

Architektura i organizacja komputerów

Sprawozdanie z laboratorium nr 3

Temat zajęć: Mikroprogramy
rozkazów przesłań i arytmetycznych

Borkowski Kamil WCY22IY1S1

Data wykonania: 2023.11.14

Treść zadania:

Lab2 WCY22IY1S1 zima 2023

Dana jest zawartość początkowa rejestrów i pamięci operacyjnej PAO jak w poniższej tabeli:

Rejestry	
A	-(1000)
LR	160+nr
RI	nr
PAO	
Adres	Zawartość
0	nr
nr	255
nr+1	5000+nr
LR	SUB 001 nr
LR+1	CMA nr
LR+2	LAI nr+1
LR+3	ADX 010 1
LR+4	STA 001 nr
LR+5	TXA 111 nr
254	2021
255	-(1000+nr)

Pozostałe komórki PAO są wyzerowane.

Stopień trudności zadania:

- Na dostatecznie – poprawnie pobrać i wykonać pierwsze 3 rozkazy.
- Na dobrze – poprawnie pobrać i wykonać pierwsze 4 rozkazy.
- Na bardzo dobrze – poprawnie pobrać i wykonać pierwsze 5 rozkazów.

Pozostałe komórki PAO są wyzerowane.

W Pamięci Mikroprogramów mają być wpisane do wytworzenia sprawozdania (najlepiej przed zajęciami, ale niekoniecznie) mikroprogramy, realizujące **wszystkie rozkazy z grup, objętych tematyką dzisiejszych zajęć (bez mnożenia i dzielenia oraz pozostałych z zestawu: MUL, DIV, SIO, LIO, BDN, CND, ENI, LDS)**,

np. mimo że w treści przykładowych zadań nie ma dodawania, to zarówno

- pod adresem 1 w PM ma się znajdować odpowiedni skok do 52,

- jak i pod adresem 52, 53 ma znajdować się mikroprogram dodawania.

Brak kompletnej PM dla bieżących grup rozkazów w sprawozdaniu **oznacza pół oceny w**

dół - nie dotyczy: MUL, DIV, SIO, LIO, BDN, CND, ENI, LDS.

Uwaga: w trakcie tego ćwiczenia **nie wolno edytować RAPS na zero** po zakończeniu pobierania każdego rozkazu - po fazie pobrania nastąpi samoczynnie (o ile została właściwie wypełniona pamięć mikroprogramów) faza wykonania danego rozkazu maszynowego komputera LabZSK.

Wydruk zawartości PM:

0 Test ____TINT Brak przerwania

NA ____48

1 Test ____UNB Zawsze pozytywny

NA ____52

2 Test ____UNB Zawsze pozytywny

NA ____54

5 Test ____UNB Zawsze pozytywny

NA ____56

6 Test ____UNB Zawsze pozytywny

NA ____58

7 Test ____UNB Zawsze pozytywny

NA ____60

8 Test ____UNB Zawsze pozytywny

NA ____62

9 Test ____UNB Zawsze pozytywny

NA ____64

10 Test ____UNB Zawsze pozytywny
NA ____66

11 Test ____UNB Zawsze pozytywny
NA ____68

12 Test ____UNB Zawsze pozytywny
NA ____69

13 Test ____UNB Zawsze pozytywny
NA ____70

33 Test ____UNB Zawsze pozytywny
NA ____106

41 Test ____UNB Zawsze pozytywny
NA ____124

42 Test ____UNB Zawsze pozytywny
NA ____125

43 Test ____UNB Zawsze pozytywny
NA ____126

44 Test ____UNB Zawsze pozytywny
NA ____128

48 S1 ____OLR LR -> BUS
D1 ____IRAP BUS -> RAP

S3 ___ORBP RBP -> BUS
D3 ___IRR BUS -> RR
C1 ___RRC Rozpoczęcie RRC

49 S1 ___ORR RR -> BUS
D1 ___ILK BUS -> LK
S2 ___IRAE SUMA -> RAE
D2 ___NSI LR+1 -> LR
C2 ___CEA Oblicz adres efektywny
Test ___TIND Adresowanie pośrednie
NA ___50

50 S1 ___ORAE RAE -> BUS
D1 ___IRAP BUS -> RAP
S3 ___ORBP RBP -> BUS
D3 ___IX BUS -> X
C1 ___RRC Rozpoczęcie RRC

51 S2 ___OX X -> BUS
D2 ___IBI BUS -> RAE
C2 ___OPC OP albo AOP+32 -> RAPS

52 S1 ___ORAE RAE -> BUS
D1 ___IRAP BUS -> RAP
S3 ___ORBP RBP -> BUS
D3 ___IX BUS -> X
C1 ___RRC Rozpoczęcie RRC

53 S1 ___IALU A -> LALU

D1 ___OXE X -> RALU
S2 ___OBE ALU -> BUS
D2 ___IA BUS -> A
C1 ___END Koniec mikroprogramu
ALU ___ADD ALU = LALU + RALU

54 S1 ___ORAE RAE -> BUS
D1 ___IRAP BUS -> RAP
S3 ___ORBP RBP -> BUS
D3 ___IX BUS -> X
C1 ___RRC Rozpoczęcie RRC

55 S1 ___IALU A -> LALU
D1 ___OXE X -> RALU
S2 ___OBE ALU -> BUS
D2 ___IA BUS -> A
C1 ___END Koniec mikroprogramu
ALU ___SUB ALU = LALU - RALU

56 S1 ___ORAE RAE -> BUS
D1 ___IRAP BUS -> RAP
S3 ___OMQ MQ -> BUS
D3 ___IRBP BUS -> RBP
C1 ___CWC Rozpoczęcie CWC

57 C1 ___END Koniec mikroprogramu

58 S1 ___ORAE RAE -> BUS
D1 ___IRAP BUS -> RAP

S3 ___OA A -> BUS
D3 ___IRBP BUS -> RBP
C1 ___CWC Rozpoczęcie CWC

59 C1 ___END Koniec mikroprogramu

60 S1 ___ORAE RAE -> BUS
D1 ___IRAP BUS -> RAP
S3 ___ORI RI -> BUS
D3 ___IRBP BUS -> RBP
C1 ___CWC Rozpoczęcie CWC

61 C1 ___END Koniec mikroprogramu

62 S1 ___ORAE RAE -> BUS
D1 ___IRAP BUS -> RAP
S3 ___ORBP RBP -> BUS
D3 ___IA BUS -> A
C1 ___RRC Rozpoczęcie RRC

63 C1 ___END Koniec mikroprogramu

64 S1 ___ORAE RAE -> BUS
D1 ___IRAP BUS -> RAP
S3 ___ORBP RBP -> BUS
D3 ___IRI BUS -> RI
C1 ___RRC Rozpoczęcie RRC

65 C1 ___END Koniec mikroprogramu

66 S1 ___ORAE RAE -> BUS
 D1 ___IRAP BUS -> RAP
 S3 ___OLR LR -> BUS
 D3 ___IRBP BUS -> RBP
 C1 ___CWC Rozpoczęcie CWC

67 C1 ___END Koniec mikroprogramu

68 S2 ___ORI RI -> BUS
 D2 ___IA BUS -> A
 C1 ___END Koniec mikroprogramu

69 S2 ___OMQ MQ -> BUS
 D2 ___IA BUS -> A
 C1 ___END Koniec mikroprogramu

70 S1 ___ORAE RAE -> BUS
 D1 ___IRAP BUS -> RAP
 S3 ___ORBP RBP -> BUS
 D3 ___IX BUS -> X
 C1 ___RRC Rozpoczęcie RRC

71 S1 ___IXRE RI -> LALU
 D1 ___OXE X -> RALU
 S2 ___OBE ALU -> BUS
 D2 ___IRI BUS -> RI
 C1 ___END Koniec mikroprogramu
 ALU ___ADD ALU = LALU + RALU

106 S1 ___IALU A -> LALU
S2 ___OBE ALU -> BUS
D2 ___IA BUS -> A
C1 ___END Koniec mikroprogramu
ALU ___CMA ALU = (NOT LALU)+1

124 S2 ___IRAE SUMA -> RAE
S3 ___ORAE RAE -> BUS
D3 ___IA BUS -> A
C1 ___END Koniec mikroprogramu

125 S2 ___IRAE SUMA -> RAE
S3 ___ORAE RAE -> BUS
D3 ___IRI BUS -> RI
C1 ___END Koniec mikroprogramu

126 S2 ___IRAE SUMA -> RAE
S3 ___ORAE RAE -> BUS
D3 ___IX BUS -> X

127 S1 ___IXRE RI -> LALU
D1 ___OXE X -> RALU
S2 ___OBE ALU -> BUS
D2 ___IRI BUS -> RI
C1 ___END Koniec mikroprogramu
ALU ___ADD ALU = LALU + RALU

128 S2 ___IRAE SUMA -> RAE

S3 ___ORAE RAE -> BUS

D3 ___IX BUS -> X

129 S1 ___IXRE RI -> LALU

D1 ___OXE X -> RALU

S2 ___OBE ALU -> BUS

D2 ___IRI BUS -> RI

C1 ___END Koniec mikroprogramu

ALU ___SUB ALU = LALU - RALU

Wydruk zawartości PAO:

0	0000000000000111b	0007h	7		
7	0000000011111111b	00FFh	255		
8	0001001110001111b	138Fh	5007		
167	0001000100000111b	1107h	OP=2	XSI=001	DA=7
168	0000000010000111b	0087h	AOP=1		N=7
169	0000010010001000b	0488h	AOP=9		N=8
170	0110101000000001b	6A01h	OP=13	XSI=010	DA=1
171	0011000100000111b	3107h	OP=6	XSI=001	DA=7
172	0101111100000111b	5F07h	OP=11	XSI=111	DA=7
254	0000011111100101b	07E5h	2021		
255	1111110000010001b	FC11h	-1007		

Wydruk logu z wykonania ćwiczenia:

Start symulatora 2023-11-14 08:55:40

Stacja "WAT-KOMPUTER"

Zalogowano jako: "Student"

Wersja aplikacji: 1.2.3.0

Dostępne interfejsy sieciowe: 169.254.8.55

10.6.9.2

192.168.56.1

=====Start symulacji=====

09:03.59

=====Zawartość rejestrów=====

LK = 0h 0

A = FC18h -1000

MQ = 0h 0

X = 0h 0

RAP = 0h 0

LALU = 0h 0

RALU = 0h 0

RBP = 0h 0

ALU = 0h 0

BUS = 0h 0

RR = 0h 0

LR = A7h 167

RI = 7h 7

RAPS = 0h 0

RAE = 0h 0

L = 0h 0

R = 0h 0

SUMA = 0h 0

MAV = 1, IA = 0, INT = 0

ZNAK = 0, XRO = 0, OFF = 0

MAKRO

=====0=====

Takt0: RBPS=000000020030h

Takt7:

INT = 0

TEST | TINT : Brak przerwania(INT != 0)

RAPS = 48 / 30h

MAKRO

=====48=====

Takt0: RBPS=5006C4000000h

Takt1:

S1 | OLR : LR -> BUS

BUS = 167 / A7h

D1 | IRAP : BUS -> RAP

RAP = 167 / A7h

C1 | RRC : Rozpoczęcie RRC

RBP = 4359 / 1107h

Takt7:

S3 | ORBP : RBP -> BUS

BUS = 4359 / 1107h

D3 | IRR : BUS -> RR

RR = 4359 / 1107h

RAPS = 49 / 31h

MAKRO

=====49=====

Takt0: RBPS=68C801830032h

Takt1:

S1 | ORR : RR -> BUS

BUS = 4359 / 1107h

D1 | ILK : BUS -> LK

LK = 7 / 7h

C2 | CEA : Oblicz adres efektywny

L = 7 / 7h

R = 0 / 0h

SUMA = 7 / 7h

XRO = 0

Takt6:

S2 | IRAE : SUMA -> RAE

RAE = 7 / 7h

D2 | NSI : LR+1 -> LR

LR = 168 / A8h

Takt7:

TEST | TIND : Adresowanie pośrednie

RAPS = 50 / 32h

MAKRO

=====50=====

Takt0: RBPS=900624000000h

Takt1:

S1 | ORAE : RAE -> BUS

BUS = 7 / 7h

D1 | IRAP : BUS -> RAP

RAP = 7 / 7h

C1 | RRC : Rozpoczęcie RRC

RBP = 255 / FFh

Takt7:

S3 | ORBP : RBP -> BUS

BUS = 255 / FFh

D3 | IX : BUS -> X

X = 255 / FFh

RAPS = 51 / 33h

MAKRO

=====51=====

Takt0: RBPS=03A801600000h

Takt6:

S2 | OX : X -> BUS

BUS = 255 / FFh

D2 | IBI : BUS -> RAE

RAE = 255 / FFh

Takt7:

C2 | OPC : OP albo AOP+32 -> RAPS

RAPS = 2 / 2h

MAKRO

=====2=====

Takt0: RBPS=0000000010036h

Takt7:

TEST | UNB : Zawsze pozytywny

RAPS = 54 / 36h

MAKRO

=====54=====

Takt0: RBPS=900624000000h

Takt1:

S1 | ORAE : RAE -> BUS

BUS = 255 / FFh

D1 | IRAP : BUS -> RAP

RAP = 255 / FFh

C1 | RRC : Rozpoczęcie RRC

RBP = -1007 / FC11h

Takt7:

S3 | ORBP : RBP -> BUS

BUS = -1007 / FC11h

D3 | IX : BUS -> X

X = -1007 / FC11h

RAPS = 55 / 37h

MAKRO

=====55=====

Takt0: RBPS=BC300E000200h

Takt1:

S1 | IALU : A -> LALU

LALU = -1000 / FC18h

D1 | OXE : X -> RALU

RALU = -1007 / FC11h

Takt2:

ALU | SUB : ALU = LALU - RALU

ALU = 7 / 7h

ZNAK = 0, OFF = 0

Takt6:

S2 | OBE : ALU -> BUS

BUS = 7 / 7h

D2 | IA : BUS -> A

A = 7 / 7h

Takt7:

C1 | END : (Cykl 8) Koniec mikroprogramu (09:08.06)

RAPS = 0 / 0h

MAKRO

=====0=====

Takt0: RBPS=000000020030h

Takt7:

INT = 0

TEST | TINT : Brak przerwania(INT != 0)

RAPS = 48 / 30h

MAKRO

=====48=====

Takt0: RBPS=5006C4000000h

Takt1:

S1 | OLR : LR -> BUS

BUS = 168 / A8h

D1 | IRAP : BUS -> RAP

RAP = 168 / A8h

C1 | RRC : Rozpoczęcie RRC

RBP = 135 / 87h

Takt7:

S3 | ORBP : RBP -> BUS

BUS = 135 / 87h

D3 | IRR : BUS -> RR

RR = 135 / 87h

RAPS = 49 / 31h

MAKRO

=====49=====

Takt0: RBPS=68C801830032h

Takt1:

S1 | ORR : RR -> BUS

BUS = 135 / 87h

D1 | ILK : BUS -> LK

LK = 7 / 7h

C2 | CEA : Oblicz adres efektywny

L = 7 / 7h

R = 0 / 0h

SUMA = 7 / 7h

Takt6:

S2 | IRAE : SUMA -> RAE

RAE = 7 / 7h

D2 | NSI : LR+1 -> LR

LR = 169 / A9h

Takt7:

TEST | TIND : Adresowanie pośrednie

RAPS = 50 / 32h

MAKRO

=====50=====

Takt0: RBPS=900624000000h

Takt1:

S1 | ORAE : RAE -> BUS

BUS = 7 / 7h

D1 | IRAP : BUS -> RAP

RAP = 7 / 7h

C1 | RRC : Rozpoczęcie RRC

RBP = 255 / FFh

Takt7:

S3 | ORBP : RBP -> BUS

BUS = 255 / FFh

D3 | IX : BUS -> X

X = 255 / FFh

RAPS = 51 / 33h

MAKRO

=====51=====

Takt0: RBPS=03A801600000h

Takt6:

S2 | OX : X -> BUS

BUS = 255 / FFh

D2 | IBI : BUS -> RAE

RAE = 255 / FFh

Takt7:

C2 | OPC : OP albo AOP+32 -> RAPS

RAPS = 33 / 21h

MAKRO

=====33=====

Takt0: RBPS=00000001006Ah

Takt7:

TEST | UNB : Zawsze pozytywny

RAPS = 106 / 6Ah

MAKRO

=====106=====

Takt0: RBPS=A4300E000400h

Takt1:

S1 | IALU : A -> LALU

LALU = 7 / 7h

Takt2:

ALU | CMA : ALU = (NOT LALU)+1

ALU = -7 / FFF9h

ZNAK = 1, OFF = 0

Takt6:

S2 | OBE : ALU -> BUS

BUS = -7 / FFF9h

D2 | IA : BUS -> A

A = -7 / FFF9h

Takt7:

C1 | END : (Cykl 15) Koniec mikroprogramu (09:11.09)

RAPS = 0 / 0h

MAKRO

=====0=====

Takt0: RBPS=0000000020030h

Takt7:

INT = 0

TEST | TINT : Brak przerwania(INT != 0)

RAPS = 48 / 30h

MAKRO

=====48=====

Takt0: RBPS=5006C4000000h

Takt1:

S1 | OLR : LR -> BUS

BUS = 169 / A9h

D1 | IRAP : BUS -> RAP

RAP = 169 / A9h

C1 | RRC : Rozpoczęcie RRC

RBP = 1160 / 488h

Takt7:

S3 | ORBP : RBP -> BUS

BUS = 1160 / 488h

D3 | IRR : BUS -> RR

RR = 1160 / 488h

RAPS = 49 / 31h

MAKRO

=====49=====

Takt0: RBPS=68C801830032h

Takt1:

S1 | ORR : RR -> BUS

BUS = 1160 / 488h

D1 | ILK : BUS -> LK

LK = 8 / 8h

C2 | CEA : Oblicz adres efektywny

L = 8 / 8h

R = 0 / 0h

SUMA = 8 / 8h

Takt6:

S2 | IRAE : SUMA -> RAE

RAE = 8 / 8h

D2 | NSI : LR+1 -> LR

LR = 170 / AAh

Takt7:

TEST | TIND : Adresowanie pośrednie

RAPS = 50 / 32h

MAKRO

=====50=====

Takt0: RBPS=900624000000h

Takt1:

S1 | ORAE : RAE -> BUS

BUS = 8 / 8h

D1 | IRAP : BUS -> RAP

RAP = 8 / 8h

C1 | RRC : Rozpoczęcie RRC

RBP = 5007 / 138Fh

Takt7:

S3 | ORBP : RBP -> BUS

BUS = 5007 / 138Fh

D3 | IX : BUS -> X

X = 5007 / 138Fh

RAPS = 51 / 33h

MAKRO

=====51=====

Takt0: RBPS=03A801600000h

Takt6:

S2 | OX : X -> BUS

BUS = 5007 / 138Fh

D2 | IBI : BUS -> RAE

RAE = 5007 / 138Fh

Takt7:

C2 | OPC : OP albo AOP+32 -> RAPS

RAPS = 41 / 29h

MAKRO

=====41=====

Takt0: RBPS=000000001007Ch

Takt7:

TEST | UNB : Zawsze pozytywny

RAPS = 124 / 7Ch

MAKRO

=====124=====

Takt0: RBPS=00846E000000h

Takt6:

S2 | IRAE : SUMA -> RAE

RAE = 8 / 8h

Takt7:

S3 | ORAE : RAE -> BUS

BUS = 8 / 8h

D3 | IA : BUS -> A

A = 8 / 8h

C1 | END : (Cykl 22) Koniec mikroprogramu (09:16.43)

RAPS = 0 / 0h

MAKRO

=====0=====

Takt0: RBPS=000000020030h

Takt7:

INT = 0

TEST | TINT : Brak przerwania(INT != 0)

RAPS = 48 / 30h

MAKRO

=====48=====

Takt0: RBPS=5006C4000000h

Takt1:

S1 | OLR : LR -> BUS

BUS = 170 / AAh

D1 | IRAP : BUS -> RAP

RAP = 170 / AAh

C1 | RRC : Rozpoczęcie RRC

RBP = 27137 / 6A01h

Takt7:

S3 | ORBP : RBP -> BUS

BUS = 27137 / 6A01h

D3 | IRR : BUS -> RR

RR = 27137 / 6A01h

RAPS = 49 / 31h

MAKRO

=====49=====

Takt0: RBPS=68C801830032h

Takt1:

S1 | ORR : RR -> BUS

BUS = 27137 / 6A01h

D1 | ILK : BUS -> LK

LK = 1 / 1h

C2 | CEA : Oblicz adres efektywny

L = 1 / 1h

R = 170 / AAh

SUMA = 171 / ABh

XRO = 0

Takt6:

S2 | IRAE : SUMA -> RAE

RAE = 171 / ABh

D2 | NSI : LR+1 -> LR

LR = 171 / ABh

Takt7:

TEST | TIND : Adresowanie pośrednie

RAPS = 13 / Dh

MAKRO

=====13=====

Takt0: RBPS=000000010046h

Takt7:

TEST | UNB : Zawsze pozytywny

RAPS = 70 / 46h

MAKRO

=====70=====

Takt0: RBPS=900624000000h

Takt1:

S1 | ORAE : RAE -> BUS

BUS = 171 / ABh

D1 | IRAP : BUS -> RAP

RAP = 171 / ABh

C1 | RRC : Rozpoczęcie RRC

RBP = 12551 / 3107h

Takt7:

S3 | ORBP : RBP -> BUS

BUS = 12551 / 3107h

D3 | IX : BUS -> X

X = 12551 / 3107h

RAPS = 71 / 47h

MAKRO

=====71=====

Takt0: RBPS=3C200E000100h

Takt1:

S1 | IXRE : RI -> LALU

LALU = 7 / 7h

D1 | OXE : X -> RALU

RALU = 12551 / 3107h

Takt2:

ALU | ADD : ALU = LALU + RALU

ALU = 12558 / 310Eh

ZNAK = 0, OFF = 0

Takt6:

S2 | OBE : ALU -> BUS

BUS = 12558 / 310Eh

D2 | IRI : BUS -> RI

RI = 12558 / 310Eh

Takt7:

C1 | END : (Cykl 28) Koniec mikroprogramu (09:20.46)

RAPS = 0 / 0h

MAKRO

=====0=====

Takt0: RBPS=000000020030h

Takt7:

INT = 0

TEST | TINT : Brak przerwania(INT != 0)

RAPS = 48 / 30h

MAKRO

=====48=====

Takt0: RBPS=5006C4000000h

Takt1:

S1 | OLR : LR -> BUS

BUS = 171 / ABh

D1 | IRAP : BUS -> RAP

RAP = 171 / ABh

C1 | RRC : Rozpoczęcie RRC

RBP = 12551 / 3107h

Takt7:

S3 | ORBP : RBP -> BUS

BUS = 12551 / 3107h

D3 | IRR : BUS -> RR

RR = 12551 / 3107h

RAPS = 49 / 31h

MAKRO

=====49=====

Takt0: RBPS=68C801830032h

Takt1:

S1 | ORR : RR -> BUS

BUS = 12551 / 3107h

D1 | ILK : BUS -> LK

LK = 7 / 7h

C2 | CEA : Oblicz adres efektywny

L = 7 / 7h

R = 0 / 0h

SUMA = 7 / 7h

XRO = 0

Takt6:

S2 | IRAE : SUMA -> RAE

RAE = 7 / 7h

D2 | NSI : LR+1 -> LR

LR = 172 / ACh

Takt7:

TEST | TIND : Adresowanie pośrednie

RAPS = 50 / 32h

MAKRO

=====50=====

Takt0: RBPS=900624000000h

Takt1:

S1 | ORAE : RAE -> BUS

BUS = 7 / 7h

D1 | IRAP : BUS -> RAP

RAP = 7 / 7h

C1 | RRC : Rozpoczęcie RRC

RBP = 255 / FFh

Takt7:

S3 | ORBP : RBP -> BUS

BUS = 255 / FFh

D3 | IX : BUS -> X

X = 255 / FFh

RAPS = 51 / 33h

MAKRO

=====51=====

Takt0: RBPS=03A801600000h

Takt6:

S2 | OX : X -> BUS

BUS = 255 / FFh

D2 | IBI : BUS -> RAE

RAE = 255 / FFh

Takt7:

C2 | OPC : OP albo AOP+32 -> RAPS

RAPS = 6 / 6h

MAKRO

=====6=====

Takt0: RBPS=00000001003Ah

Takt7:

TEST | UNB : Zawsze pozytywny

RAPS = 58 / 3Ah

MAKRO

=====58=====

Takt0: RBPS=9003D2000000h

Takt1:

S1 | ORAE : RAE -> BUS

BUS = 255 / FFh

D1 | IRAP : BUS -> RAP

RAP = 255 / FFh

Takt7:

S3 | OA : A -> BUS

BUS = 8 / 8h

D3 | IRBP : BUS -> RBP

RBP = 8 / 8h

C1 | CWC : Rozpoczęcie CWC

PAO[255] = 0xFC11 -zmiana-> PAO[255] = 0x0008

RAPS = 59 / 3Bh

MAKRO

=====59=====

Takt0: RBPS=00000E000000h

Takt7:

C1 | END : (Cykl 36) Koniec mikroprogramu (09:24.30)

RAPS = 0 / 0h

09:29.24

=====Stop symulacji=====

Ocena: 5 Błędy: 0

戇恂

Zrzut ekranu z obrazem ze stanem końcowym LabZSK:

The screenshot displays the 'Komputer LabZSK - sterowanie mikroprogramowe' interface. The main window is divided into several sections:

- Top Bar:** Includes tabs for 'Symulacja', 'Mikroprogramy', 'Pamięć operacyjna', and 'Info'. The 'Info' tab is active.
- Instruction Table:** A table with columns: Adres, S1, D1, S2, D2, S3, D3, C1, C2, Test, ALU, NA. The instruction at address 59 is highlighted as 'END'.
- Control Panel:** Contains buttons for 'MAKRO' and 'MIKRO', a 'Stop' button, and a 'Pokaż log' button.
- Registers and ALU:** A central area showing the state of various registers and the ALU. The ALU result is 310Eh (12558). Other registers shown include LK (7h), A (8h), MQ (0h), X (FFh), RAP (FFh), LALU (0h), RALU (0h), RBP (8h), BUS (0h), RR (3107h), LR (ACh), RI (310Eh), RBPS (00000E000000h), RAPS (0h), and RAE (FFh).
- Flags:** A section showing the status of various flags: ZNAK (0), XRO (0), OFF (0), MAV (0), IA (1), and INT (0).
- Memory Table:** A table on the right showing memory addresses, contents, and hex values. The address 255 contains the value 0000000000001000.
- Status Bar:** At the bottom, it shows the current time and date: 09:27, 2023-11-14.

Krótkie uzasadnienie końcowej zawartości: LR, RAPS, RAE na koniec mikroprogramu pobrania rozkazu dla każdego wykonanego rozkazu:

Rozkaz 1 – SUB 001 7

LR:

D2 | NSI : LR+1 -> LR

$$LR = 168 / A8h$$

Wartość komórki LR wzrasta o jeden na wskutek mikrooperacji NSI i na koniec pobierania rozkazu wynosi 168.

RAPS:

C2 | OPC : OP albo AOP+32 -> RAPS

$$RAPS = 2 / 2h$$

Mikrooperacja OPC przekazuje sterowanie do odpowiedniej komórki Pamięci Stałej. Ponieważ w RR znajduje się obecnie rozkaz zwykły, to nowa wartość RAPS będzie równa wartości 2 OP aktualnie wykonywanego rozkazu.

RAE:

C2 | CEA : Oblicz adres efektywny

$$L = 7 / 7h$$

$$R = 0 / 0h$$

$$SUMA = 7 / 7h$$

$$XRO = 0$$

S2 | IRAE : SUMA -> RAE

$$RAE = 7 / 7h$$

Wprowadzono do RAE wartość komórki SUMA, która jest wynikiem dodawania DA aktualnego rozkazu oraz 0, co wynika z adresowania rozkazu.

S1 | ORAE : RAE -> BUS

$$BUS = 7 / 7h$$

D1 | IRAP : BUS -> RAP

$$RAP = 7 / 7h$$

C1 | RRC : Rozpoczęcie RRC

$$RBP = 255 / FFh$$

Przesłano magistralą wartość RAE do RAP i wykonano mikrorozkaz RRC, który uzupełnił RBP wartością PAO[RAP] równą 255.

S3 | ORBP : RBP -> BUS

BUS = 255 / FFh

D3 | IX : BUS -> X

X = 255 / FFh

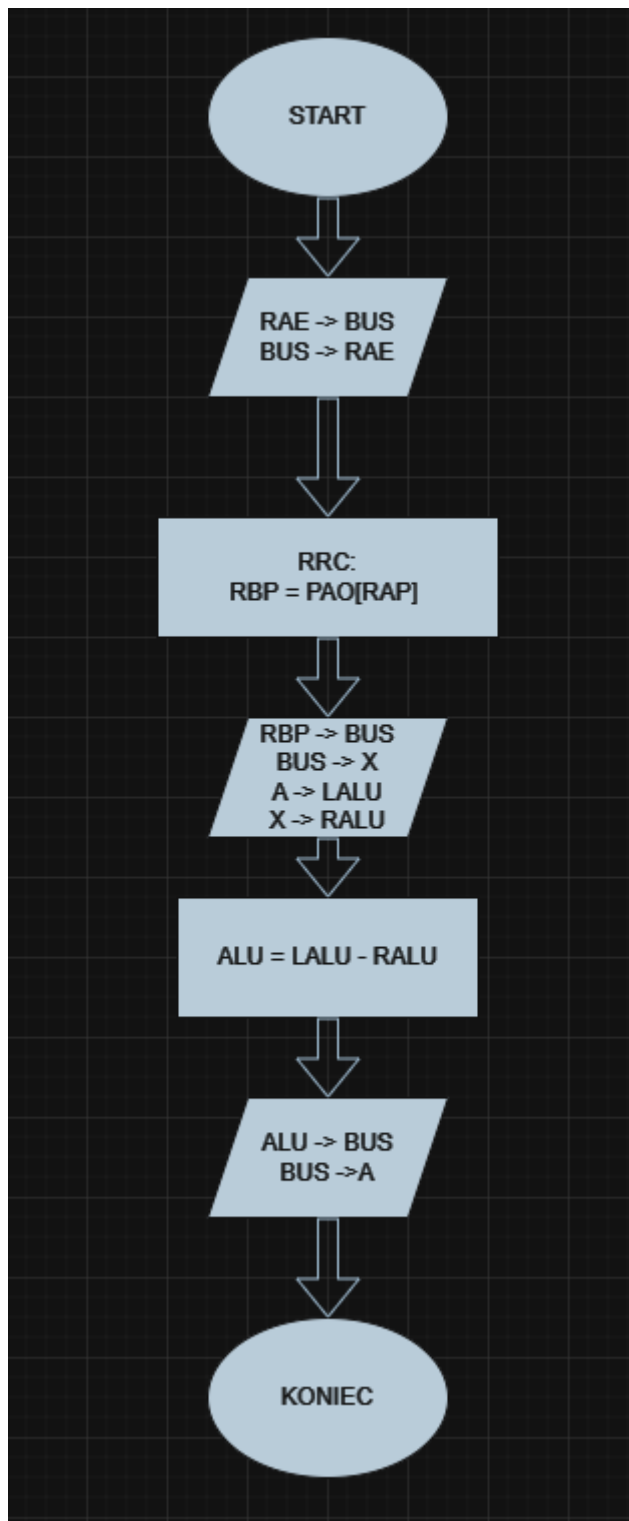
S2 | OX : X -> BUS

BUS = 255 / FFh

D2 | IBI : BUS -> RAE

RAE = 255 / FFh

Przesłano wartość 255 z RBP magistralą do komórki X, następnie z X przez magistralę do RAE.



Wartość 255 z RAE przysyłana jest przez magistrale do RAP, następnie mikrorozkaz RRC uzupełnia komórkę RBP wartością -1007 PAO[RAP]. Potem wartość -1007 z RBP przesyłana jest magistralą do X. Wartość (-1000) z komórki A przechodzi do komórki LALU, a wartość (-1007) z komórki X do komórki RALU. Mikrorozkaz SUB uzupełnia komórkę ALU różnicą wartości z LALU i RALU, czyli $ALU = (-1000) - (-1007) = 7$. ZNAK = 0 ponieważ wartość ALU nie jest ujemna i OFF = 0 ponieważ nie wystąpił nadmiar. Na koniec rozkazu przesyłamy

magistralą wartość 7 z ALU do A i ustawiamy RAPS na 0.

Rozkaz 2 – CMA 7

LR:

D2 | NSI : LR+1 -> LR

LR = 169 / A9h

Wartość komórki LR wzrasta o jeden na wskutek mikrooperacji NSI i na koniec pobierania rozkazu wynosi 169.

RAPS:

C2 | OPC : OP albo AOP+32 -> RAPS

RAPS = 33 / 21h

Mikrooperacja OPC przekazuje sterowanie do odpowiedniej komórki Pamięci Stałej. Ponieważ w RR znajduje się obecnie rozkaz rozszerzony, to nowa wartość RAPS będzie równa wartości 1 pola AOP aktualnie wykonywanego rozkazu plus 32, co daje 33.

RAE:

C2 | CEA : Oblicz adres efektywny

L = 7 / 7h

R = 0 / 0h

SUMA = 7 / 7h

XRO = 0

S2 | IRAE : SUMA -> RAE

RAE = 7 / 7h

Wprowadzono do RAE wartość komórki SUMA, która jest wynikiem dodawania N aktualnego rozkazu oraz 0, ponieważ rozkaz jest typu rozszerzonego.

S1 | ORAE : RAE -> BUS

BUS = 7 / 7h

D1 | IRAP : BUS -> RAP

RAP = 7 / 7h

C1 | RRC : Rozpoczęcie RRC

RBP = 255 / FFh

Przesłano magistralą wartość RAE do RAP i wykonano mikrorozkaz RRC, który uzupełnił RBP wartością PAO[RAP] równą 255.

S3 | ORBP : RBP -> BUS

BUS = 255 / FFh

D3 | IX : BUS -> X

X = 255 / FFh

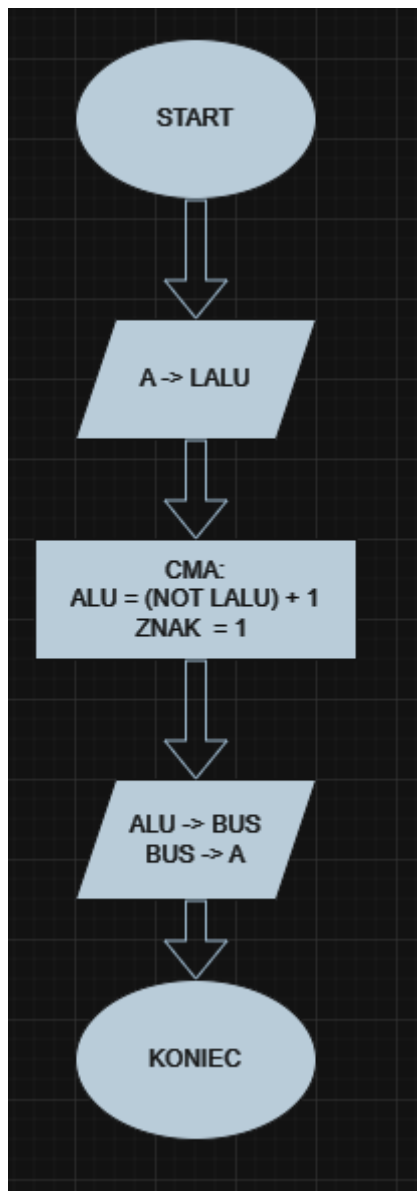
S2 | OX : X -> BUS

BUS = 255 / FFh

D2 | IBI : BUS -> RAE

RAE = 255 / FFh

Przesłano wartość 255 z RBP magistralą do komórki X, następnie z X przez magistralę do RAE.



Rozkaz CMA zaczyna działanie od przesłanie wartości 7 z komórki A do LALU. Mikrorozkaz CMA przepisuje wartość 7 ze zmienionym znakiem do komórki ALU czyli wartość (-7). ZNAK = 1 ponieważ wartość ALU jest ujemna, OFF = 0 ponieważ nie wystąpił nadmiar. Na koniec rozkazu przesyłamy magistralą wartość (-7) z ALU do A i ustawiamy RAPS na 0.

Rozkaz 3 – LAI 8

LR:

D2 | NSI : LR+1 -> LR

LR = 170 / AAh

Wartość komórki LR wzrasta o jeden na skutek mikrooperacji NSI i na koniec pobierania rozkazu wynosi 170.

RAPS:

C2 | OPC : OP albo AOP+32 -> RAPS

RAPS = 41 / 29h

Mikrooperacja OPC przekazuje sterowanie do odpowiedniej komórki Pamięci Stałej.

Ponieważ w RR znajduje się obecnie rozkaz rozszerzony, to nowa wartość RAPS będzie równa wartości 9 pola AOP aktualnie wykonywanego rozkazu plus 32, co daje 41.

RAE:

C2 | CEA : Oblicz adres efektywny

$L = 8 / 8h$

$R = 0 / 0h$

SUMA = 8 / 8h

S2 | IRAE : SUMA -> RAE

RAE = 8 / 8h

Wprowadzono do RAE wartość komórki SUMA, która jest wynikiem dodawania N aktualnego rozkazu oraz 0, ponieważ rozkaz jest typu rozszerzonego.

S1 | ORAE : RAE -> BUS

BUS = 8 / 8h

D1 | IRAP : BUS -> RAP

RAP = 8 / 8h

C1 | RRC : Rozpoczęcie RRC

RBP = 5007 / 138Fh

Przesłano magistralą wartość RAE do RAP i wykonano mikrorozkaz RRC, który uzupełnił RBP wartością PAO[RAP] równą 5007.

S3 | ORBP : RBP -> BUS

BUS = 5007 / 138Fh

D3 | IX : BUS -> X

$X = 5007 / 138Fh$

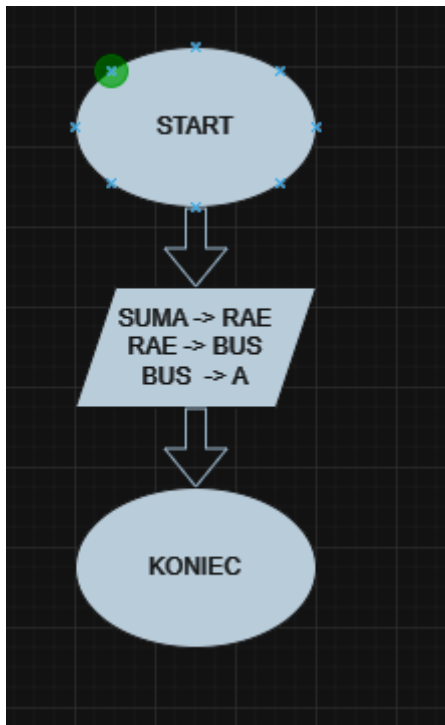
S2 | OX : X -> BUS

BUS = 5007 / 138Fh

D2 | IBI : BUS -> RAE

RAE = 5007 / 138Fh

Przesłano wartość 5007 z RBP magistralą do komórki X, następnie z X przez magistralę do RAE.



Mikrorozkaz IRAE przesyła wartość 8 równą N aktualnego rozkazu spod komórki SUMA do RAE. Wartość 8 z RAE jest przesyłana magistralą do A. Na koniec rozkazu RAPS ustawiany jest na 0.

Rozkaz 4 – ADX 010 1

LR:

D2 | NSI : LR+1 -> LR

LR = 171 / ABh

Wartość komórki LR wzrasta o jeden na skutek mikrooperacji NSI i na koniec pobierania rozkazu wynosi 171.

RAPS:

TEST | TIND : Adresowanie pośrednie

RAPS = 13 / Dh

RAPS został ustawiony na wartość 13 z pola OP aktualnego rozkazu, ponieważ rozkaz nie jest rozkazem rozszerzonym ani nie wykorzystuje adresowania pośredniego (I=0).

RAE:

C2 | CEA : Oblicz adres efektywny

$$L = 1 / 1h$$

$$R = 170 / AAh$$

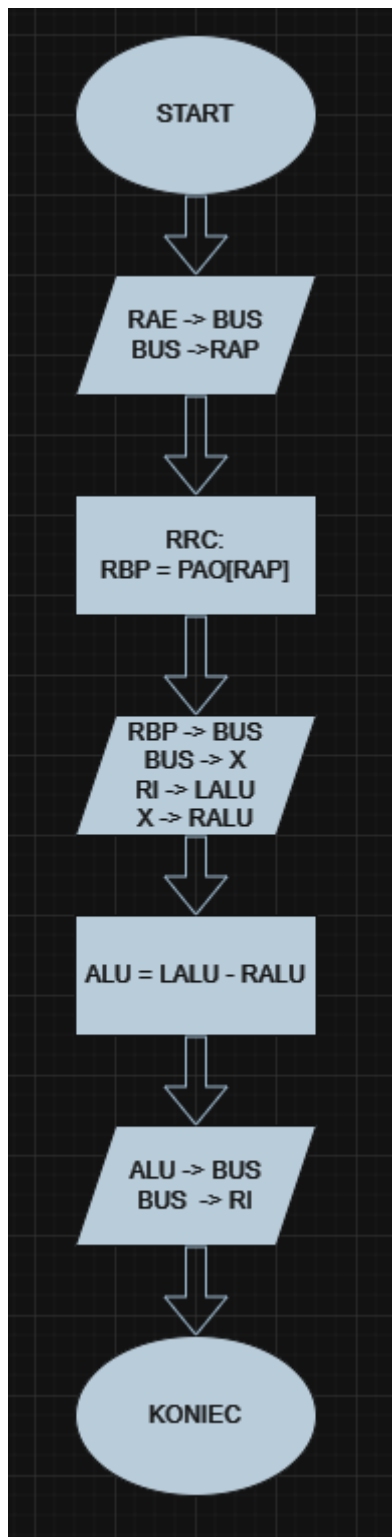
$$SUMA = 171 / ABh$$

$$XRO = 0$$

$$S2 \mid IRAE : SUMA \rightarrow RAE$$

$$RAE = 171 / ABh$$

Wprowadzono do RAE wartość komórki SUMA, która jest wynikiem dodawania DA aktualnego rozkazu oraz wartości komórki LR równej 170, co wynika z metody adresacji rozkazu.



Rozkaz ADX rozpoczyna się od przesłania magistralą wartości 171 spod RAE do RAP. Mikrorozkaz RRC uzupełnia komórkę RBP wartością 12551 PAO[RAP]. Wartość z RBP przesyłana jest magistralą do X. Wartość 7 spod komórki RI przechodzi do LALU, a wartość 12551 spod X do RALU. Mikrorozkaz ADD uzupełnia ALU sumą LALU i RALU, czyli 12558. ZNAK = 0 ponieważ wartość ALU jest nieujemna, OFF = 0 ponieważ nie wystąpił nadmiar. Na koniec rozkazu wartość 12558 spod ALU przesyłana jest magistralą do RI i następnie RAPS ustawiony zostaje na 0.

Rozkaz 5 – STA 001 7

LR:

D2 | NSI : LR+1 -> LR

LR = 172 / Ach

Wartość komórki LR wzrasta o jeden na wskutek mikrooperacji NSI i na koniec pobierania rozkazu wynosi 172.

RAPS:

C2 | OPC : OP albo AOP+32 -> RAPS

RAPS = 6 / 6h

Mikrooperacja OPC przekazuje sterowanie do odpowiedniej komórki Pamięci Stałej. Ponieważ w RR znajduje się obecnie rozkaz zwykły, to nowa wartość RAPS będzie równa wartości 6 OP aktualnie wykonywanego rozkazu.

RAE:

C2 | CEA : Oblicz adres efektywny

L = 7 / 7h

R = 0 / 0h

SUMA = 7 / 7h

XRO = 0

S2 | IRAE : SUMA -> RAE

RAE = 7 / 7h

Wprowadzono do RAE wartość komórki SUMA, która jest wynikiem dodawania DA aktualnego rozkazu oraz 0, co wynika z metody adresacji rozkazu.

S1 | ORAE : RAE -> BUS

BUS = 7 / 7h

D1 | IRAP : BUS -> RAP

RAP = 7 / 7h

C1 | RRC : Rozpoczęcie RRC

RBP = 255 / FFh

Przesłano magistralą wartość RAE do RAP i wykonano mikrorozkaz RRC, który uzupełnił RBP wartością PAO[RAP] równą 255.

S3 | ORBP : RBP -> BUS

BUS = 255 / FFh

D3 | IX : BUS -> X

X = 255 / FFh

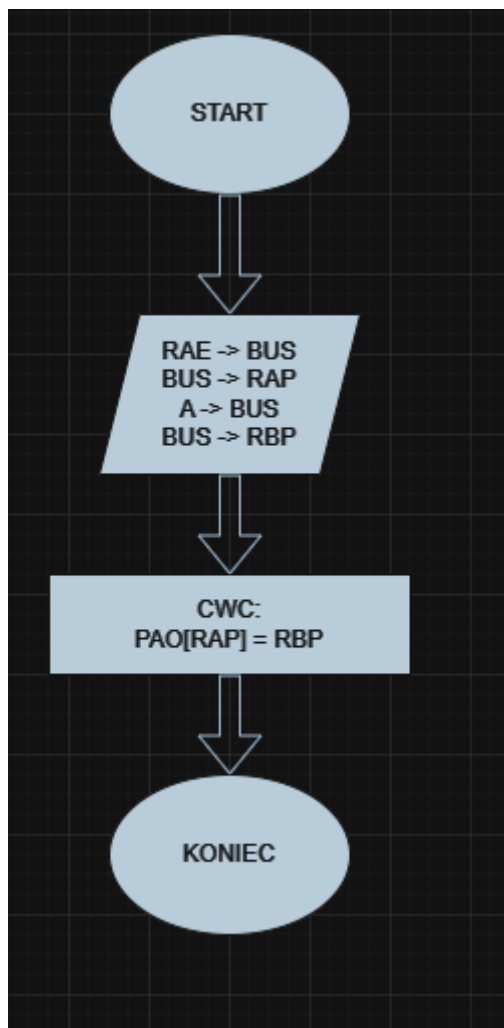
S2 | OX : X -> BUS

BUS = 255 / FFh

D2 | IBI : BUS -> RAE

RAE = 255 / FFh

Przesłano wartość 255 z RBP magistralą do komórki X, następnie z X przez magistralę do RAE.



Wartość 255 RAE przesyłana jest magistralą do RAP, a wartość 8 A do RBP. Mikrorozkaz CWC zmienia aktualne PAO[255] = 0xFC11 na PAO[255] = 0x0008 (PAO[RBP]). RAPS zostaje ustawiony na 0.