TreadMill

COA Lab Project (2016-17)
The LNMIIT, Jaipur

Khushboo Baheti (15UCS168)
Smriti Jha (15UCS142)
Shreya Rajput (15UCS134)
Sakshi Goyal (15UCS116)

Introduction

- We are designing a **16-bit** Treadmill Machine.
- Includes functions that can measure and analyse procedures or the levels of an activity that can properly attribute performance factor.

  Example:

- **CaloriesBurnt**: It will tell the amount of calories burnt during a particular run for a particular time.
- **VO2Consumption**:which will tell the amount of oxygen consumed in a run.

Functions

1. VO2Consumption:Vo2 measurement represents the rate of oxygen consumption in that activity,which is, ***VO2(ml/(kg*min))***=[(0.2*speed(m/min))+(0.9*speed*% grade)+3.5
2. setGrade:This function is used to initialise the inclination of the treadmill floor in terms of tan theta taken as input.
3. MET: Metabolic Equivalent of Task, 1 met is the energy equivalent expended by an individual while seated at rest.(1MET=3.5ml/(kg*min))
   MET(mets)=VO2/3.5
4. CaloriesBurnt: This can be derived from the relation
   Calories burnt:Time spent running(min)*[MET value*3.5*weight of body (in kg)/ 200]
5. CalculateDistance:Given the speed(m/min) and the time (min),***Distance(m)***=speed/time.

6. SetSpeed:This function is used to initialise the speed of machine (in m/min).

7. Settime:This function starts the treadmill by taking time of running (in minutes) as an argument. It starts the timer accordingly.

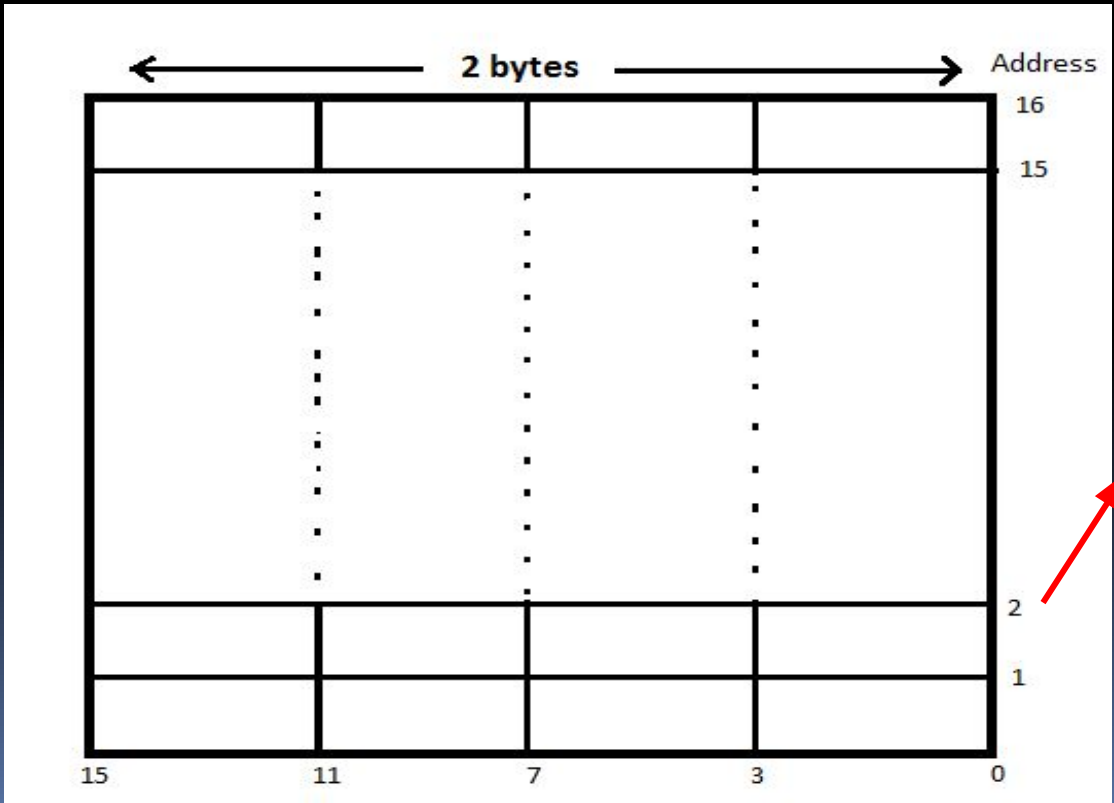
8. SpeedInc:This will increment the current speed by 1.

9. SpeedDec:This will decrement the current speed by 1.

10. SetWeight: This function will initialise the weight as the weight given by the user(in kgs).

Memory Model

- We are using Little Endian model,where word is made up of 16 bits(2 bytes).
- Our memory is word addressable memory.
- We are using an Aligned memory model.



Registers

- R0-R6 are **general purpose** registers which will be used to store key local variables and immediate results of calculations.

R0	0000
R1	0001
.	.
R6	0101

- **Special Registers**:

rGRD	0111
rWGT	1000
rTME	1001
rSPD	1010
rIN	1011
rOUT	1100
rPC	1101
rIR	1110
rFR	1111

- Special-purpose registers control various caches, memory, I/O devices and other hardware features.
- IN/OUT registers will be used to take input & display output.
- GRD(Grade),WGT(Weight),TME(Time),SPD(Speed),PC(Program Counter),IR(Instruction register),FR(Flag register) are special purpose registers.

Instruction Format

- **2 - Address Instructions:**

<------8 bits ---><----4 bits-----><----- 4 bits------>

Example-	Caloxygen	R0	R1
	Caldist	R1	R2

- **Immediate Address Instruction:**

Example-	setspeed	imm	R1
	settime	imm	R1
	setweight	imm	R1
	setgrade	imm	R1

- **1- Address Instructions:**

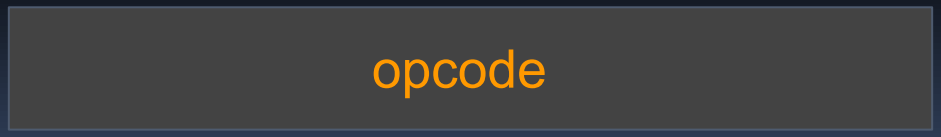
<---------12 bits-------><----------4 bits---------->

Example-	caloburnt	R1
	calmet	R1
	cspdinc	R1
	cspddec	R1

<-----------4bits------------><-------------12bits-------->

Example-	sjump	LABEL

- o-Address Instructions:



<----------------------16bits----------------------->

Example- hlt

In our instructions, second operand will be the destination register in which the result will be stored.

Instruction Design

4 bits:	setspeed	0000
	settime	0001
	setweight	0010
	setgrade	0011
	sjump	0100
8 bits:	Caldist	01010000
	Caloxygen	01010001
12 bits:	caloburnt	010100100000
	calmet	010100100001
	cspdinc	010100100010
	cspddec	010100100011
16 bits:	hlt	0101001001000000

- RISC due to faster clock speeds.
- Fixed length instructions because of easy decoding.

Instruction Types

- Data Movement:
  Register to Register= Caldist
  Immediate to Register=setspeed
- Dyadic Instructions:

Caldist	op1 op2	settime	op1 op2
Caloxygen	op1 op2	setweight	op1 op2
setspeed	op1 op2	setgrade	op1 op2

- Monodic Instructions:

caloburnt	op1	cspdinc	op1
calmet	op1	sjump	label

- Branching:

sjump	label

Data Types

We are using only Unsigned Integers and floats .

Addressing Modes

- Immediate Addressing Modes:

setspeed	value	R1
settime	value	R1
setweight	value	R1
setgrade	value	R1

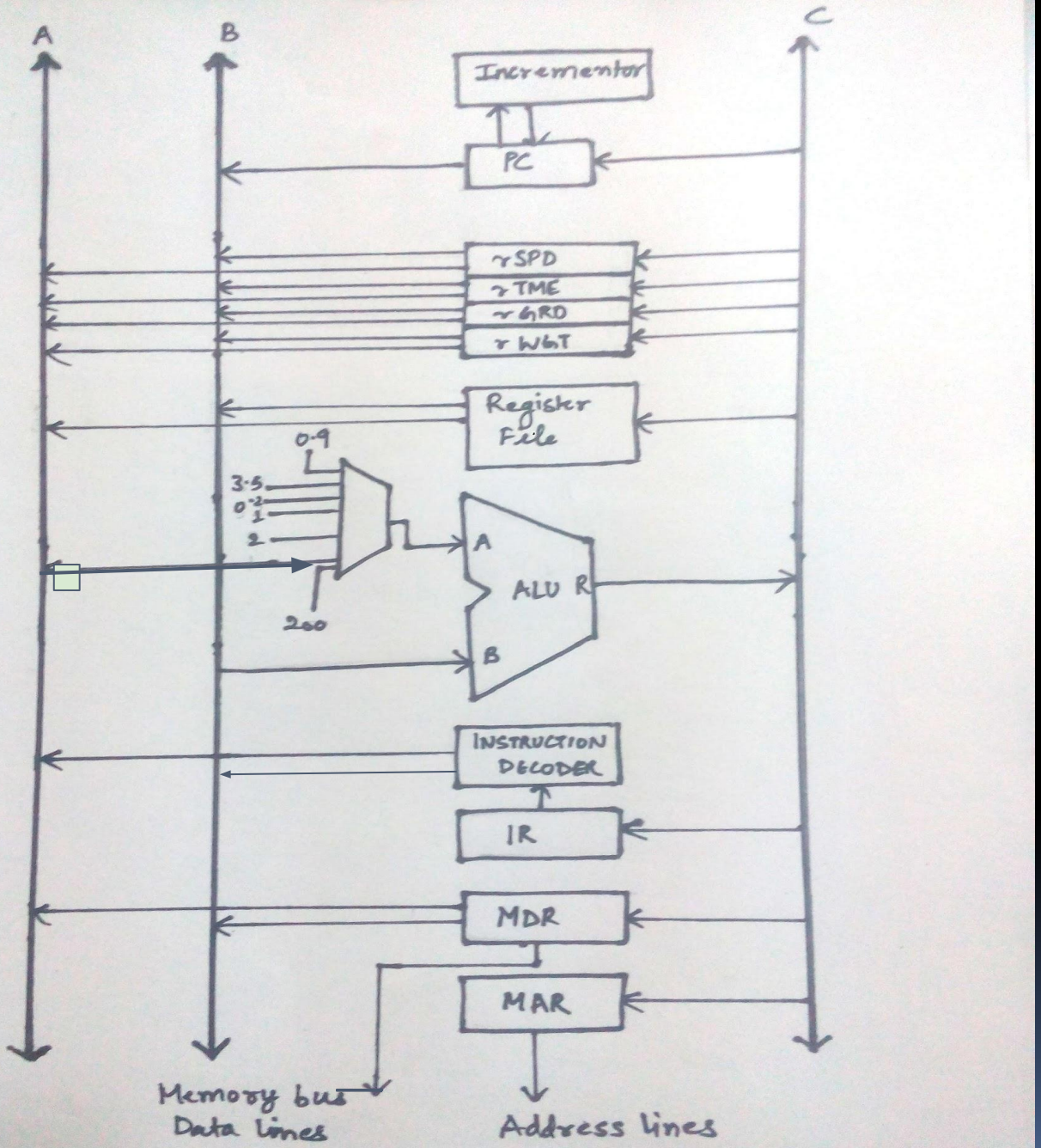
- Register Addressing Modes:

Caloxygen	Ro	R1
Caldist	Ro	R1

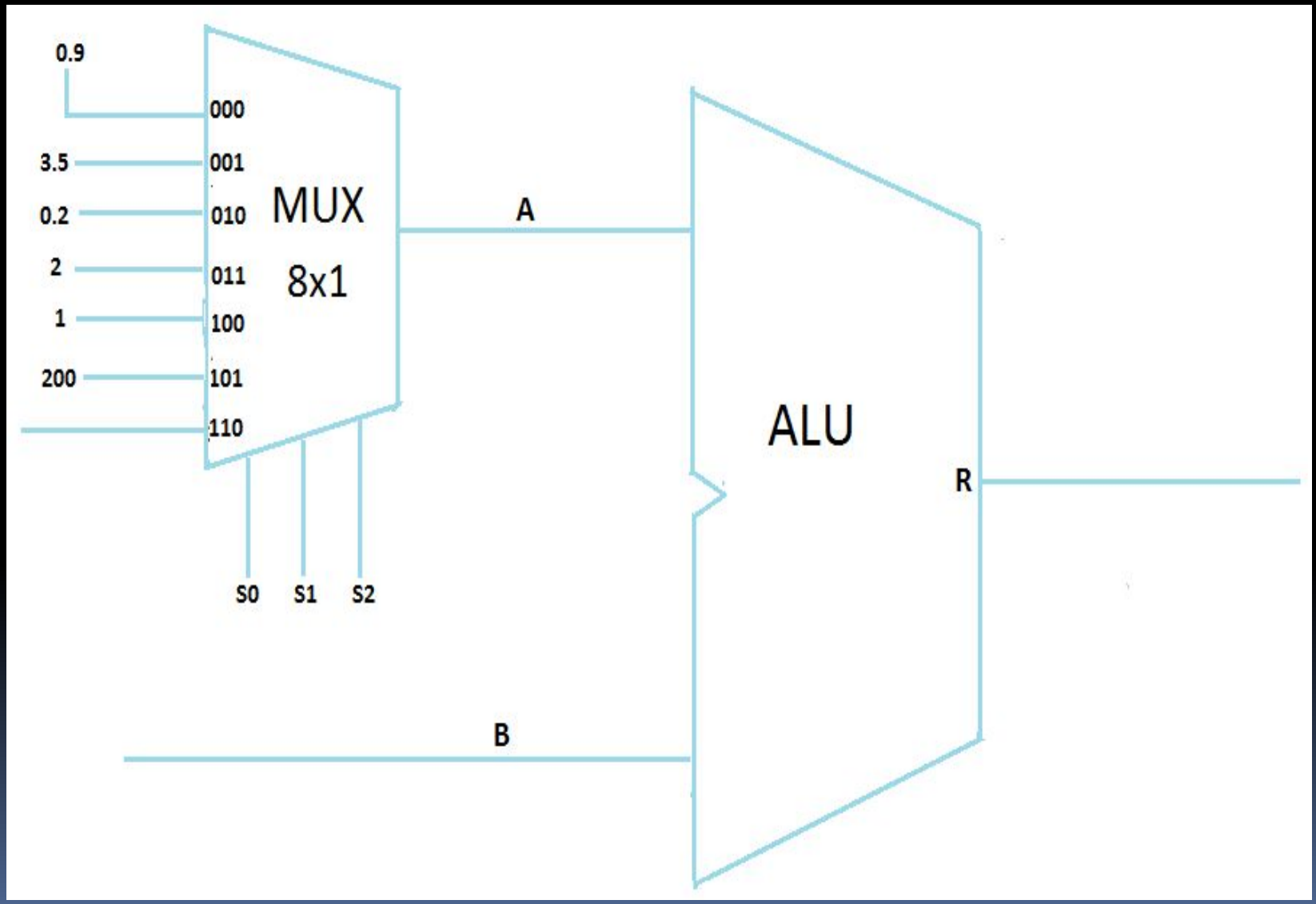
Flow of Control Handling

Sequential flow of control and Branching are checked in which LABEL provides the name for a particular address where jump statements are used to change or control the flow of program.

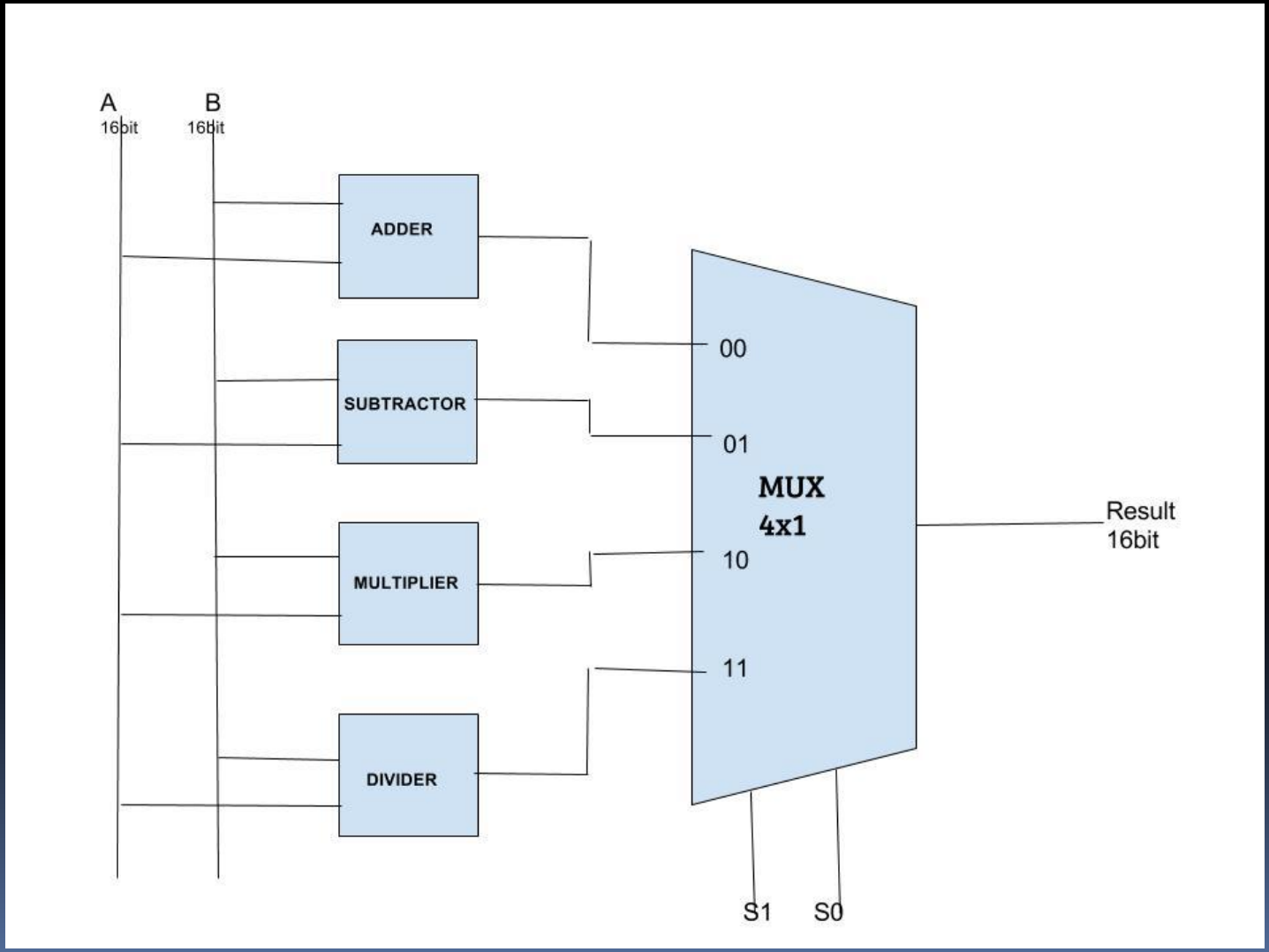
DATA-PATH


A
B
C
Incrementor
PC
rSPD
rTME
rGRD
rWGT
Register File
0-9
3-5
0-2
1
2
200
A
ALU R
B
INSTRUCTION DECODER
IR
MDR
MAR
Memory bus Data lines
Address lines

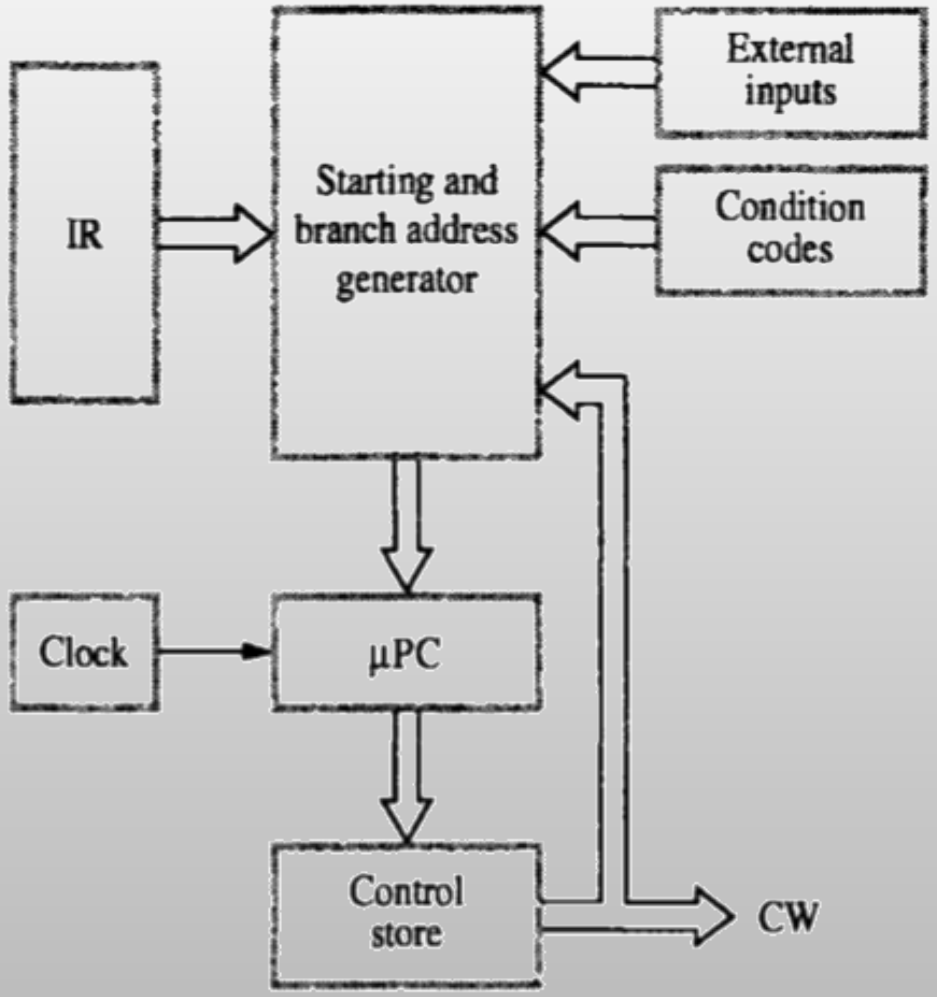
ALU


0.9
3.5
0.2
2
1
200
000
001
010
011
100
101
110
MUX
8x1
A
S0
S1
S2
ALU
R
B

ALU DESIGN


A
16bit
B
16bit
ADDER
SUBTRACTOR
MULTIPLIER
DIVIDER
00
01
10
11
MUX
4x1
Result
16bit
S1
S0

CONTROL-UNIT



CONTROL UNIT (Microprogrammed)

1. **Fetch Cycle:**

   [MAR] ← [PC]

   [PC] ← [PC] + 2

   Read, WMFC

   [IR] ← [MDR]

Control Signals:

1. PC_{out}, R=B, MAR_{in}, Read, IncPC
2. WMFC
3. MDR_{out}, R=B, IR_{in}

	0	0	1	Read
	0	1	0	WMFC
	0	0	1	INC PC
	0	0	0	Offset
	0	0	0	Imm IR out B
	0	0	1	PC out B
	1	0	0	MDR out B
	0	0	0	IR out B
	0	0	0	rSPD out A
	0	0	0	rTME out A
	0	0	0	rGRD out A
	0	0	0	rWGT out A
	0	0	0	r0 out A
	0	0	0	r1 out A
	0	0	0	r2 out A
	0	0	0	r3 out A
	0	0	0	r5 out A
	0	0	0	rSPD out B
	0	0	0	rTME out B
	0	0	0	rGRD out B
	0	0	0	rWGT out B
	0	0	0	r0 out B
	0	0	0	r1 out B
	0	0	0	r2 out B
	0	0	0	r3 out B
	0	0	0	r5 out B
	0	0	0	END

	3	2	1	Cycle no.
	0	0	0	PC in
	0	0	1	MAR in
	1	0	0	IR in
	0	0	0	rSPD in
	0	0	0	rTME in
	0	0	0	rWGT in
	0	0	0	rGRD in
	0	0	0	r0 in
	0	0	0	r1 in
	0	0	0	r2 in
	0	0	0	r3 in
	0	0	0	r5 in
	0	0	0	Select 0.2
	0	0	0	Select 0.9
	0	0	0	Select 3.5
	0	0	0	select A
	0	0	0	R=A
	1	0	1	R=B
	0	0	0	Select 1
	0	0	0	Select 2
	0	0	0	Select 200
	0	0	0	Add
	0	0	0	Sub
	0	0	0	Mul
	0	0	0	Div

2. **setspeed:**

[rSPD] ←[Imm value IR]

Control Signals:

Immediate value of IR_{outB}, R=B, $rSPD_{in}$, END

Cycle no	PC in	MAR in	IR in	rSPD in	rTME in	rWGT in	rGRD in	r0 in	r1 in	r2 in	r3 in	r5 in	Select 0.2	Select 0.9	Select 3.5	select A	R=A	R=B	Select 1	Select 2	Select 200	Add	Sub	Mul	Div
1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0

Read	WMFC	INC PC	Offset	Imm IR out B	PC out B	MDR out B	IR out B	rSPD out A	rTME out A	rGRD out A	rWGT out A	r0 out A	r1 out A	r2 out A	r3 out A	r5 out A	rSPD out B	rTME out B	rGRD out B	rWGT out B	r0 out B	r1 out B	r2 out B	r3 out B	r5 out B	END
0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

3. Caldist:

$[R_3] \leftarrow [rSPD] \times [rTME]$

Control Signals:

$rSPD_{outA}$, $rTME_{outB}$, select A, mul, $R3_{in}$, END

Cycle no	PC in	MAR in	IR in	rSPD in	rTME in	rWGT in	rGRD in	r0 in	r1 in	r2 in	r3 in	r5 in	Select 0.2	Select 0.9	Select 3.5	select A	R=A	R=B	Select 1	Select 2	Select 200	Add	Sub	Mul	Div	Read
1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0

WMFC	INC PC	Offset	Imm IR out B	PC out B	MDR out B	IR out B	rSPD out A	rTME out A	rGRD out A	rWGT out A	r0 out A	r1 out A	r2 out A	r3 out A	r5 out A	rSPD out B	rTME out B	rGRD out B	rWGT out B	r0 out B	r1 out B	r2 out B	r3 out B	r5 out B	END
0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1

4. **settime:**

[rTME] ←[Imm value IR]

Control SIgnals:

Immediate value of IR_{outB}, R=B, $rTME_{in}$, END

Cycle No	PC in	MAR in	IR in	rSPD in	rTME in	rWGT in	rGRD in	r0 in	r1 in	r2 in	r3 in	r5 in	Select 0.2	Select 0.9	Select 3.5	select A	R=A	R=B	Select 1	Select 2	Select 200	Add	Sub	Mul	Div
1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0

Read	WMFC	INC PC	Offset	Imm IR out B	PC out B	MDR out B	IR out B	rSPD out A	rTME out A	rGRD out A	rWGT out A	r0 out A	r1 out A	r2 out A	r3 out A	r5 out A	rSPD out B	rTME out B	rGRD out B	rWGT out B	r0 out B	r1 out B	r2 out B	r3 out B	r5 out B	END
0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

5. **setweight:**

[rWGT] ←[Imm value IR]

Control SIgnals:

Immediate value of IR_{outB}, R=B, $rWGT_{in}$, END

Cycle no	PC in	MAR in	IR in	rSPD in	rTME in	rWGT in	rGRD in	r0 in	r1 in	r2 in	r3 in	r5 in	Select 0.2	Select 0.9	Select 3.5	select A	R=A	R=B	Select 1	Select 2	Select 200	Add	Sub	Mul	Div
1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0

Read	WMFC	INC PC	Offset	Imm IR out B	PC out B	MDR out B	IR out B	rSPD out A	rTME out A	rGRD out A	rWGT out A	r0 out A	r1 out A	r2 out A	r3 out A	r5 out A	rSPD out B	rTME out B	rGRD out B	rWGT out B	r0 out B	r1 out B	r2 out B	r3 out B	r5 out B	END
0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

6. **Caloxygen:**

[R0] ←[rSPD] x [0.2]

[R5] ←[rGRD] x [0.9]

[R5] ←[R5] x [rSPD]

[R0] ←[R0] + [R5]

[R0] ←[R0] + [3.5]

Control Signals:

1. $rSPD_{outB}$, select 0.2, mul, $R0_{in}$
2. $rGRD_{outB}$, select 0.9, mul, $R5_{in}$
3. $R5_{outA}$, $rSPD_{outB}$, select A, mul, $R5_{in}$
4. $R0_{outA}$, $R5_{outB}$, select A, add, $R0_{in}$
5. $R0_{outB}$, select 3.5, add, $R0_{in}$, END

	0	0	0	0	0	Read
	0	0	0	0	0	WMFC
	0	0	0	0	0	INC PC
	0	0	0	0	0	Offset
	0	0	0	0	0	Imm IR out B
	0	0	0	0	0	PC out B
	0	0	0	0	0	MDR out B
	0	0	0	0	0	IR out B
	0	0	0	0	0	rSPD out A
	0	0	0	0	0	rTME out A
	0	0	0	0	0	rGRD out A
	0	0	0	0	0	rWGT out A
	0	1	0	0	0	r0 out A
	0	0	0	0	0	r1 out A
	0	0	0	0	0	r2 out A
	0	0	0	0	0	r3 out A
	0	0	1	0	0	r5 out A
	0	0	1	0	1	rSPD out B
	0	0	0	0	0	rTME out B
	0	0	0	1	0	rGRD out B
	0	0	0	0	0	rWGT out B
	1	0	0	0	0	r0 out B
	0	0	0	0	0	r1 out B
	0	0	0	0	0	r2 out B
	0	0	0	0	0	r3 out B
	0	1	0	0	0	r5 out B
	1	0	0	0	0	END

	5	4	3	2	1	Cycle no
	0	0	0	0	0	PC in
	0	0	0	0	0	MAR in
	0	0	0	0	0	IR in
	0	0	0	0	0	rSPD in
	0	0	0	0	0	rTME in
	0	0	0	0	0	rWGT in
	0	0	0	0	0	rGRD in
	1	1	0	0	1	r0 in
	0	0	0	0	0	r1 in
	0	0	0	0	0	r2 in
	0	0	0	0	0	r3 in
	0	0	1	1	0	r5 in
	0	0	0	0	1	Select 0.2
	0	0	0	1	0	Select 0.9
	1	0	0	0	0	Select 3.5
	0	1	1	0	0	select A
	0	0	0	0	0	R=A
	0	0	0	0	0	R=B
	0	0	0	0	0	Select 1
	0	0	0	0	0	Select 2
	0	0	0	0	0	Select 200
	1	1	0	0	0	Add
	0	0	0	0	0	Sub
	0	0	1	1	1	Mul
	0	0	0	0	0	Div

7. **calmet:**

[R1] ←[R0] / [3.5]

Control Signals:

$R0_{outB}$, select 3.5, div, $R1_{in}$, END

Cycle no	PC in	MAR in	IR in	rSPD in	rTME in	rWGT in	rGRD in	r0 in	r1 in	r2 in	r3 in	r5 in	Select 0.2	Select 0.9	Select 3.5	select A	R=A	R=B	Select 1	Select 2	Select 200	Add	Sub	Mul	Div
1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1

Read	WMFC	INC PC	Offset	Imm IR out B	PC out B	MDR out B	IR out B	rSPD out A	rTME out A	rGRD out A	rWGT out A	r0 out A	r1 out A	r2 out A	r3 out A	r5 out A	rSPD out B	rTME out B	rGRD out B	rWGT out B	r0 out B	r1 out B	r2 out B	r3 out B	r5 out B	END
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1

8. **caloburnt:**

[R1] ←[R1] x [rWGT]

[R1] ←[R1] x [3.5]

[R1] ←[R1] x [rTME]

[R2] ←[R1] / [200]

Control signals:

1. $R1_{outA}$, $rWGT_{outB}$, select A, mul, $R1_{in}$
2. $R1_{outB}$, select 3.5, mul, $R1_{in}$
3. $R1_{outB}$, $rTME_{outA}$, select A, mul, $R1_{in}$
4. $R1_{outB}$, select 200, div, $R2_{in}$, END

	0	0	0	0	Read
	0	0	0	0	WMFC
	0	0	0	0	INC PC
	0	0	0	0	Offset
	0	0	0	0	Imm IR out B
	0	0	0	0	PC out B
	0	0	0	0	MDR out B
	0	0	0	0	IR out B
	0	0	0	0	rSPD out A
	0	1	0	0	rTME out A
	0	0	0	0	rGRD out A
	0	0	0	0	rWGT out A
	0	0	0	0	r0 out A
	0	0	0	1	r1 out A
	0	0	0	0	r2 out A
	0	0	0	0	r3 out A
	0	0	0	0	r5 out A
	0	0	0	0	rSPD out B
	0	0	0	0	rTME out B
	0	0	0	0	rGRD out B
	0	0	0	1	rWGT out B
	0	0	0	0	r0 out B
	1	1	1	0	r1 out B
	0	0	0	0	r2 out B
	0	0	0	0	r3 out B
	0	0	0	0	r5 out B
	1	0	0	0	END

	4	3	2	1	Cycle no
	0	0	0	0	PC in
	0	0	0	0	MAR in
	0	0	0	0	IR in
	0	0	0	0	rSPD in
	0	0	0	0	rTME in
	0	0	0	0	rWGT in
	0	0	0	0	rGRD in
	0	0	0	0	r0 in
	0	1	1	1	r1 in
	1	0	0	0	r2 in
	0	0	0	0	r3 in
	0	0	0	0	r5 in
	0	0	0	0	Select 0.2
	0	0	0	0	Select 0.9
	0	0	1	0	Select 3.5
	0	1	0	1	select A
	0	0	0	0	R=A
	0	0	0	0	R=B
	0	0	0	0	Select 1
	0	0	0	0	Select 2
	1	0	0	0	Select 200
	0	0	0	0	Add
	0	0	0	0	Sub
	0	1	1	1	Mul
	1	0	0	0	Div

9. **setgrade:**

[rGRD] ←[Imm value IR]

Control SIgnals:

Immediate value of IR_{outB}, R=B, $rGRD_{in}$, END

Cycle no	PC in	MAR in	IR in	rSPD in	rTME in	rWGT in	rGRD in	r0 in	r1 in	r2 in	r3 in	r5 in	Select 0.2	Select 0.9	Select 3.5	select A	R=A	R=B	Select 1	Select 2	Select 200	Add	Sub	Mul	Div
1	0	0	0	0	0	0	**1**	0	0	0	0	0	0	0	0	0	0	**1**	0	0	0	0	0	0	0

Read	WMFC	INC PC	Offset	Imm IR out B	PC out B	MDR out B	IR out B	rSPD out A	rTME out A	rGRD out A	rWGT out A	r0 out A	r1 out A	r2 out A	r3 out A	r5 out A	rSPD out B	rTME out B	rGRD out B	rWGT out B	r0 out B	r1 out B	r2 out B	r3 out B	r5 out B	END
0	0	0	0	**1**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**1**

10. **cspdInc:**

[rSPD] ←[rSPD] + [1]

Control signals:

$rSPD_{outB}$, select 1, add, $rSPD_{in}$, END

Cycle no	PC in	MAR in	IR in	rSPD in	rTME in	rWGT in	rGRD in	r0 in	r1 in	r2 in	r3 in	r5 in	Select 0.2	Select 0.9	Select 3.5	select A	R=A	R=B	Select 1	Select 2	Select 200	Add	Sub	Mul	Div
1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0

Read	WMFC	INC PC	Offset	Imm IR out B	PC out B	MDR out B	IR out B	rSPD out A	rTME out A	rGRD out A	rWGT out A	r0 out A	r1 out A	r2 out A	r3 out A	r5 out A	rSPD out B	rTME out B	rGRD out B	rWGT out B	r0 out B	r1 out B	r2 out B	r3 out B	r5 out B	END
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1

11. **cspdDec:**

[rSPD] ←[rSPD] - [1]

Control signals:

$rSPD_{outB}$, select 1, sub, $rSPD_{in}$, END

Cycle no	PC in	MAR in	IR in	rSPD in	rTME in	rWGT in	rGRD in	r0 in	r1 in	r2 in	r3 in	r5 in	Select 0.2	Select 0.9	Select 3.5	select A	R=A	R=B	Select 1	Select 2	Select 200	Add	Sub	Mul	Div
1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0

Read	WMFC	INC PC	Offset	Imm IR out B	PC out B	MDR out B	IR out B	rSPD out A	rTME out A	rGRD out A	rWGT out A	r0 out A	r1 out A	r2 out A	r3 out A	r5 out A	rSPD out B	rTME out B	rGRD out B	rWGT out B	r0 out B	r1 out B	r2 out B	r3 out B	r5 out B	END
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1

12. sjump:

[PC] ←[offset field IR]

Control signals:

offset field IRoutB, R=B, PCin,END

Cycle no	PC in	MAR in	IR in	rSPD in	rTME in	rWGT in	rGRD in	r0 in	r1 in	r2 in	r3 in	r5 in	Select 0.2	Select 0.9	Select 3.5	select A	R=A	R=B	Select 1	Select 2	Select 200	Add	Sub	Mul	Div
1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0

Read	WMFC	INC PC	Offset	Imm IR out B	PC out B	MDR out B	IR out B	rSPD out A	rTME out A	rGRD out A	rWGT out A	r0 out A	r1 out A	r2 out A	r3 out A	r5 out A	rSPD out B	rTME out B	rGRD out B	rWGT out B	r0 out B	r1 out B	r2 out B	r3 out B	r5 out B	END
0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

THANK YOU!!!!