Architektura i organizacja komputerów

Sprawozdanie z laboratorium nr 4

Temat zajęć: Mikroprogramy rozkazów logicznych i skoków

Borkowski Kamil WCY22IY1S1

Data wykonania: 2023.11.21

Treść zadania:

Lab4\_IY1

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

|  |  |
| --- | --- |
| Rejestry |  |
| A | 32100+nr |
| LR | nr |
| RI | 100 |
| MQ | 111 |
|  |  |
| PAO |  |
| **Adres** | **Zawartość** |
| 0 | ABABh |
|  |  |
| LR | ADD 010 0 |
| LR+1 | ARA 3 |
| LR+2 | BAN 000 LR+10 |
| LR+3 | EOR 110 1 |
|  |  |
| LR+10 | LNG 000 nr |
| LR+11 | UNB 100 nr |
|  |  |
| 100+nr | UNB 000 LR |
|  |  |

gdzie nr = numer studentki/ studenta na liście grupy.

Pozostałe komórki PAO są wyzerowane.

Stopień trudności zadania:

* Na dostatecznie – poprawnie pobrać i wykonać 3 rozkazy, począwszy od PAO[LR].
* Na dobrze  – poprawnie pobrać i wykonać 4 rozkazy, począwszy od PAO[LR].
* Na bardzo dobrze – poprawnie pobrać i wykonać 5 rozkazów, począwszy od PAO[LR].

 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 poprzednich i dzisiejszych** zajęć (bez mnożenia i dzielenia oraz pozostałych z zestawu: MUL, DIV, SIO, LIO, BDN, CND, ENI, LDS),

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.

Niepoprawne (niezgodne z definicją z listy rozkazów) działanie któregokolwiek rozkazu z grupy na dst oznacza po wykonaniu innych wymagań ocenę ndst.

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

16 Test \_\_\_UNB Zawsze pozytywny

NA \_\_\_72

17 Test \_\_\_UNB Zawsze pozytywny

NA \_\_\_74

18 Test \_\_\_UNB Zawsze pozytywny

NA \_\_\_76

19 Test \_\_\_UNB Zawsze pozytywny

NA \_\_\_78

20 Test \_\_\_UNB Zawsze pozytywny

NA \_\_\_80

21 Test \_\_\_UNB Zawsze pozytywny

NA \_\_\_82

22 Test \_\_\_UNB Zawsze pozytywny

NA \_\_\_86

23 Test \_\_\_UNB Zawsze pozytywny

NA \_\_\_88

24 Test \_\_\_UNB Zawsze pozytywny

NA \_\_\_90

25 Test \_\_\_UNB Zawsze pozytywny

NA \_\_\_94

26 Test \_\_\_UNB Zawsze pozytywny

NA \_\_\_96

27 Test \_\_\_UNB Zawsze pozytywny

NA \_\_\_100

28 Test \_\_\_UNB Zawsze pozytywny

NA \_\_\_102

29 Test \_\_\_UNB Zawsze pozytywny

NA \_\_\_104

32 Test \_\_\_UNB Zawsze pozytywny

NA \_\_\_130

33 Test \_\_\_UNB Zawsze pozytywny

NA \_\_\_106

34 Test \_\_\_UNB Zawsze pozytywny

NA \_\_\_108

35 Test \_\_\_UNB Zawsze pozytywny

NA \_\_\_112

36 Test \_\_\_UNB Zawsze pozytywny

NA \_\_\_114

37 Test \_\_\_UNB Zawsze pozytywny

NA \_\_\_116

38 Test \_\_\_UNB Zawsze pozytywny

NA \_\_\_118

39 Test \_\_\_UNB Zawsze pozytywny

NA \_\_\_120

40 Test \_\_\_UNB Zawsze pozytywny

NA \_\_\_122

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

72 S3 \_\_\_ORAE RAE -> BUS

D3 \_\_\_ILR BUS -> LR

C1 \_\_\_END Koniec mikroprogramu

74 Test \_\_\_TAO OFF = 0

NA \_\_\_110

75 Test \_\_\_UNB Zawsze pozytywny

NA \_\_\_16

76 Test \_\_\_TXP RI <= 0

NA \_\_\_110

77 Test \_\_\_UNB Zawsze pozytywny

NA \_\_\_16

78 Test \_\_\_TXZ BXZ i RI != 0 || TLD i RI = 0

NA \_\_\_110

79 Test \_\_\_UNB Zawsze pozytywny

NA \_\_\_16

80 Test \_\_\_TXS RI >= 0

NA \_\_\_110

81 Test \_\_\_UNB Zawsze pozytywny

NA \_\_\_16

82 Test \_\_\_TXP RI <= 0

NA \_\_\_110

83 C2 \_\_\_DRI RI = RI-1

Test \_\_\_UNB Zawsze pozytywny

NA \_\_\_16

86 Test \_\_\_TAP A <= 0

NA \_\_\_110

87 Test \_\_\_UNB Zawsze pozytywny

NA \_\_\_16

88 Test \_\_\_TAZ A = 0

NA \_\_\_16

89 C1 \_\_\_END Koniec mikroprogramu

90 Test \_\_\_TAS A >= 0

NA \_\_\_110

91 Test \_\_\_UNB Zawsze pozytywny

NA \_\_\_16

94 S1 \_\_\_ORAE RAE -> BUS

D1 \_\_\_IRAP BUS -> RAP

S3 \_\_\_ORBP RBP -> BUS

D3 \_\_\_IX BUS -> X

C1 \_\_\_RRC Rozpoczęcie RRC

95 S1 \_\_\_IALU A -> LALU

D1 \_\_\_OXE X -> RALU

S2 \_\_\_OBE ALU -> BUS

D2 \_\_\_IA BUS -> A

C1 \_\_\_END Koniec mikroprogramu

ALU \_\_\_OR ALU = LALU OR RALU

96 S1 \_\_\_ORAE RAE -> BUS

D1 \_\_\_IRAP BUS -> RAP

S3 \_\_\_ORBP RBP -> BUS

D3 \_\_\_IX BUS -> X

C1 \_\_\_RRC Rozpoczęcie RRC

97 S1 \_\_\_IALU A -> LALU

D1 \_\_\_OXE X -> RALU

S2 \_\_\_OBE ALU -> BUS

D2 \_\_\_IA BUS -> A

C1 \_\_\_END Koniec mikroprogramu

ALU \_\_\_AND ALU = LALU AND RALU

100 S1 \_\_\_IALU A -> LALU

S2 \_\_\_OBE ALU -> BUS

D2 \_\_\_IA BUS -> A

C1 \_\_\_END Koniec mikroprogramu

ALU \_\_\_NOTL ALU = NOT LALU

102 S1 \_\_\_ORAE RAE -> BUS

D1 \_\_\_IRAP BUS -> RAP

S3 \_\_\_ORBP RBP -> BUS

D3 \_\_\_IX BUS -> X

C1 \_\_\_RRC Rozpoczęcie RRC

103 S1 \_\_\_IALU A -> LALU

D1 \_\_\_OXE X -> RALU

S2 \_\_\_OBE ALU -> BUS

D2 \_\_\_IA BUS -> A

C1 \_\_\_END Koniec mikroprogramu

ALU \_\_\_EOR ALU = LALU XOR RALU

104 S1 \_\_\_ORAE RAE -> BUS

D1 \_\_\_IRAP BUS -> RAP

S3 \_\_\_OLR LR -> BUS

D3 \_\_\_IRBP BUS -> RBP

C1 \_\_\_CWC Rozpoczęcie CWC

105 S2 \_\_\_ORAE RAE -> BUS

D2 \_\_\_ILR BUS -> LR

D3 \_\_\_NSI LR+1 -> LR

C1 \_\_\_END Koniec mikroprogramu

106 S1 \_\_\_IALU A -> LALU

S2 \_\_\_OBE ALU -> BUS

D2 \_\_\_IA BUS -> A

C1 \_\_\_END Koniec mikroprogramu

ALU \_\_\_CMA ALU = (NOT LALU)+1

108 D2 \_\_\_ALA arytmetyczne A w lewo

C1 \_\_\_SHT Operacja przesunięcia

C2 \_\_\_DLK LK = [LK]-1

Test \_\_\_TLK SHT, LK=0 || !SHT, LK!=0

NA \_\_\_110

109 Test \_\_\_UNB Zawsze pozytywny

NA \_\_\_108

110 C1 \_\_\_END Koniec mikroprogramu

112 D2 \_\_\_ARA arytmetyczne A w prawo

C1 \_\_\_SHT Operacja przesunięcia

C2 \_\_\_DLK LK = [LK]-1

Test \_\_\_TLK SHT, LK=0 || !SHT, LK!=0

NA \_\_\_110

113 Test \_\_\_UNB Zawsze pozytywny

NA \_\_\_112

114 D2 \_\_\_LRQ logiczne A i MQ w prawo

C1 \_\_\_SHT Operacja przesunięcia

C2 \_\_\_DLK LK = [LK]-1

Test \_\_\_TLK SHT, LK=0 || !SHT, LK!=0

NA \_\_\_110

115 Test \_\_\_UNB Zawsze pozytywny

NA \_\_\_114

116 D2 \_\_\_LLQ logiczne A i MQ w lewo

C1 \_\_\_SHT Operacja przesunięcia

C2 \_\_\_DLK LK = [LK]-1

Test \_\_\_TLK SHT, LK=0 || !SHT, LK!=0

NA \_\_\_110

117 Test \_\_\_UNB Zawsze pozytywny

NA \_\_\_116

118 D2 \_\_\_LLA logiczne A w lewo

C1 \_\_\_SHT Operacja przesunięcia

C2 \_\_\_DLK LK = [LK]-1

Test \_\_\_TLK SHT, LK=0 || !SHT, LK!=0

NA \_\_\_110

119 Test \_\_\_UNB Zawsze pozytywny

NA \_\_\_118

120 D2 \_\_\_LRA logiczne A w prawo

C1 \_\_\_SHT Operacja przesunięcia

C2 \_\_\_DLK LK = [LK]-1

Test \_\_\_TLK SHT, LK=0 || !SHT, LK!=0

NA \_\_\_110

121 Test \_\_\_UNB Zawsze pozytywny

NA \_\_\_120

122 D2 \_\_\_LCA cykliczne A w lewo

C1 \_\_\_SHT Operacja przesunięcia

C2 \_\_\_DLK LK = [LK]-1

Test \_\_\_TLK SHT, LK=0 || !SHT, LK!=0

NA \_\_\_110

123 Test \_\_\_UNB Zawsze pozytywny

NA \_\_\_122

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

130 S3 \_\_\_OLR LR -> BUS

D3 \_\_\_IX BUS -> X

131 S1 \_\_\_OXE X -> RALU

S2 \_\_\_OBE ALU -> BUS

D2 \_\_\_ILR BUS -> LR

C1 \_\_\_END Koniec mikroprogramu

ALU \_\_\_DECR ALU = RALU - 1

Wydruk zawartości PAO:

0 1010101110101011b ABABh -21589

7 0000101000000000b 0A00h OP=1 XSI=010 DA=0

8 0000000110000011b 0183h AOP=3 N=3

9 1100000000010001b C011h OP=24 XSI=000 DA=17

10 1110011000000001b E601h OP=28 XSI=110 DA=1

17 1101100000000111b D807h OP=27 XSI=000 DA=7

18 1000010000000111b 8407h OP=16 XSI=100 DA=7

107 1000000000000111b 8007h OP=16 XSI=000 DA=7

Wydruk logu z wykonania ćwiczenia:

Start symulatora 2023-11-21 12:28:30

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======

12:34.47

======Zawartość rejestrów======

LK = 0h 0

A = 7D6Bh 32107

MQ = 6Fh 111

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 = 7h 7

RI = 64h 100

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 = 7 / 7h

D1 | IRAP : BUS -> RAP

RAP = 7 / 7h

C1 | RRC : Rozpoczęcie RRC

RBP = 2560 / A00h

Takt7:

S3 | ORBP : RBP -> BUS

BUS = 2560 / A00h

D3 | IRR : BUS -> RR

RR = 2560 / A00h

RAPS = 49 / 31h

MAKRO

===============49================

Takt0: RBPS=68C801830032h

Takt1:

S1 | ORR : RR -> BUS

BUS = 2560 / A00h

D1 | ILK : BUS -> LK

LK = 0 / 0h

C2 | CEA : Oblicz adres efektywny

L = 0 / 0h

R = 7 / 7h

SUMA = 7 / 7h

XRO = 0

Takt6:

S2 | IRAE : SUMA -> RAE

RAE = 7 / 7h

D2 | NSI : LR+1 -> LR

LR = 8 / 8h

Takt7:

TEST | TIND : Adresowanie pośrednie

RAPS = 1 / 1h

MAKRO

===============1================

Takt0: RBPS=000000010034h

Takt7:

TEST | UNB : Zawsze pozytywny

RAPS = 52 / 34h

MAKRO

===============52================

Takt0: RBPS=900624000000h

Takt1:

S1 | ORAE : RAE -> BUS

BUS = 7 / 7h

D1 | IRAP : BUS -> RAP

RAP = 7 / 7h

C1 | RRC : Rozpoczęcie RRC

RBP = 2560 / A00h

Takt7:

S3 | ORBP : RBP -> BUS

BUS = 2560 / A00h

D3 | IX : BUS -> X

X = 2560 / A00h

RAPS = 53 / 35h

MAKRO

===============53================

Takt0: RBPS=BC300E000100h

Takt1:

S1 | IALU : A -> LALU

LALU = 32107 / 7D6Bh

D1 | OXE : X -> RALU

RALU = 2560 / A00h

Takt2:

ALU | ADD : ALU = LALU + RALU

Błąd(1): ALU = 3466 / D8Ah (Poprawna ALU = -30869 / 876Bh)

ZNAK = 1, OFF = 1

Takt6:

S2 | OBE : ALU -> BUS

BUS = -30869 / 876Bh

D2 | IA : BUS -> A

A = -30869 / 876Bh

Takt7:

C1 | END : (Cykl 6) Koniec mikroprogramu (12:41.29)

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 = 8 / 8h

D1 | IRAP : BUS -> RAP

RAP = 8 / 8h

C1 | RRC : Rozpoczęcie RRC

RBP = 387 / 183h

Takt7:

S3 | ORBP : RBP -> BUS

BUS = 387 / 183h

D3 | IRR : BUS -> RR

RR = 387 / 183h

RAPS = 49 / 31h

MAKRO

===============49================

Takt0: RBPS=68C801830032h

Takt1:

S1 | ORR : RR -> BUS

BUS = 387 / 183h

D1 | ILK : BUS -> LK

LK = 3 / 3h

C2 | CEA : Oblicz adres efektywny

L = 3 / 3h

R = 0 / 0h

SUMA = 3 / 3h

Takt6:

S2 | IRAE : SUMA -> RAE

RAE = 3 / 3h

D2 | NSI : LR+1 -> LR

LR = 9 / 9h

Takt7:

TEST | TIND : Adresowanie pośrednie

RAPS = 50 / 32h

MAKRO

===============50================

Takt0: RBPS=900624000000h

Takt1:

S1 | ORAE : RAE -> BUS

BUS = 3 / 3h

D1 | IRAP : BUS -> RAP

RAP = 3 / 3h

C1 | RRC : Rozpoczęcie RRC

RBP = 0 / 0h

Takt7:

S3 | ORBP : RBP -> BUS

BUS = 0 / 0h

D3 | IX : BUS -> X

X = 0 / 0h

RAPS = 51 / 33h

MAKRO

===============51================

Takt0: RBPS=03A801600000h

Takt6:

S2 | OX : X -> BUS

BUS = 0 / 0h

D2 | IBI : BUS -> RAE

RAE = 0 / 0h

Takt7:

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

RAPS = 35 / 23h

MAKRO

===============35================

Takt0: RBPS=000000010070h

Takt7:

TEST | UNB : Zawsze pozytywny

RAPS = 112 / 70h

MAKRO

===============112================

Takt0: RBPS=00680A27006Eh

Takt1:

D2 | ARA : arytmetyczne A w prawo

C1 | SHT : Operacja przesunięcia

A = -15435 / C3B5h

Takt6:

C2 | DLK : LK = [LK]-1

LK = 2 / 2h

Takt7:

TEST | TLK : SHT, LK=0 || !SHT, LK!=0

RAPS = 113 / 71h

MAKRO

===============113================

Takt0: RBPS=000000010070h

Takt7:

TEST | UNB : Zawsze pozytywny

RAPS = 112 / 70h

MAKRO

===============112================

Takt0: RBPS=00680A27006Eh

Takt1:

D2 | ARA : arytmetyczne A w prawo

C1 | SHT : Operacja przesunięcia

A = -7718 / E1DAh

Takt6:

C2 | DLK : LK = [LK]-1

LK = 1 / 1h

Takt7:

TEST | TLK : SHT, LK=0 || !SHT, LK!=0

RAPS = 113 / 71h

MAKRO

===============113================

Takt0: RBPS=000000010070h

Takt7:

TEST | UNB : Zawsze pozytywny

RAPS = 112 / 70h

MAKRO

===============112================

Takt0: RBPS=00680A27006Eh

Takt1:

D2 | ARA : arytmetyczne A w prawo

C1 | SHT : Operacja przesunięcia

A = -3859 / F0EDh

Takt6:

C2 | DLK : LK = [LK]-1

LK = 0 / 0h

Takt7:

TEST | TLK : SHT, LK=0 || !SHT, LK!=0

RAPS = 110 / 6Eh

MAKRO

===============110================

Takt0: RBPS=00000E000000h

Takt7:

C1 | END : (Cykl 18) Koniec mikroprogramu (12:54.02)

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 = 9 / 9h

D1 | IRAP : BUS -> RAP

RAP = 9 / 9h

C1 | RRC : Rozpoczęcie RRC

RBP = -16367 / C011h

Takt7:

S3 | ORBP : RBP -> BUS

BUS = -16367 / C011h

D3 | IRR : BUS -> RR

RR = -16367 / C011h

RAPS = 49 / 31h

MAKRO

===============49================

Takt0: RBPS=68C801830032h

Takt1:

S1 | ORR : RR -> BUS

BUS = -16367 / C011h

D1 | ILK : BUS -> LK

LK = 17 / 11h

C2 | CEA : Oblicz adres efektywny

L = 17 / 11h

R = 0 / 0h

SUMA = 17 / 11h

XRO = 0

Takt6:

S2 | IRAE : SUMA -> RAE

RAE = 17 / 11h

D2 | NSI : LR+1 -> LR

LR = 10 / Ah

Takt7:

TEST | TIND : Adresowanie pośrednie

RAPS = 24 / 18h

MAKRO

===============24================

Takt0: RBPS=00000001005Ah

Takt7:

TEST | UNB : Zawsze pozytywny

RAPS = 90 / 5Ah

MAKRO

===============90================

Takt0: RBPS=00000004006Eh

Takt7:

TEST | TAS : A >= 0

RAPS = 91 / 5Bh

MAKRO

===============91================

Takt0: RBPS=000000010010h

Takt7:

TEST | UNB : Zawsze pozytywny

RAPS = 16 / 10h

MAKRO

===============16================

Takt0: RBPS=000000010048h

Takt7:

TEST | UNB : Zawsze pozytywny

RAPS = 72 / 48h

MAKRO

===============72================

Takt0: RBPS=00041E000000h

Takt7:

S3 | ORAE : RAE -> BUS

BUS = 17 / 11h

D3 | ILR : BUS -> LR

LR = 17 / 11h

C1 | END : (Cykl 26) Koniec mikroprogramu (12:57.06)

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 = 17 / 11h

D1 | IRAP : BUS -> RAP

RAP = 17 / 11h

C1 | RRC : Rozpoczęcie RRC

RBP = -10233 / D807h

Takt7:

S3 | ORBP : RBP -> BUS

BUS = -10233 / D807h

D3 | IRR : BUS -> RR

RR = -10233 / D807h

RAPS = 49 / 31h

MAKRO

===============49================

Takt0: RBPS=68C801830032h

Takt1:

S1 | ORR : RR -> BUS

BUS = -10233 / D807h

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 = 18 / 12h

Takt7:

TEST | TIND : Adresowanie pośrednie

RAPS = 27 / 1Bh

MAKRO

===============27================

Takt0: RBPS=000000010064h

Takt7:

TEST | UNB : Zawsze pozytywny

RAPS = 100 / 64h

MAKRO

===============100================

Takt0: RBPS=A4300E000800h

Takt1:

S1 | IALU : A -> LALU

LALU = -3859 / F0EDh

Takt2:

ALU | NOTL : ALU = NOT LALU

ALU = 3858 / F12h

ZNAK = 0, OFF = 0

Takt6:

S2 | OBE : ALU -> BUS

BUS = 3858 / F12h

D2 | IA : BUS -> A

A = 3858 / F12h

Takt7:

C1 | END : (Cykl 31) Koniec mikroprogramu (13:00.47)

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 = 18 / 12h

D1 | IRAP : BUS -> RAP

RAP = 18 / 12h

C1 | RRC : Rozpoczęcie RRC

RBP = -31737 / 8407h

Takt7:

S3 | ORBP : RBP -> BUS

BUS = -31737 / 8407h

D3 | IRR : BUS -> RR

RR = -31737 / 8407h

RAPS = 49 / 31h

MAKRO

===============49================

Takt0: RBPS=68C801830032h

Takt1:

S1 | ORR : RR -> BUS

BUS = -31737 / 8407h

D1 | ILK : BUS -> LK

LK = 7 / 7h

C2 | CEA : Oblicz adres efektywny

L = 7 / 7h

R = 100 / 64h

SUMA = 107 / 6Bh

XRO = 0

Takt6:

S2 | IRAE : SUMA -> RAE

RAE = 107 / 6Bh

D2 | NSI : LR+1 -> LR

LR = 19 / 13h

Takt7:

TEST | TIND : Adresowanie pośrednie

RAPS = 16 / 10h

MAKRO

===============16================

Takt0: RBPS=000000010048h

Takt7:

TEST | UNB : Zawsze pozytywny

RAPS = 72 / 48h

MAKRO

===============72================

Takt0: RBPS=00041E000000h

Takt7:

S3 | ORAE : RAE -> BUS

BUS = 107 / 6Bh

D3 | ILR : BUS -> LR

LR = 107 / 6Bh

C1 | END : (Cykl 36) Koniec mikroprogramu (13:03.41)

RAPS = 0 / 0h

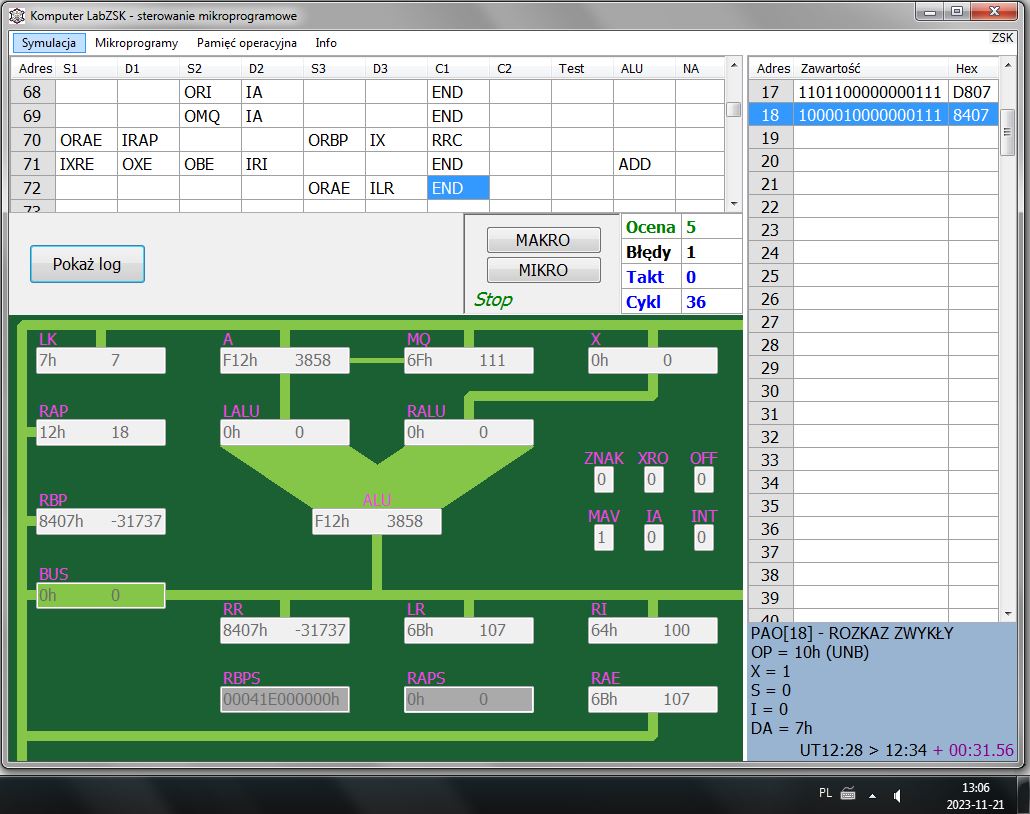
13:08.10

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

Ocena: 5 Błędy: 1

䵉ꄪ

Zrzut ekranu z obrazem ze stanem końcowym LabZSK:



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

Rozkaz 1 – ADD 010 0

**LR:**

D2 | NSI : LR+1 -> LR

LR = 8 / 8h

Wartość komórki LR wzrasta o jeden na wskutek mikrorozkazu NSI i na koniec pobierania rozkazu wynosi 8.

**RAPS:**

TEST | TIND : Adresowanie pośrednie

RAPS = 1 / 1h

RAPS został ustawiony na wartość 1 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 = 0 / 0h

R = 7 / 7h

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 wartość 7 komórki LR, co wynika z adresowania rozkazu.

Obraz zawierający tekst, zrzut ekranu, typografia, design

Opis wygenerowany automatycznie

Wartość 7 z RAE przysyłana jest przez magistrale do RAP, następnie mikrorozkaz RRC uzupełnia komórkę RBP wartością 2560 PAO[RAP]. Potem wartość 2560 z RBP przesyłana jest magistralą do X. Wartość 32107 z komórki A przechodzi do komórki LALU, a wartość 2560 z komórki X do komórki RALU. Mikrorozkaz ADD uzupełnia komórkę ALU sumą wartości z LALU i RALU, czyli ALU = 32107 + 2560 = -30869. Wynik nie jest zgodny z logicznym myśleniem przez nadmiar. ZNAK = 1 ponieważ wartość ALU jest ujemna i OFF = 1 ponieważ wystąpił nadmiar. Na koniec rozkazu przesyłamy magistralą wartość -30869 z ALU do A i ustawiamy RAPS na 0.

Rozkaz 2 – ARA 3

**LR:**

D2 | NSI : LR+1 -> LR

LR = 9 / 9h

Wartość komórki LR wzrasta o jeden na wskutek mikrorozkazu NSI i na koniec pobierania rozkazu wynosi 9.

**RAPS:**

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

RAPS = 35 / 23h

Mikrorozkaz OPC przekazuje sterowanie do odpowiedniej komórki Pamięci Stałej. Ponieważ obecny rozkaz jest rozszerzony, to nowa wartość RAPS będzie równa wartości 3 pola AOP aktualnie wykonywanego rozkazu plus 32, co daje 35.

**RAE:**

C2 | CEA : Oblicz adres efektywny

L = 3 / 3h

R = 0 / 0h

SUMA = 3 / 3h

S2 | IRAE : SUMA -> RAE

RAE = 3 / 3h

Wprowadzono do RAE wartość komórki SUMA, która jest wynikiem dodawania wartości 3 N aktualnego rozkazu oraz 0, ponieważ rozkaz jest typu rozszerzonego.

S1 | ORAE : RAE -> BUS

BUS = 3 / 3h

D1 | IRAP : BUS -> RAP

RAP = 3 / 3h

C1 | RRC : Rozpoczęcie RRC

RBP = 0 / 0h

Przesłano magistralą wartość RAE do RAP i wykonano mikrorozkaz RRC, który uzupełnił RBP wartością PAO[RAP] równą 0.

S3 | ORBP : RBP -> BUS

BUS = 0 / 0h

D3 | IX : BUS -> X

X = 0 / 0h

S2 | OX : X -> BUS

BUS = 0 / 0h

D2 | IBI : BUS -> RAE

RAE = 0 / 0h

Przesłano wartość 0 z RBP magistralą do komórki X, następnie z X przez magistralę do RAE.

Obraz zawierający tekst, zrzut ekranu, diagram, linia

Opis wygenerowany automatycznie

Rozkaz ARA przesuwa arytmetycznie A w prawo, więc komórka zmienia wartość A z 876Bh na C3B5h, następnie mikrorozkaz DLK dekrementuje wartość LK, potem mikrorozkaz TLK sprawdza czy wartość LK jest różna od zera, tak długo jak wartość LK nie równa się zero rozkaz się powtórzy, na początku rozpoczęcia rozkazu LK ma wartość 3, po pierwszym przesunięciu wartość 2. Wartość A zostanie jeszcze przesunięta arytmetycznie dwa razy, najpierw na wartość E1DAh, następnie na wartość F0EDh, gdy to się stanie wartość LK będzie równa 0 i rozkaz się zakończy.

Rozkaz 3 – BAN 000 17

**LR:**

D2 | NSI : LR+1 -> LR

LR = 10 / Ah

Wartość komórki LR wzrasta o jeden na wskutek mikrooperacji NSI i na koniec pobierania rozkazu wynosi 10.

**RAPS:**

TEST | TIND : Adresowanie pośrednie

RAPS = 24 / 18h

RAPS został ustawiony na wartość 24 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 = 17 / 11h

R = 0 / 0h

SUMA = 17 / 11h

XRO = 0

S2 | IRAE : SUMA -> RAE

RAE = 17 / 11h

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

Obraz zawierający tekst, zrzut ekranu, diagram, linia

Opis wygenerowany automatycznie

Mikrorozkaz TAS sprawdza czy wartość -3859 A jest większa od zera, jeżeli nie to przesyła wartość 17 RAE przez magistralę do LR i kończy rozkaz i ustawia RAPS na 0. Jeżeli wartość A jest nie większa od 0 to rozkaz nic nie robi poza ustawieniem RAPS na 0;

Rozkaz 4 – LNG 000 7

**LR:**

D2 | NSI : LR+1 -> LR

LR = 18 / 12h

Wartość komórki LR wzrasta o jeden na wskutek mikrorozkazu NSI i na koniec pobierania rozkazu wynosi 18 (Między pobraniami rozkazów wartość LR została zmieniona z 9 na 17 przez poprzedni rozkaz).

**RAPS:**

TEST | TIND : Adresowanie pośrednie

RAPS = 27 / 1Bh

RAPS został ustawiony na wartość 27 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 = 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 wartości 0, co wynika z metody adresacji rozkazu.

Obraz zawierający tekst, typografia, design

Opis wygenerowany automatycznie

Rozkaz LNG rozpoczyna się od przesłania wartości -3859 spod A do LALU. Mikrorozkaz NOTL uzupełnia komórkę ALU wartością 3858 (NOT LALU). Wartość z ALU przesyłana jest magistralą do A.

Rozkaz 5 – UNB 000 7

**LR:**

D2 | NSI : LR+1 -> LR

LR = 19 / 13h

Wartość komórki LR wzrasta o jeden na wskutek mikrorozkazu NSI i na koniec pobierania rozkazu wynosi 19.

**RAPS:**

TEST | TIND : Adresowanie pośrednie

RAPS = 16 / 10h

RAPS został ustawiony na wartość 27 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 = 7 / 7h

R = 100 / 64h

SUMA = 107 / 6Bh

XRO = 0

S2 | IRAE : SUMA -> RAE

RAE = 107 / 6Bh

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

Obraz zawierający tekst, zrzut ekranu, krąg, linia

Opis wygenerowany automatycznie

Wartość 107 RAE przesyłana jest magistralą do LR. RAPS zostaje ustawiony na 0.