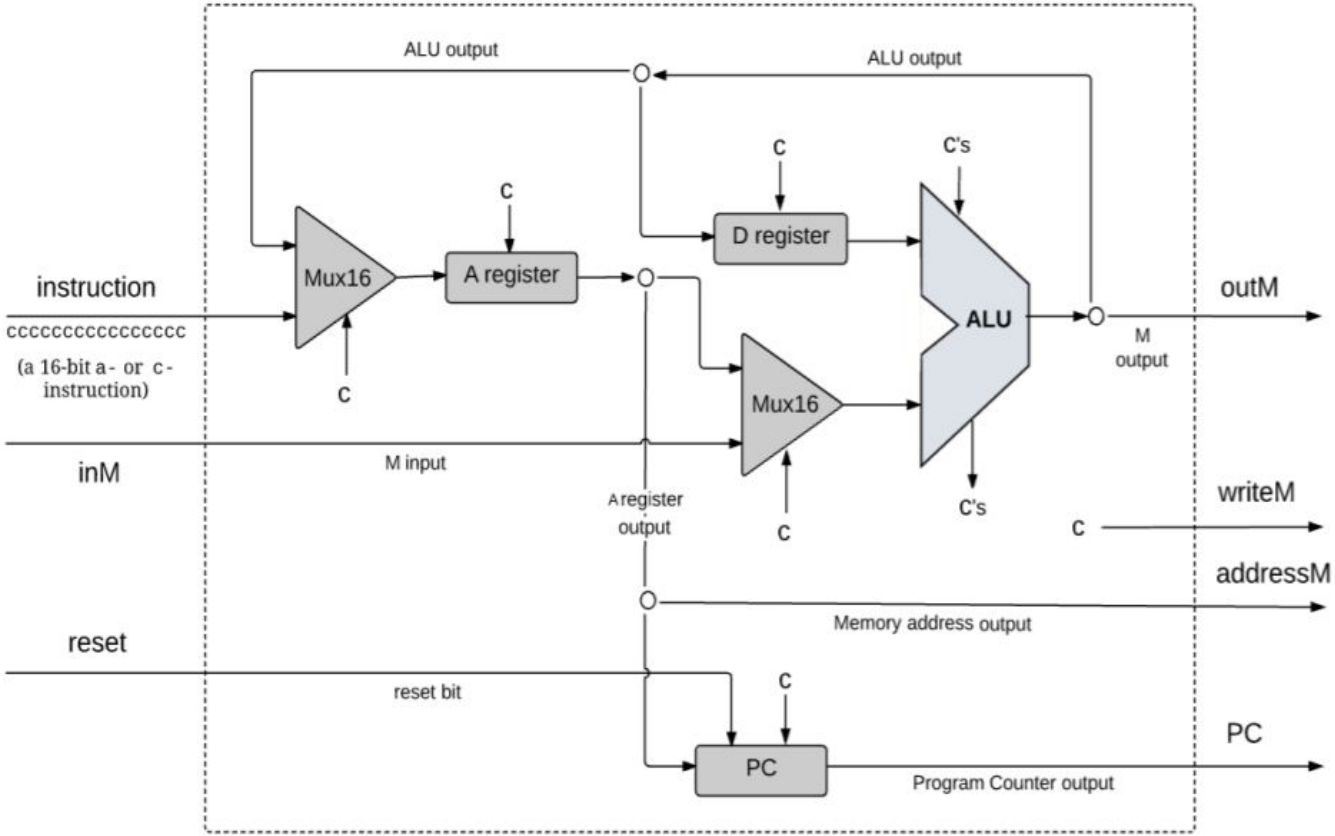
ELEMENTS OF COMPUTING SYSTEM-1

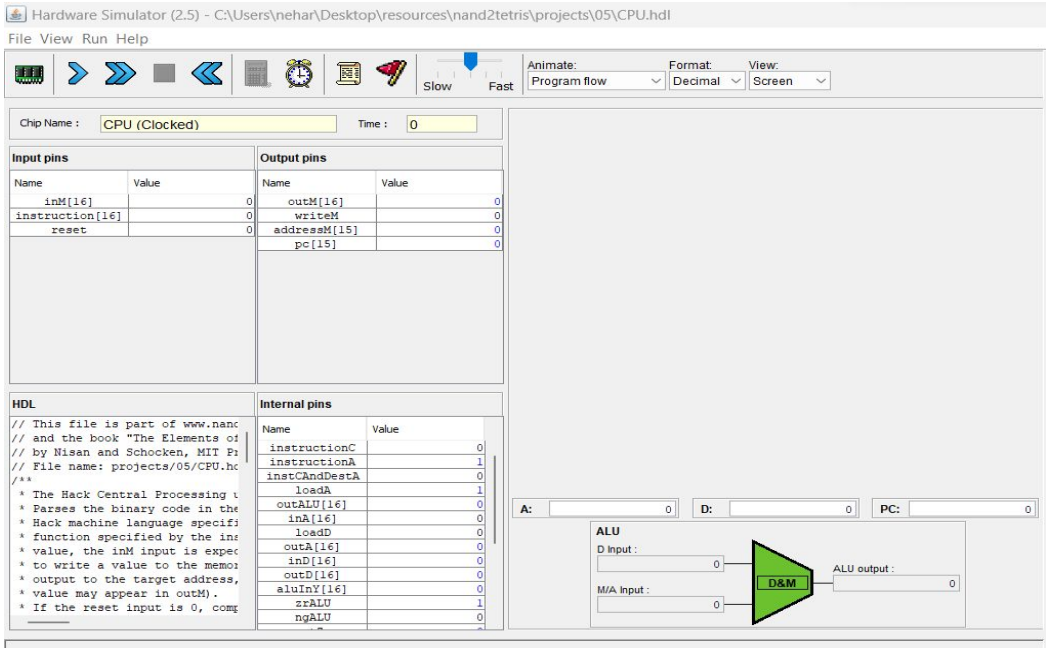
Part A

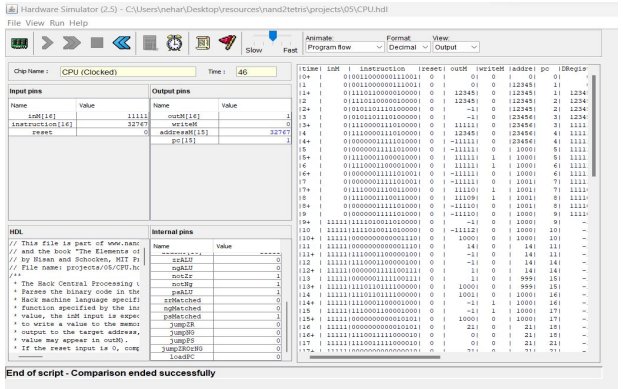
Building of HACK CPU

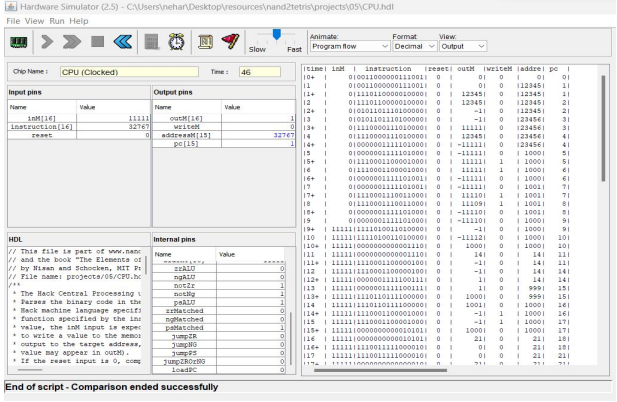
Block Diagram



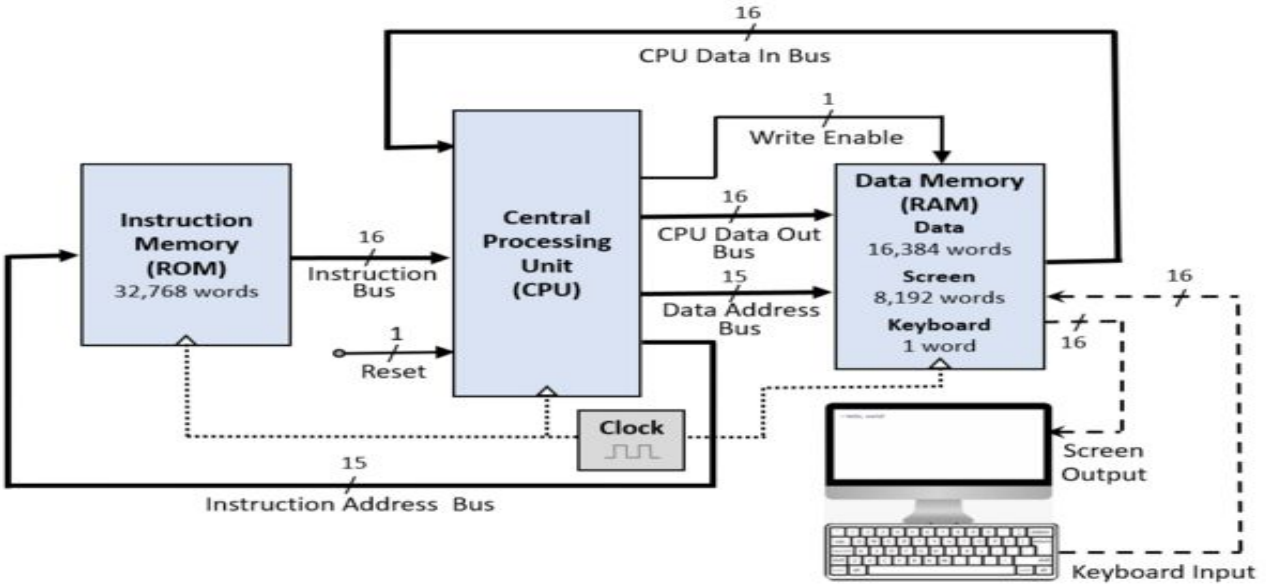
OUTPUT


Hardware Simulator (2.5) - C:\Users\nehar\Desktop\resources\nand2tetris\projects\05\CPU.hdl
File View Run Help
Slow
Fast
Animate:
Program flow
Format:
Decimal
View:
Screen
Chip Name :
CPU (Clocked)
Time :
0
Input pins
Name
Value
inM[16]
0
instruction[16]
0
reset
0
Output pins
Name
Value
outM[16]
0
writeM
0
addressM[15]
0
pc[15]
0
HDL
// This file is part of www.nanc
// and the book "The Elements of
// by Nisan and Schocken, MIT Pr
// File name: projects/05/CPU.hd
/**
* The Hack Central Processing u
* Parses the binary code in the
* Hack machine language specifi
* function specified by the ins
* value, the inM input is expec
* to write a value to the memor
* output to the target address,
* value may appear in outM).
* If the reset input is 0, comp
Internal pins
Name
Value
instructionC
0
instructionA
1
instCAndDestA
0
loadA
1
outALU[16]
0
inA[16]
0
loadD
0
outA[16]
0
inD[16]
0
outD[16]
0
aluInY[16]
0
zrALU
1
ngALU
0
A:
0
D:
0
PC:
0
ALU
D Input :
0
M/A Input :
0
D&M
ALU output :
0

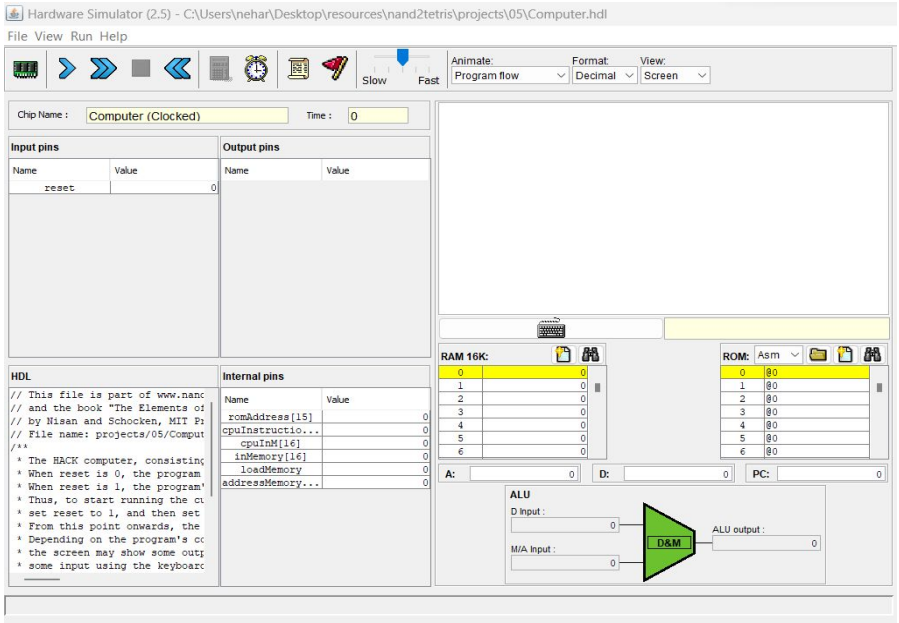

Hardware Simulator (2.5) - C:\Users\nehar\Desktop\resources\nand2tetris\projects\05\CPU.hdl
File View Run Help
Slow
Fast
Animate:
Program flow
Format:
Decimal
View:
Output
Chip Name :
CPU (Clocked)
Time :
46
Input pins
Output pins
HDL
Internal pins
End of script - Comparison ended successfully

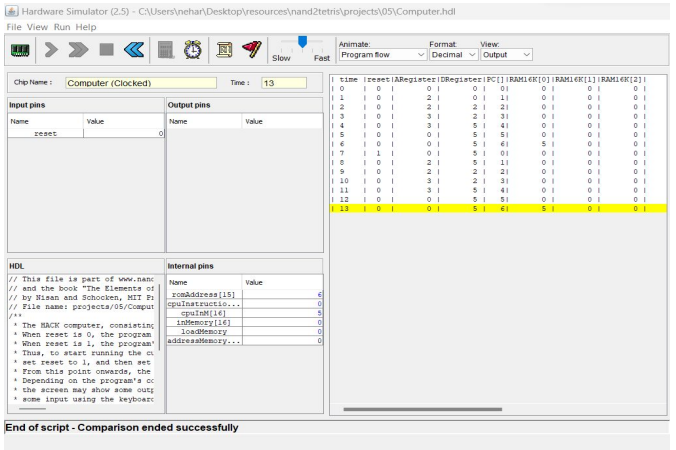

Hardware Simulator (2.5) - C:\Users\nehar\Desktop\resources\nand2tetris\projects\05\CPU.hdl
File View Run Help
Slow
Fast
Animate:
Program flow
Format:
Decimal
View:
Output
Chip Name :
CPU (Clocked)
Time :
46
Input pins
Output pins
HDL
Internal pins
End of script - Comparison ended successfully

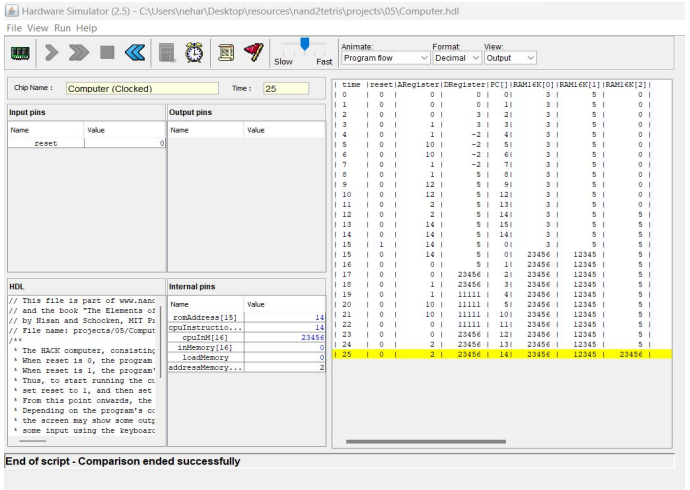
Additional Development -Using CPU in Computer

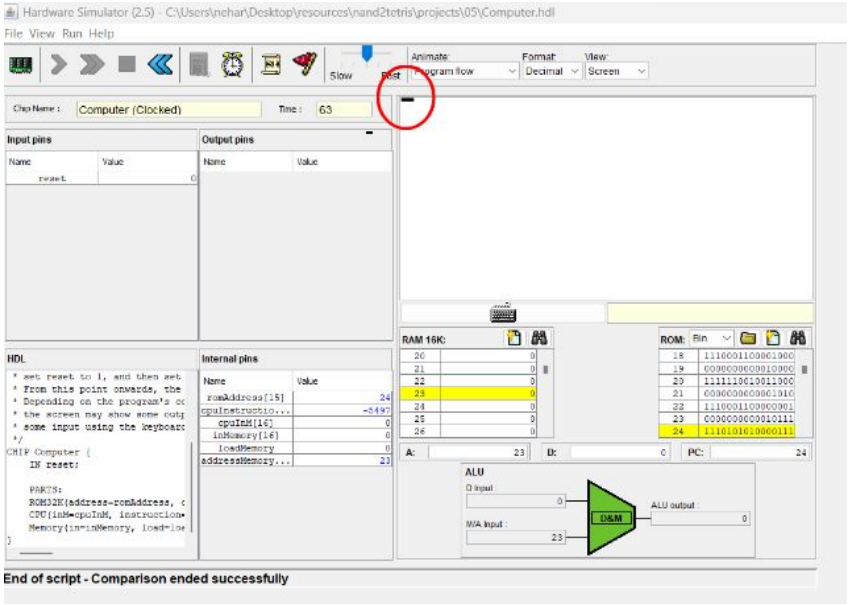


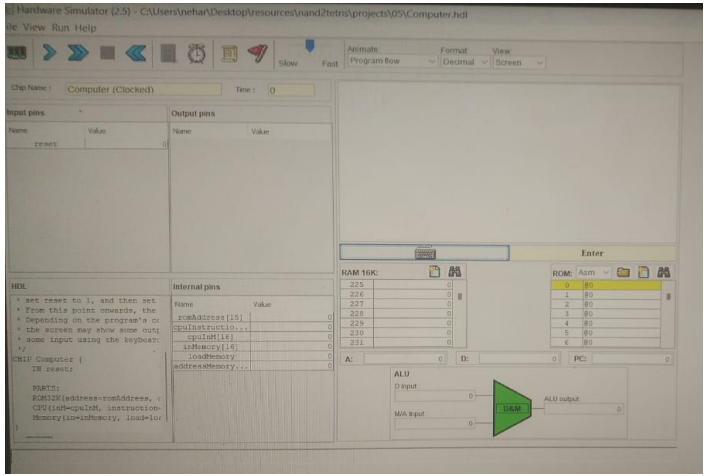
Output


Hardware Simulator (2.5) - C:\Users\nehar\Desktop\resources\nand2tetris\projects\05\Computer.hdl
File View Run Help
Slow
Fast
Animate:
Program flow
Format:
Decimal
View:
Screen
Chip Name :
Computer (Clocked)
Time :
0
Input pins
Name
Value
reset
0
Output pins
Name
Value
HDL
// This file is part of www.nanc
// and the book "The Elements of
// by Nisan and Schocken, MIT Pr
// File name: projects/05/Comput
/**
* The HACK computer, consisting
* When reset is 0, the program
* When reset is 1, the program'
* Thus, to start running the cu
* set reset to 1, and then set
* From this point onwards, the
* Depending on the program's cc
* the screen may show some outp
* some input using the keyboarc
Internal pins
Name
Value
romAddress[15]
0
cpuInstructio...
0
cpuInM[16]
0
inMemory[16]
0
loadMemory
0
addressMemory...
0
RAM 16K:
0
1
2
3
4
5
6
ROM:
Asm
@0
A:
0
D:
0
PC:
0
ALU
D Input :
M/A Input :
D&M
ALU output :


Hardware Simulator (2.5) - C:\Users\nehar\Desktop\resources\nand2tetris\projects\05\Computer.hdl
File View Run Help
Slow
Fast
Animate:
Program flow
Format:
Decimal
View:
Output
Chip Name :
Computer (Clocked)
Time :
13
Input pins
Name
Value
reset
0
Output pins
Name
Value
HDL
Internal pins
Name
Value
romAddress[15]
6
cpuInstructio...
0
cpuInM[16]
5
inMemory[16]
0
loadMemory
0
addressMemory...
0
time	reset	ARegister	DRegister	PC[]	RAM16K[0]	RAM16K[1]	RAM16K[2]
0	0	0	0	0	0	0	0
1	0	2	0	1	0	0	0
2	0	2	2	2	0	0	0
3	0	3	2	3	0	0	0
4	0	3	5	4	0	0	0
5	0	0	5	5	0	0	0
6	0	0	5	6	5	0	0
7	1	0	5	0	0	0	0
8	0	2	5	1	0	0	0
9	0	2	2	2	0	0	0
10	0	3	2	3	0	0	0
11	0	3	5	4	0	0	0
12	0	0	5	5	0	0	0
13	0	0	5	6	5	0	0
End of script - Comparison ended successfully


Hardware Simulator (2.5) - C:\Users\nehar\Desktop\resources\nand2tetris\projects\05\Computer.hdl
File View Run Help
Slow
Fast
Animate:
Program flow
Format:
Decimal
View:
Output
Chip Name :
Computer (Clocked)
Time :
25
Input pins
Name
Value
reset
0
Output pins
Name
Value
HDL
Internal pins
Name
Value
romAddress[15]
14
cpuInstructio...
14
cpuInM[16]
23456
inMemory[16]
0
loadMemory
0
addressMemory...
2
time	reset	ARegister	DRegister	PC[]	RAM16K[0]	RAM16K[1]	RAM16K[2]
0	0	0	0	0	3	5	0
1	0	0	0	1	3	5	0
2	0	0	3	2	3	5	0
3	0	1	3	3	3	5	0
4	0	1	−2	4	3	5	0
5	0	10	−2	5	3	5	0
6	0	10	−2	6	3	5	0
7	0	1	−2	7	3	5	0
8	0	1	5	8	3	5	0
9	0	12	5	9	3	5	0
10	0	12	5	12	3	5	0
11	0	2	5	13	3	5	0
12	0	2	5	14	3	5	5
13	0	14	5	15	3	5	5
14	0	14	5	14	3	5	5
15	1	14	5	0	3	5	5
16	0	0	5	1	23456	12345	5
17	0	0	23456	2	23456	12345	5
18	0	1	23456	3	23456	12345	5
19	0	1	11111	4	23456	12345	5
20	0	10	11111	5	23456	12345	5
21	0	10	11111	10	23456	12345	5
22	0	0	11111	11	23456	12345	5
23	0	0	23456	12	23456	12345	5
24	0	2	23456	13	23456	12345	5
25	0	2	23456	14	23456	12345	23456
End of script - Comparison ended successfully


Hardware Simulator (2.5) - C:\Users\nehar\Desktop\resources\nand2tetris\projects\05\Computer.hdl
File View Run Help
Slow
Fast
Animate:
Program flow
Format:
Decimal
View:
Screen
Chip Name :
Computer (Clocked)
Time :
63
Input pins
Name
Value
reset
Output pins
Name
Value
RAM 16K:
ROM:
Bin
HDL
* set reset to 1, and then set
* From this point onwards, the
* Depending on the program's cc
* the screen may show some outp
* some input using the keyboard
*/
CHIP Computer {
IN reset;
PARTS:
ROM32K(address=romAddress, c
CPU(inM=cpuInM, instruction=
Memory(in=inMemory, load=loa
}
Internal pins
Name
Value
romAddress[15]
24
cpuInstructio...
-5497
cpuInM[16]
0
inMemory[16]
0
loadMemory
0
addressMemory...
23
A:
23
D:
0
PC:
24
ALU
D input :
0
M/A input :
23
D&M
ALU output :
0
End of script - Comparison ended successfully



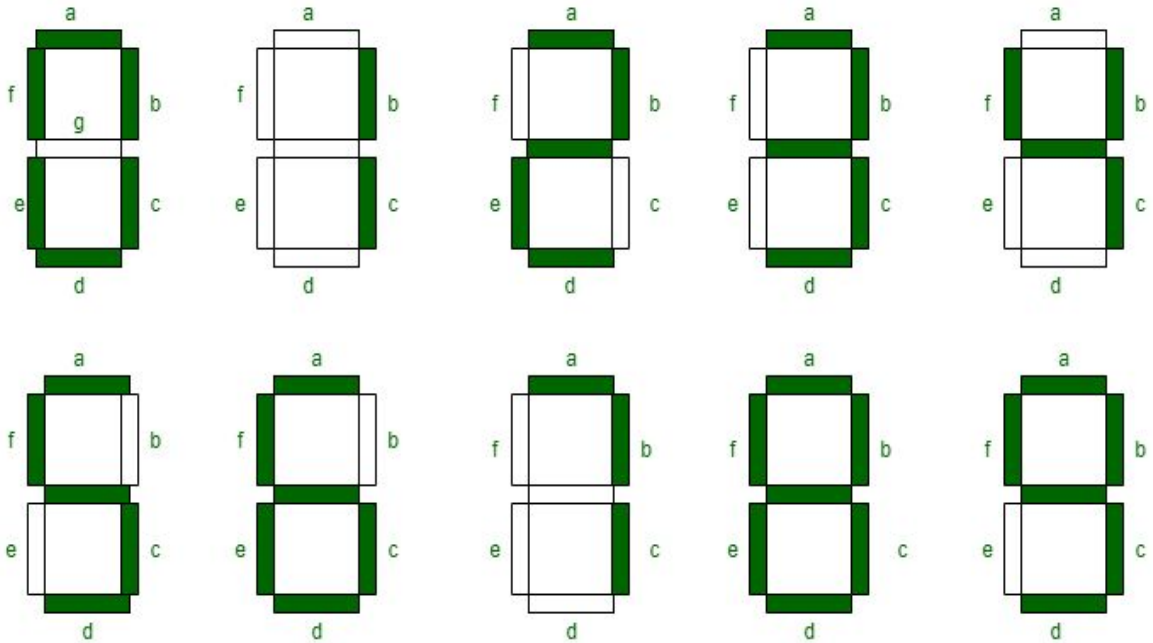
Part B

Design and implement a digital system that takes the Gray code output from a rotary encoder and displays the rotary shaft position on a 7-segment display

What a Rotary Encoder is ?

- Definition: Rotary encoders are electromechanical devices used to convert the angular position of a shaft into an electrical signal.
- Commonly used in various applications for measuring rotation and angular displacement.
- Utilizes optical or magnetic sensors to detect changes in position.
- Optical encoders use light and a patterned disc, while magnetic encoders use a magnetized disc and sensors.

What is a 7 Segment display ?



Seven segment display is an electric component that consists of seven individual LEDs arranged in the form of the number 8 labeled from a to g. The primary function of the seven segment display is to visually represent the decimal numbers from 0 to 9 by illuminating the seven segments.

Proposed project

- Converting Gray Code output of Rotary encoder to display shafts position to display the shaft’s position in a 7 segment display
- Involves 3 basic steps which involves various conversions, which is gray to binary, binary to decimal, decimal to 7 segment display
- Function can be amplified using a microcontroller
- Rigorous testing and further debugging crucial to know its feasibility
- Involves usage of Logic gates for a huge portion

Consider each digit of Gray code as A,B,C,D respectively

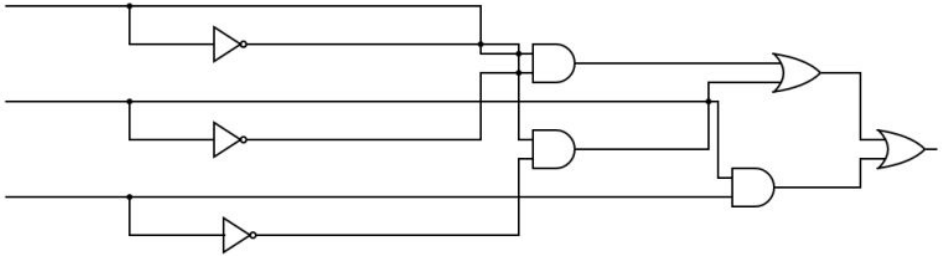
A	B	C	D	a	b	c	d	e	f	g
0	0	0	0	1	1	1	1	1	1	0
0	0	0	1	0	1	1	0	0	0	0
0	0	1	1	1	1	0	1	1	0	1
0	0	1	0	1	1	1	1	0	0	1
0	1	1	0	0	1	1	0	0	1	1
0	1	1	1	1	0	1	1	0	1	1
0	1	0	1	1	0	1	1	1	1	1
0	1	0	0	1	1	1	0	0	0	0
1	1	0	0	1	1	1	1	1	1	1
1	1	0	1	1	1	1	1	0	1	1

ABCD	00	01	11	10
00	1	0	1	1
01	1	1	1	0
11	1	1	X	X
10	X	X	X	X

Involves the usage of multiple k-maps

This can be obtained by the standard algorithm for its solution involving isolation of even number of ones to the maximum extent possible

The above example is for first line of gray code which is being converted to binary



Application of Logic Gates

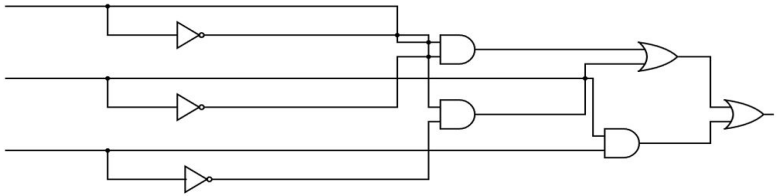
Logic gates can be utilized to interpret the outcomes from Kmaps to understand whole thing easier

The above is logic gate for kmap given in previous slide

For segment ‘a’

The three major lines represent B,C,D

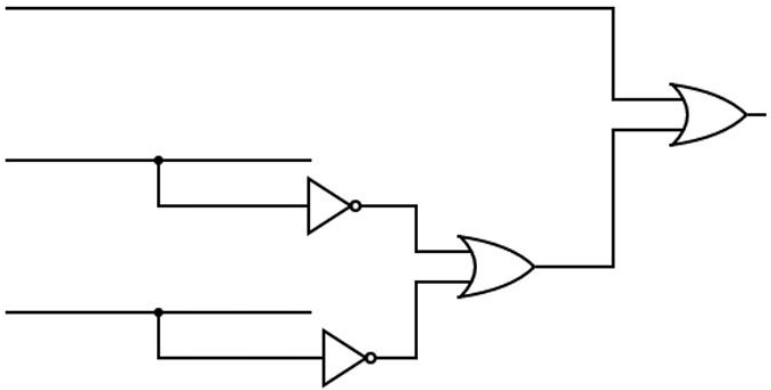
a= BC’+B’D’+CD



For segment ‘b’

The three major lines represents A,B,D

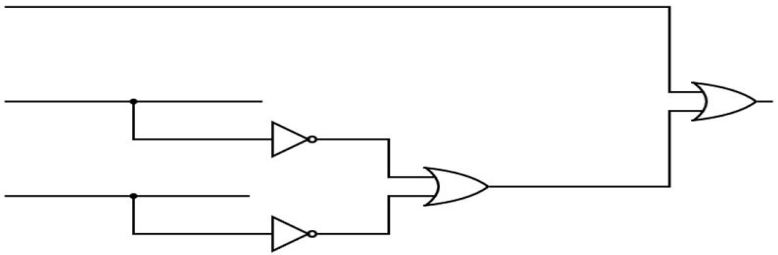
b= C’+D’+B



For segment ‘c’.

The three major lines represents B,C,D.

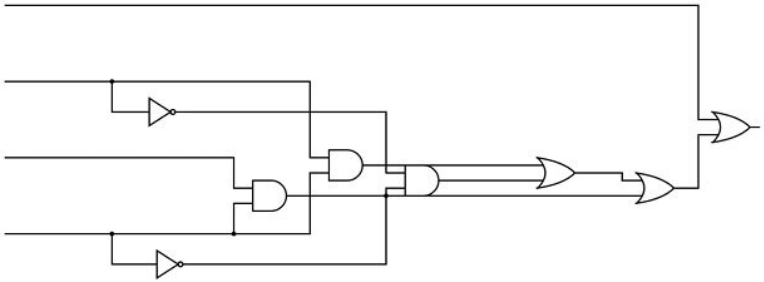
c= C’+D’+B



For segment d

The three major lines represent A,B,C,D.

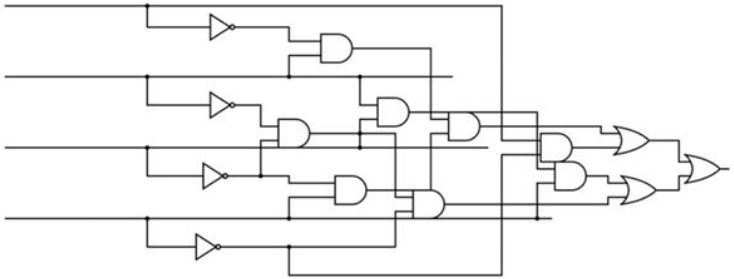
d= A+BD+CD+B'D'



For segment 'e':

The four major lines are A,B,C,D

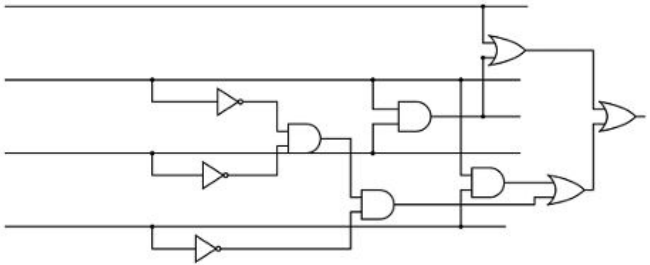
e= A'BC'D+AD'+BCD+B'C'D'



For segment 'f'

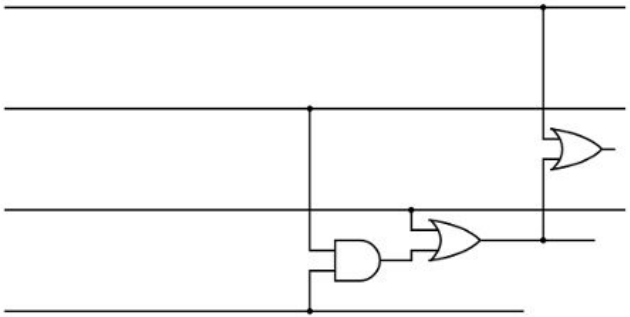
The major lines are A,B,C,D.

f= BD+BC+A+B'C'D

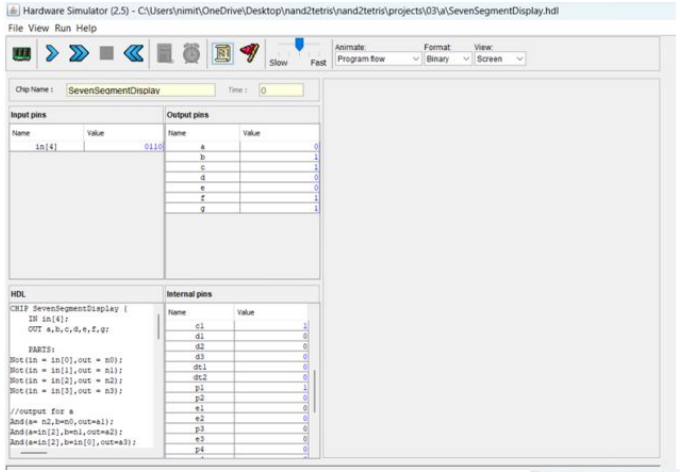


For segment 'g'

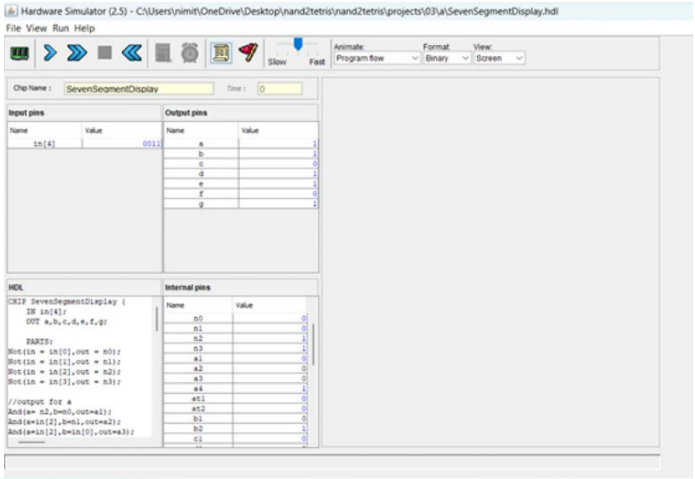
The major lines are A,B,C,D
g= BD+A+C

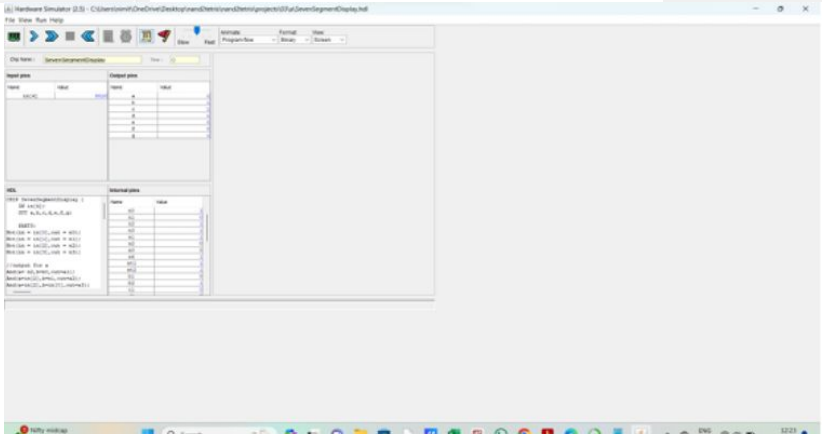


Generating the output



The following three Are the output pages





THANK YOU