CSE 331

Computer Organization

Fall 2020– 2021

HW4 Report

Caner KARAKAŞ

131044061

Shifter modül:

```
# time =  0, Input = 11111111111100001101111111111111, Result = 11111111110000110111111111111100
# time = 20, Input = 00001111000000011111000000000001, Result = 00111100000001111100000000000100
# time = 40, Input = 00000000000011111000000000001011, Result = 00000000001111000000000000101100
# time = 60, Input = 00000000000010000000000000000000, Result = 00000000001000000000000000000000
```

Register modül :

```
# time =  0,
# ReadReg1 = 00010,
# ReadReg2 =                              00110
# ReadData1 = 01010,
# ReadData2 =                               01010
# WriteReg1 = 01110,
# WriteReg2 = 01110,
# WriteData1 = 00000000000000000000000000001110,
# WriteData2 = 00000000000000000000000000001100,
# RegWrite_signal = 10
#
#
# time = 180,
# ReadReg1 = 00011,
# ReadReg2 =                              00111
# ReadData1 = 01011,
# ReadData2 =                               01010
# WriteReg1 = 01110,
# WriteReg2 = 01110,
# WriteData1 = 00000000000000000000000000001110,
# WriteData2 = 00000000000000000000000000001100,
# RegWrite_signal = 11
#
#
# time = 240,
# ReadReg1 = 00011,
# ReadReg2 =                              00111
# ReadData1 = 01011,
# ReadData2 =                               01010
# WriteReg1 = 01110,
# WriteReg2 = 01110,
# WriteData1 = 00000000000000000000000000001110,
# WriteData2 = 11111111111111111111111111111111,
# RegWrite_signal = 11
#
```

Multiplexer 2x1 bir bitlik modül:

```
# time= 0, i0=0, i1=0, select=0, output=0
# time=20, i0=0, i1=0, select=1, output=0
# time=40, i0=0, i1=1, select=0, output=0
# time=60, i0=0, i1=1, select=1, output=1
# time=80, i0=1, i1=0, select=0, output=1
# time=100, i0=1, i1=0, select=1, output=0
# time=120, i0=1, i1=1, select=0, output=1
# time=140, i0=1, i1=1, select=1, output=1
```

Multiplexer 2x1 otuz iki bitlik modül :

```
# time= 0, input0=00000000000000000000000000000011, input1=00000000000000000000000000000001, selectbit=1, result=00000000000000000000000000000001
# time=20, input0=00000000000000000000000000000010, input1=00100000000000000000000000000001, selectbit=0, result=00000000000000000000000000000010
# time=40, input0=01111111000001111111111111111111, input1=11111111100011111111111111111111, selectbit=0, result=01111111000001111111111111111111
# time=60, input0=01111111111100001111111111111111, input1=00000000000000000000000000000001, selectbit=1, result=00000000000000000000000000000001
# time=80, input0=00000000000000000000000000111111, input1=00000000000000000000000000111011, selectbit=0, result=00000000000000000000000000111111
```

Multiplexer 4x1 bir bitlik modül :

```
# num1 = 1, num2 = 0, num3 = 0, num4 = 0, s1 = 0, s0 = 0, result = 1
# num1 = 0, num2 = 1, num3 = 0, num4 = 0, s1 = 0, s0 = 1, result = 1
# num1 = 0, num2 = 0, num3 = 1, num4 = 0, s1 = 1, s0 = 0, result = 1
# num1 = 0, num2 = 0, num3 = 0, num4 = 1, s1 = 1, s0 = 1, result = 1
```

Multiplexer 8x1 bir bitlik modül :

```
# num1 = 1, num2 = 0, num3 = 0, num4 = 0,num5 = 0, num6 = 0, num7 = 0, num8 = 0, s2 = 0, s1 = 0, s0 = 0, result = 1
# num1 = 0, num2 = 1, num3 = 0, num4 = 0,num5 = 0, num6 = 0, num7 = 0, num8 = 0, s2 = 0, s1 = 0, s0 = 1, result = 1
# num1 = 0, num2 = 0, num3 = 1, num4 = 0,num5 = 0, num6 = 0, num7 = 0, num8 = 0, s2 = 0, s1 = 1, s0 = 0, result = 1
# num1 = 0, num2 = 0, num3 = 0, num4 = 1,num5 = 0, num6 = 0, num7 = 0, num8 = 0, s2 = 0, s1 = 1, s0 = 1, result = 1
# num1 = 0, num2 = 0, num3 = 0, num4 = 0,num5 = 1, num6 = 0, num7 = 0, num8 = 0, s2 = 1, s1 = 0, s0 = 0, result = 1
# num1 = 0, num2 = 0, num3 = 0, num4 = 0,num5 = 0, num6 = 1, num7 = 0, num8 = 0, s2 = 1, s1 = 0, s0 = 1, result = 1
# num1 = 0, num2 = 0, num3 = 0, num4 = 0,num5 = 0, num6 = 0, num7 = 1, num8 = 0, s2 = 1, s1 = 1, s0 = 0, result = 1
# num1 = 0, num2 = 0, num3 = 0, num4 = 0,num5 = 0, num6 = 0, num7 = 0, num8 = 1, s2 = 1, s1 = 1, s0 = 1, result = 1
```

Multiplexer 8x1 otuz iki bitlik modül :

```
VSIM 4> step -current
# num32 = 00000000000000000000000000000001, num2 = 00000000000000000000000000000000, num3 = 00000000000000000000000000000000, num4 = 00000000000000000000000000000000,num5 = 00000000000000000000000000000000, num6 = 00000000000000000000000000000000, num7 = 00000000000000000000000000000000, num8 = 00000000000000000000000000000000, s2 = 0, s1 = 0, s0 = 0, result = 00000000000000000000000000000001
# num32 = 00000000000000000000000000000000, num2 = 00000000000000000000000000000001, num3 = 00000000000000000000000000000000, num4 = 00000000000000000000000000000000,num5 = 00000000000000000000000000000000, num6 = 00000000000000000000000000000000, num7 = 00000000000000000000000000000000, num8 = 00000000000000000000000000000000, s2 = 0, s1 = 0, s0 = 1, result = 00000000000000000000000000000001
# num32 = 00000000000000000000000000000000, num2 = 00000000000000000000000000000000, num3 = 00000000000000000000000000000001, num4 = 00000000000000000000000000000000,num5 = 00000000000000000000000000000000, num6 = 00000000000000000000000000000000, num7 = 00000000000000000000000000000000, num8 = 00000000000000000000000000000000, s2 = 0, s1 = 1, s0 = 0, result = 00000000000000000000000000000001
# num32 = 00000000000000000000000000000000, num2 = 00000000000000000000000000000000, num3 = 00000000000000000000000000000000, num4 = 00000000000000000000000000000001,num5 = 00000000000000000000000000000000, num6 = 00000000000000000000000000000000, num7 = 00000000000000000000000000000000, num8 = 00000000000000000000000000000000, s2 = 0, s1 = 1, s0 = 1, result = 00000000000000000000000000000001
# num32 = 00000000000000000000000000000000, num2 = 00000000000000000000000000000000, num3 = 00000000000000000000000000000000, num4 = 00000000000000000000000000000000,num5 = 00000000000000000000000000000001, num6 = 00000000000000000000000000000000, num7 = 00000000000000000000000000000000, num8 = 00000000000000000000000000000000, s2 = 1, s1 = 0, s0 = 0, result = 00000000000000000000000000000001
# num32 = 00000000000000000000000000000000, num2 = 00000000000000000000000000000000, num3 = 00000000000000000000000000000000, num4 = 00000000000000000000000000000000,num5 = 00000000000000000000000000000000, num6 = 00000000000000000000000000000001, num7 = 00000000000000000000000000000000, num8 = 00000000000000000000000000000000, s2 = 1, s1 = 0, s0 = 1, result = 00000000000000000000000000000001
# num32 = 00000000000000000000000000000000, num2 = 00000000000000000000000000000000, num3 = 00000000000000000000000000000000, num4 = 00000000000000000000000000000000,num5 = 00000000000000000000000000000000, num6 = 00000000000000000000000000000000, num7 = 00000000000000000000000000000000, num8 = 00000000000000000000000000000001, s2 = 1, s1 = 1, s0 = 0, result = 00000000000000000000000000000001
# num32 = 00000000000000000000000000000000, num2 = 00000000000000000000000000000000, num3 = 00000000000000000000000000000000, num4 = 00000000000000000000000000000000,num5 = 00000000000000000000000000000000, num6 = 00000000000000000000000000000000, num7 = 00000000000000000000000000000000, num8 = 00000000000000000000000000000001, s2 = 1, s1 = 1, s0 = 1, result = 00000000000000000000000000000001
```

ALU bir bitlik modül :

```
# OPCODE = 0 0 1 ,      Num1 = 0,       Num2 = 0,       Cin = 0,        Cout = 0,       Result = 0
# OPCODE = 0 0 1 ,      Num1 = 1,       Num2 = 1,       Cin = 0,        Cout = 1,       Result = 1
# OPCODE = 0 0 1 ,      Num1 = 1,       Num2 = 0,       Cin = 0,        Cout = 0,       Result = 1
# OPCODE = 0 0 1 ,      Num1 = 0,       Num2 = 1,       Cin = 0,        Cout = 0,       Result = 1
# OPCODE = 0 0 1 ,      Num1 = 1,       Num2 = 1,       Cin = 1,        Cout = 1,       Result = 1
# OPCODE = 0 1 1 ,      Num1 = 0,       Num2 = 0,       Cin = 0,        Cout = 0,       Result = 0
# OPCODE = 0 1 1 ,      Num1 = 1,       Num2 = 1,       Cin = 0,        Cout = 1,       Result = 0
# OPCODE = 0 1 1 ,      Num1 = 1,       Num2 = 0,       Cin = 0,        Cout = 0,       Result = 1
# OPCODE = 0 1 1 ,      Num1 = 0,       Num2 = 1,       Cin = 0,        Cout = 0,       Result = 1
# OPCODE = 0 1 1 ,      Num1 = 1,       Num2 = 1,       Cin = 1,        Cout = 1,       Result = 0
# OPCODE = 0 1 0 ,      Num1 = 0,       Num2 = 0,       Cin = 0,        Cout = 0,       Result = 0
# OPCODE = 0 1 0 ,      Num1 = 1,       Num2 = 1,       Cin = 0,        Cout = 1,       Result = 0
# OPCODE = 0 1 0 ,      Num1 = 1,       Num2 = 0,       Cin = 0,        Cout = 0,       Result = 1
# OPCODE = 0 1 0 ,      Num1 = 0,       Num2 = 1,       Cin = 0,        Cout = 0,       Result = 1
# OPCODE = 0 1 0 ,      Num1 = 1,       Num2 = 1,       Cin = 1,        Cout = 1,       Result = 1
# OPCODE = 0 0 0 ,      Num1 = 0,       Num2 = 0,       Cin = 0,        Cout = 0,       Result = 0
# OPCODE = 0 0 0 ,      Num1 = 1,       Num2 = 1,       Cin = 0,        Cout = 1,       Result = 1
# OPCODE = 0 0 0 ,      Num1 = 1,       Num2 = 0,       Cin = 0,        Cout = 0,       Result = 0
# OPCODE = 0 0 0 ,      Num1 = 0,       Num2 = 1,       Cin = 0,        Cout = 0,       Result = 0
# OPCODE = 0 0 0 ,      Num1 = 1,       Num2 = 1,       Cin = 1,        Cout = 1,       Result = 1
# OPCODE = 1 1 0 ,      Num1 = 0,       Num2 = 0,       Cin = 1,        Cout = 1,       Result = 0
# OPCODE = 1 1 0 ,      Num1 = 1,       Num2 = 0,       Cin = 1,        Cout = 1,       Result = 1
# OPCODE = 1 1 0 ,      Num1 = 0,       Num2 = 1,       Cin = 1,        Cout = 0,       Result = 1
# OPCODE = 1 1 0 ,      Num1 = 1,       Num2 = 1,       Cin = 1,        Cout = 1,       Result = 0

VSIM 5>
```

ALU otuz iki bitlik modül :

```
# time = 0,     OPCODE = 0 1 0 , Num1 = 00000000000000000000000000000000, Num2 = 00000000000000000000000000000000, Result = 00000000000000000000000000000000, Zero = 1
# time =30,     OPCODE = 0 1 0 , Num1 = 00000000000000000000000000000001, Num2 = 00000000000000000000000000000000, Result = 00000000000000000000000000000001, Zero = 0
# time =60,     OPCODE = 0 1 0 , Num1 = 00000000000000000000000000000000, Num2 = 00000000000000000000000000000001, Result = 00000000000000000000000000000001, Zero = 0
# time =90,     OPCODE = 0 1 0 , Num1 = 00000000000000000000000000000001, Num2 = 00000000000000000000000000000001, Result = 00000000000000000000000000000010, Zero = 0
# time =120,    OPCODE = 0 0 1 , Num1 = 00000000000000000000000000000000, Num2 = 00000000000000000000000000000000, Result = 00000000000000000000000000000000, Zero = 1
# time =150,    OPCODE = 0 0 1 , Num1 = 00000000000000000000000000000001, Num2 = 00000000000000000000000000000000, Result = 00000000000000000000000000000001, Zero = 0
# time =180,    OPCODE = 0 0 1 , Num1 = 00000000000000000000000000000000, Num2 = 00000000000000000000000000000001, Result = 00000000000000000000000000000001, Zero = 0
# time =210,    OPCODE = 0 0 1 , Num1 = 00000000000000000000000000000001, Num2 = 00000000000000000000000000000001, Result = 00000000000000000000000000000001, Zero = 0
# time =240,    OPCODE = 0 1 1 , Num1 = 00000000000000000000000000000000, Num2 = 00000000000000000000000000000000, Result = 00000000000000000000000000000000, Zero = 1
# time =270,    OPCODE = 0 1 1 , Num1 = 00000000000000000000000000000001, Num2 = 00000000000000000000000000000000, Result = 00000000000000000000000000000001, Zero = 0
# time =300,    OPCODE = 0 1 1 , Num1 = 00000000000000000000000000000000, Num2 = 00000000000000000000000000000001, Result = 00000000000000000000000000000001, Zero = 0
# time =330,    OPCODE = 0 1 1 , Num1 = 00000000000000000000000000000001, Num2 = 00000000000000000000000000000001, Result = 00000000000000000000000000000000, Zero = 1
# time =360,    OPCODE = 1 1 0 , Num1 = 00000000000000000000000000000000, Num2 = 00000000000000000000000000000000, Result = 00000000000000000000000000000000, Zero = 1
# time =390,    OPCODE = 1 1 0 , Num1 = 00000000000000000000000000000001, Num2 = 00000000000000000000000000000000, Result = 00000000000000000000000000000001, Zero = 0
# time =420,    OPCODE = 1 1 0 , Num1 = 00000000000000000000000000000000, Num2 = 00000000000000000000000000000001, Result = 11111111111111111111111111111111, Zero = 0
# time =450,    OPCODE = 1 1 0 , Num1 = 00000000000000000000000000000001, Num2 = 00000000000000000000000000000001, Result = 00000000000000000000000000000000, Zero = 1
# time =480,    OPCODE = 0 0 0 , Num1 = 00000000000000000000000000000000, Num2 = 00000000000000000000000000000000, Result = 00000000000000000000000000000000, Zero = 1
# time =510,    OPCODE = 0 0 0 , Num1 = 00000000000000000000000000000001, Num2 = 00000000000000000000000000000000, Result = 00000000000000000000000000000000, Zero = 1
# time =540,    OPCODE = 0 0 0 , Num1 = 00000000000000000000000000000001, Num2 = 00000000000000000000000000000001, Result = 00000000000000000000000000000001, Zero = 0
# time =570,    OPCODE = 0 0 0 , Num1 = 00000000000000000000000000000000, Num2 = 00000000000000000000000000000001, Result = 00000000000000000000000000000000, Zero = 1
```

ALU Control modül :

```
# ALUop = 00,    Function Field = xxxxxx,        Operation = 010
# ALUop = 01,    Function Field = xxxxxx,        Operation = 110
# ALUop = 10,    Function Field = 100000,        Operation = 010
# ALUop = 10,    Function Field = 100010,        Operation = 110
# ALUop = 10,    Function Field = 100110,        Operation = 011
# ALUop = 10,    Function Field = 100100,        Operation = 000
```

Control Unit modül :

```
# Opcode = 100011, RegDst = 0, Branch = 0, Bne = 0, MemtoRead = 1, MemtoReg = 1, ALUop =  0, MemWrite = 0, ALUsrc = 1, RegWrite =1, RegWrite2 = 0, Rtype = 0
# Opcode = 101011, RegDst = 0, Branch = 0, Bne = 0, MemtoRead = 0, MemtoReg = 0, ALUop =  0, MemWrite = 1, ALUsrc = 1, RegWrite =0, RegWrite2 = 0, Rtype = 0
# Opcode = 000100, RegDst = 0, Branch = 1, Bne = 0, MemtoRead = 0, MemtoReg = 0, ALUop =  1, MemWrite = 0, ALUsrc = 0, RegWrite =0, RegWrite2 = 0, Rtype = 0
# Opcode = 000101, RegDst = 0, Branch = 0, Bne = 1, MemtoRead = 0, MemtoReg = 0, ALUop =  1, MemWrite = 0, ALUsrc = 0, RegWrite =0, RegWrite2 = 0, Rtype = 0
# Opcode = 000000, RegDst = 1, Branch = 0, Bne = 0, MemtoRead = 0, MemtoReg = 0, ALUop =  0, MemWrite = 0, ALUsrc = 0, RegWrite =1, RegWrite2 = 1, Rtype = 1
```

Instruction Memory modül :

```
#  time =  0, Insruction: 10101111111101111000000100000000
#  time = 20, Insruction: 00000000111010000100100000100101
#  time = 40, Insruction: 00000011111100001000100000100010
#  time = 60, Insruction: 00101011101111000000000000011101
```

Comparator modül :

Bu modül addn subn gibi instructionlar için eklenmiştir. 1 girişi vardır ve bir çıkışı vardır. ALU'dan alınan sonuç giriş olarak kullanılır. İçerisinde sabit 32 bitlik 0 vardır. Giriş ile sabit 0 kıyaslanır. Kıyaslama işaret biti kontrolü ile sayı negatif mi, işaret biti 0 olduğun zaman kalan bitlerin OR'lanması ile sayı pozitif mi, bütün bitlerin OR'lanmasıyla sayı sıfır mı şeklinde 3 parametre oluşturmakla başlar. Bunlar multiplexerlara seçici olarak verilir. Multiplexerlar ise iki seçici ile yani S0 ve S1 seçicileri ile iki tane 4x1'lik multiplexer'a bağlanır. Buradan alınan çıktı ve kalan diğer seçici yani S2 ile 2x1'lik multiplexer'a aktarılır. Bu multiplexerlar 32 bitliktir. Bunların sonucu 32bitlik 1,2 ya da 3 sayılırını verir. Bu bizim çıktımızdır. S0, S1 ve S2 seçicileri pozitif mi negatif mi ve eşit mi parametrelerinden gelir. Aslında sadece 32bitlik 0 ile kıyaslamıştır. Tam bir kıyaslayıcı değildir.

```
# Number = 00000000000000000000000000000000, eq = 1, less = 0, upper = 0 , Data = 00000000000000000000000000000001
# Number = 10000000000000000000000000000000, eq = 0, less = 1, upper = 0 , Data = 00000000000000000000000000000010
# Number = 10000000000000000000000000000010, eq = 0, less = 1, upper = 0 , Data = 00000000000000000000000000000010
# Number = 00000000000000000000000000000010, eq = 0, less = 0, upper = 1 , Data = 00000000000000000000000000000011
```

Extender modül :

```
VSIM 4> step -current
# input = 0111111111111111, result = 00000000000000000111111111111111

VSIM 5>
```

MIPS modül :

Data yapısı bozulduğundan register yazma işleminde register dosyasının içini bozuyor. Bu yüzden bazıl işlemleri düzgün yapamamaktadır.

```
#
# time: 500,
# PC: 00000000000000000000000000000001,
# instruction: 00000000010000010001100000100000,
# Opcode= 000000 ,
# rs = 00010 ,
# rt = 00001 ,
# rd = 00011,
# function = 100000,
# immediate = 0001100000100000,
# RegDst = 1,
#  RType = 1 ,
# branch = 0,
# bne = 0,
# Mem_read = 0 ,
# MemToReg = 0,
# ALUop = 10,
# Mem_Write = 0,
# ALUsrc = 0,
# Reg_Write1 = 1,
# Reg_Write2 = 1,
# ReadData1 = 00000000000000000000000000000100,
# ReadData2 = 00000000000000000000000000011001,
# WriteData1 = 00000000000000000000xx00000xxx10x,
# WriteData2 = 00000000000000000000000000000011
#
#
# time: 1500,
# PC: 00000000000000000000000000000010,
# instruction: 00000000011001000010100000100001,
# Opcode= 000000 ,
# rs = 00011 ,
# rt = 00100 ,
# rd = 00101,
# function = 100001,
# immediate = 0010100000100001,
# RegDst = 1,
#  RType = 1 ,
# branch = 0,
# bne = 0,
# Mem_read = 0 ,
# MemToReg = 0,
# ALUop = 10,
# Mem_Write = 0,
# ALUsrc = 0,
# Reg_Write1 = 1,
# Reg_Write2 = 1,
# ReadData1 = 00000000000000000000000000001110,
# ReadData2 = 00000000000000000000000000000111,
# WriteData1 = 00000000000000000x0x00000xxxxxx,
# WriteData2 = 00000000000000000000000000000xx
#
```

```
# time: 2500,
# PC: 00000000000000000000000000000011,
# instruction: 00000000101001100011100000100100,
# Opcode= 000000 ,
# rs = 00101 ,
# rt = 00110 ,
# rd = 00111,
# function = 100100,
# immediate = 0011100000100100,
# RegDst = 1,
#  RType = 1 ,
# branch = 0,
# bne = 0,
# Mem_read = 0 ,
# MemToReg = 0,
# ALUop = 10,
# Mem_Write = 0,
# ALUsrc = 0,
# Reg_Write1 = 1,
# Reg_Write2 = 1,
# ReadData1 = 00000000000000000000000000001110,
# ReadData2 = 00000000000000000000000000100000,
# WriteData1 = 0000000000000000000000000000x00,
# WriteData2 = 000000000000000000000000000000xx
#
#
# time: 3500,
# PC: 00000000000000000000000000000100,
# instruction: 00000000111010000100100000100101,
# Opcode= 000000 ,
# rs = 00111 ,
# rt = 01000 ,
# rd = 01001,
# function = 100101,
# immediate = 0100100000100101,
# RegDst = 1,
#  RType = 1 ,
# branch = 0,
# bne = 0,
# Mem_read = 0 ,
# MemToReg = 0,
# ALUop = 10,
# Mem_Write = 0,
# ALUsrc = 0,
# Reg_Write1 = 1,
# Reg_Write2 = 1,
# ReadData1 = 00000000000000000000000000010101,
# ReadData2 = 00000000000000000000000000001100,
# WriteData1 = 0000000000000000x00x00000x1x101,
# WriteData2 = 00000000000000000000000000000011
#
```

```
#
# time: 4500,
# PC: 00000000000000000000000000000101,
# instruction: 00000001010010010101100000100111,
# Opcode= 000000 ,
# rs = 01010 ,
# rt = 01001 ,
# rd = 01011,
# function = 100111,
# immediate = 0101100000100111,
# RegDst = 1,
#  RType = 1 ,
# branch = 0,
# bne = 0,
# Mem_read = 0 ,
# MemToReg = 0,
# ALUop = 10,
# Mem_Write = 0,
# ALUsrc = 0,
# Reg_Write1 = 1,
# Reg_Write2 = 1,
# ReadData1 = 11111111111111111111111111111111,
# ReadData2 = 00000000000000000000000000010000,
# WriteData1 = xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx,
# WriteData2 = 00000000000000000000000000000xx
#
#
# time: 5500,
# PC: 00000000000000000000000000000110,
# instruction: 00000011111011000110100100000000,
# Opcode= 000000 ,
# rs = 11111 ,
# rt = 01100 ,
# rd = 01101,
# function = 000000,
# immediate = 0110100100000000,
# RegDst = 1,
#  RType = 1 ,
# branch = 0,
# bne = 0,
# Mem_read = 0 ,
# MemToReg = 0,
# ALUop = 10,
# Mem_Write = 0,
# ALUsrc = 0,
# Reg_Write1 = 1,
# Reg_Write2 = 1,
# ReadData1 = 00000000000000000000000000001110,
# ReadData2 = 00000000000000000000000000000010,
# WriteData1 = 0000000000000000xx0x00x000xxxx0,
# WriteData2 = 00000000000000000000000000000xx
#
```

```
# time: 6500,
# PC: 00000000000000000000000000000111,
# instruction: 00000001010011100111100010000010,
# Opcode= 000000 ,
# rs = 01010 ,
# rt = 01110 ,
# rd = 01111,
# function = 000010,
# immediate = 0111100010000010,
# RegDst = 1,
#  RType = 1 ,
# branch = 0,
# bne = 0,
# Mem_read = 0 ,
# MemToReg = 0,
# ALUop = 10,
# Mem_Write = 0,
# ALUsrc = 0,
# Reg_Write1 = 1,
# Reg_Write2 = 1,
# ReadData1 = 00000000000000000000000000000100,
# ReadData2 = 00000000000000000000000000010110,
# WriteData1 = 00000000000000000xxxx000x00xxxx0,
# WriteData2 = 0000000000000000000000000000000xx
#
#
# time: 7500,
# PC: 00000000000000000000000000001000,
# instruction: 00000011111110000100010000100010,
# Opcode= 000000 ,
# rs = 11111 ,
# rt = 10000 ,
# rd = 10001,
# function = 100010,
# immediate = 1000100000100010,
# RegDst = 1,
#  RType = 1 ,
# branch = 0,
# bne = 0,
# Mem_read = 0 ,
# MemToReg = 0,
# ALUop = 10,
# Mem_Write = 0,
# ALUsrc = 0,
# Reg_Write1 = 1,
# Reg_Write2 = 1,
# ReadData1 = xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx,
# ReadData2 = 00000000000000000000000000011000,
# WriteData1 = xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx,
# WriteData2 = 0000000000000000000000000000000xx
#
```

```
# time: 8500,
# PC: 00000000000000000000000000001001,
# instruction: 00000011110100011001000000100011,
# Opcode= 000000 ,
# rs = 11110 ,
# rt = 10001 ,
# rd = 10010,
# function = 100011,
# immediate = 1001000000100011,
# RegDst = 1,
#  RType = 1 ,
# branch = 0,
# bne = 0,
# Mem_read = 0 ,
# MemToReg = 0,
# ALUop = 10,
# Mem_Write = 0,
# ALUsrc = 0,
# Reg_Write1 = 1,
# Reg_Write2 = 1,
# ReadData1 = 0000000000000000xx0x00x000xxxx0,
# ReadData2 = 00000000000000000000000000011010,
# WriteData1 = xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx,
# WriteData2 = 000000000000000000000000000000xx
#
#
# time: 9500,
# PC: 00000000000000000000000000001010,
# instruction: 00000010010111011010000000101010,
# Opcode= 000000 ,
# rs = 10010 ,
# rt = 11101 ,
# rd = 10100,
# function = 101010,
# immediate = 1010000000101010,
# RegDst = 1,
#  RType = 1 ,
# branch = 0,
# bne = 0,
# Mem_read = 0 ,
# MemToReg = 0,
# ALUop = 10,
# Mem_Write = 0,
# ALUsrc = 0,
# Reg_Write1 = 1,
# Reg_Write2 = 1,
# ReadData1 = 00000000000000000000000000000110,
# ReadData2 = 00000000000000000000000000011011,
# WriteData1 = xxxxxxxxxxxxxxxx0x000000xxxxxxx,
# WriteData2 = 000000000000000000000000000000xx
#
```

```
# time: 10500,
# PC: 00000000000000000000000000001011,
# instruction: 00000010110101011010100000101011,
# Opcode= 000000 ,
# rs = 10110 ,
# rt = 10101 ,
# rd = 10101,
# function = 101011,
# immediate = 1010100000101011,
# RegDst = 1,
#  RType = 1 ,
# branch = 0,
# bne = 0,
# Mem_read = 0 ,
# MemToReg = 0,
# ALUop = 10,
# Mem_Write = 0,
# ALUsrc = 0,
# Reg_Write1 = 1,
# Reg_Write2 = 1,
# ReadData1 = 00000000000000000000000000011100,
# ReadData2 = 00000000000000000000000000000001,
# WriteData1 = xxxxxxxxxxxxxxxx0x0x0000xxxx1xx,
# WriteData2 = 00000000000000000000000000000xx
#
#
# time: 11500,
# PC: 00000000000000000000000000001100,
# instruction: 00100010101101110000000000010010,
# Opcode= 001000 ,
# rs = 10101 ,
# rt = 10111 ,
# rd = 00000,
# function = 010010,
# immediate = 0000000000010010,
# RegDst = 0,
#  RType = 0 ,
# branch = 0,
# bne = 0,
# Mem_read = 0 ,
# MemToReg = 0,
# ALUop = 00,
# Mem_Write = 0,
# ALUsrc = 0,
# Reg_Write1 = 0,
# Reg_Write2 = 0,
# ReadData1 = 00000000000000000000000000001110,
# ReadData2 = 00000000000000000000000000000111,
# WriteData1 = 00000000000000000000000000xxxxxx,
# WriteData2 = 00000000000000000000000000000xx
#
```

```
#
# time: 12500,
# PC: 00000000000000000000000000001101,
# instruction: 00110010111110010000000000011011,
# Opcode= 001100 ,
# rs = 10111 ,
# rt = 11001 ,
# rd = 00000,
# function = 011011,
# immediate = 0000000000011011,
# RegDst = 0,
#  RType = 0 ,
# branch = 0,
# bne = 0,
# Mem_read = 0 ,
# MemToReg = 0,
# ALUop = 00,
# Mem_Write = 0,
# ALUsrc = 0,
# Reg_Write1 = 0,
# Reg_Write2 = 0,
# ReadData1 = 00000000000000000000000000000111,
# ReadData2 = 00000000000000000000000000001110,
# WriteData1 = 00000000000000000000000000xxxxxx,
# WriteData2 = 0000000000000000000000000000000xx
#
#
# time: 13500,
# PC: 00000000000000000000000000001110,
# instruction: 00100111001110100000000000100011,
# Opcode= 001001 ,
# rs = 11001 ,
# rt = 11010 ,
# rd = 00000,
# function = 100011,
# immediate = 0000000000100011,
# RegDst = 0,
#  RType = 0 ,
# branch = 0,
# bne = 0,
# Mem_read = 0 ,
# MemToReg = 0,
# ALUop = 00,
# Mem_Write = 0,
# ALUsrc = 0,
# Reg_Write1 = 0,
# Reg_Write2 = 0,
# ReadData1 = 00000000000000000000000000001110,
# ReadData2 = 00000000000000000000000000000001,
# WriteData1 = 00000000000000000000000000xxxxxx,
# WriteData2 = 0000000000000000000000000000000xx
#
```

```
#
# time: 14500,
# PC: 00000000000000000000000000001111,
# instruction: 00110111010110110000000000011001,
# Opcode= 001101 ,
# rs = 11010 ,
# rt = 11011 ,
# rd = 00000,
# function = 011001,
# immediate = 0000000000011001,
# RegDst = 0,
#  RType = 0 ,
# branch = 0,
# bne = 0,
# Mem_read = 0 ,
# MemToReg = 0,
# ALUop = 00,
# Mem_Write = 0,
# ALUsrc = 0,
# Reg_Write1 = 0,
# Reg_Write2 = 0,
# ReadData1 = 00000000000000000000000000000001,
# ReadData2 = 00000000000000000000000000000010,
# WriteData1 = 00000000000000000000000000000xxxxx,
# WriteData2 = 0000000000000000000000000000000xx
#
#
# time: 15500,
# PC: 00000000000000000000000000010000,
# instruction: 00101011101111000000000000011101,
# Opcode= 001010 ,
# rs = 11101 ,
# rt = 11100 ,
# rd = 00000,
# function = 011101,
# immediate = 0000000000011101,
# RegDst = 0,
#  RType = 0 ,
# branch = 0,
# bne = 0,
# Mem_read = 0 ,
# MemToReg = 0,
# ALUop = 00,
# Mem_Write = 0,
# ALUsrc = 0,
# Reg_Write1 = 0,
# Reg_Write2 = 0,
# ReadData1 = 00000000000000000000000000000010,
# ReadData2 = 00000000000000000000000000011110,
# WriteData1 = 00000000000000000000000000xxxxxx,
# WriteData2 = 0000000000000000000000000000xx
#
```

```
#
# time: 16500,
# PC: 00000000000000000000000000010001,
# instruction: 10001111110111110000000001000000,
# Opcode= 100011 ,
# rs = 11110 ,
# rt = 11111 ,
# rd = 00000,
# function = 000000,
# immediate = 0000000001000000,
# RegDst = 0,
#  RType = 0 ,
# branch = 0,
# bne = 0,
# Mem_read = 1 ,
# MemToReg = 1,
# ALUop = 00,
# Mem_Write = 0,
# ALUsrc = 1,
# Reg_Write1 = 1,
# Reg_Write2 = 0,
# ReadData1 = 00000000000000000000000000000001,
# ReadData2 = 00000000000000000000000000000100,
# WriteData1 = xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx,
# WriteData2 = 00000000000000000000000000000011
#
```