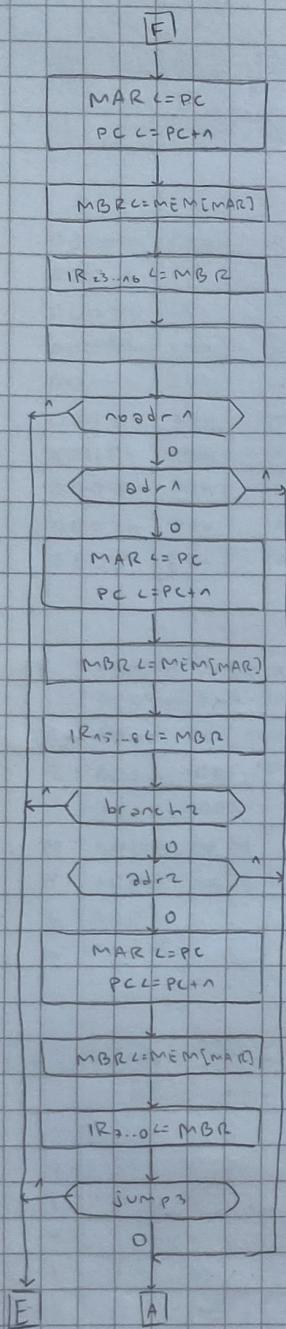
$$\text{jump 3} = \text{JMP} + \text{JSR}$$

$$\text{ad-1} = (\text{LD} + \text{ST} + \text{ADD} + \text{XOR} + \text{LSR} + \text{JADR}) \cdot (\text{regdir} + \text{regind})$$

$$\text{ad-2} = (\text{LD} + \text{ST} + \text{ADD} + \text{XOR} + \text{LSR} + \text{JADR}) \cdot (\text{immed} + \text{perelpom})$$

FETCH:

ADDR:



EXEC:

