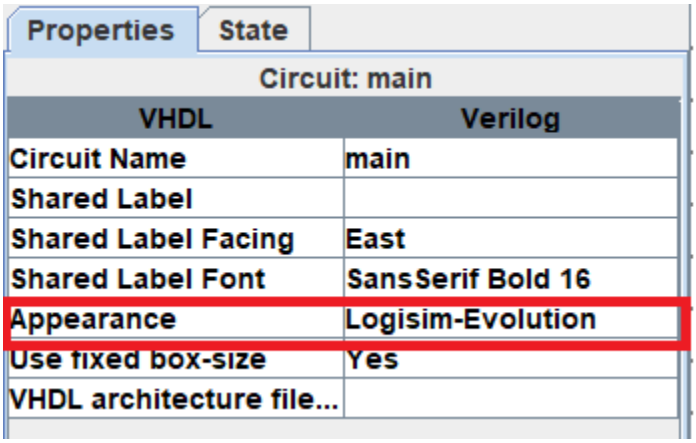
ECS 154A: Homework 4

Logisim

1. Use the version from the class Google Drive of Logisim Evolution. Other versions may not work correctly.
2. Do not rename the files you receive. If you do so you will automatically fail the tester when you submit.
3. Put your solution for each problem into implementation subcircuit
4. Do not rename the implementation subcircuit anything else. If you do so you will automatically fail the tester when you submit.
5. Do not change the appearance of the implementation subcircuit from what it is set as. Doing so will cause you to automatically fail the tester when you submit.
    a. That is this field right here



6. Do not move the pins inside of the implementation subcircuit as that affects the appearance of the circuit on the outside as you saw in discussion. Doing so will cause you to automatically fail the tester when you submit.
    a. If you want to “move the pins” instead connect tunnels to the pins and move the tunnels around.
7. Do not name any of the subcirucits in your solution main. Doing so will cause you to automatically fail the tester when you submit.
8. You **can** create as many other subcircuits as you want in your solution. Just make sure your solution ends up in the implementation subcircuit

Restrictions

For all problems in this homework, you may only use

- All of the components under Wiring
- All of the components under Gates **EXCEPT** for Controlled Buffer, Controlled Inverter, PLA
- All of the components under Plexers
- All of the components under Arithmetic
- All of the components under Memory **EXCEPT** for **RAM, ROM,** and **Random** Generator

Unless a problem specifies otherwise.

You have been provided a Register File circuit in the starting circuit. The **only outputs** of the register file that you are allowed to use are A_Out and B_Out. The rest are for testing purposes and **should not be used.** Using them will result in a 50% penalty in your grade.

Problem 1: CPU.circ (100 points)

Build a 4-bit single cycle CPU that can implement the given instructions.

Instruction Format

Our CPU will be using fixed length instructions. Our CPU will also have two types of instruction formats: R-type and I-type. In R-type instructions both operands come from registers. In I-type instructions, the first operand comes from a register and the second will be contained within the instruction.

R-Type

Name	Bits	Description
OpCode	15 - 12	Determines what operation should be performed
C	11 - 8	The destination register. The C in $Reg_C = Reg_A$ OP Reg_B
A	7 - 4	The first source register. The A in $Reg_C = Reg_A$ OP Reg_B
B	3 - 0	The second source register. The B in $Reg_C = Reg_A$ OP Reg_B

I-Type

Name	Bits	Description
OpCode	15 - 12	Determines what operation should be performed
C	11 - 8	The destination register. The C in $Reg_C = Reg_A$ OP Imm
A	7 - 4	The first source register. The A in $Reg_C = Reg_A$ OP Imm
Immediate	3 - 0	The second source register. The Imm in $Reg_C = Reg_A$ OP Imm

Instructions

Operation	Encoding (The value in the OpCodeField)	Description
STOP	0000	The CPU ceases execution
NOP	0001	Do nothing
LOAD	0010	Reg_C = Immediate
MOVE	0011	$Reg_C = Reg_A$
ANDR	0100	$Reg_C = Reg_A$ AND Reg_B
ANDI	0101	$Reg_C = Reg_A$ AND Immediate
ORR	0110	$Reg_C = Reg_A$ OR Reg_B
ORI	0111	$Reg_C = Reg_A$ OR Immediate
XORR	1000	$Reg_C = Reg_A$ XOR Reg_B
XORI	1001	$Reg_C = Reg_A$ XOR Immediate
NOT	1010	Reg_C = NOT Reg_A
NEGATE	1011	$Reg_C = -Reg_A$
ADDR	1100	$Reg_C = Reg_A + Reg_B$
ADDI	1101	$Reg_C = Reg_A$ +Immediate
SUBR	1110	$Reg_C = Reg_A - Reg_B$
SUBI	1111	$Reg_C = Reg_A$ - Immediate

Inputs

Pin	Size (in bits)	Explanation
Instruction	16	The instruction located at Instruction_Address
ClkIn	1	The Clock. Connect this to the clock ports of your registers/flip-flops. Do nothing else with this.

Outputs

Pin	Size (in bits)	Explanation
Instruction_Address_Out	5	The address of the instruction you want to execute
Reg0-15	4	The values in the register file. This has already been connected for you

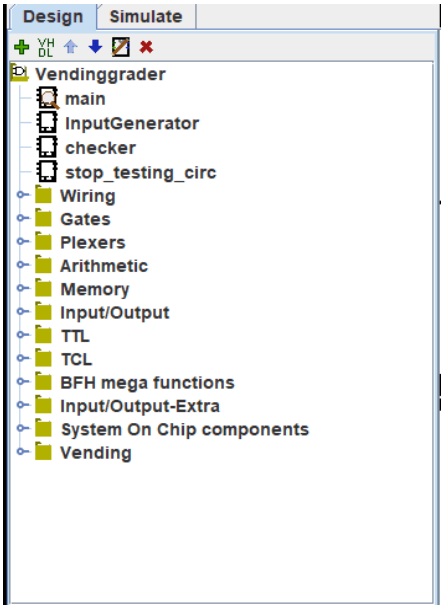
CPU Components

Your CPU should have

- A Program Counter (PC)
    - This stores and keeps track of what instruction you are on
- Instruction Decoder
    - This is a bunch of combinational logic that sets the control signals inside of your CPU
- Register File
    - A bunch of registers as well as ways to specify which ones you want. This has already been created for you.
    - The **only outputs** of the register file that you are allowed to use are A_Out and B_Out. The rest are for testing purposes and **should not be used.** Using them will result in a 50% penalty in your grade.

Testing

Testing for this problem is different than for previous assignments but is more similar to sequential circuits than combinational circuits. After you finish building your circuit and are ready to test it

1. Open the associated grader circuit
2. Scroll down on the left and side until you find your circuit. Right-click on it and select Reload Library
   a. 

3. Use cntrl+t to tick the clock and check that the outputs of the registers are what they are supposed to be as based on the test program below.

The Test Program

Instruction	Meaning	Result
LOAD REG_0, 3	Reg_0 = 3	Reg_0 = **3**
LOAD REG_1,6	Reg_1 = 6	Reg_1 = **6**
NOP	Do Nothing	No Change
MOVE REG_2, Reg1	Reg_2 = Reg_1	Reg_2 = **6**
ANDR REG_3, REG_0, REG_1	REG_3 = REG_0 AND REG_1	REG_3 = 3 & 6 = **2**
ANDI REG_4, REG_3,3	REG_4 = REG_3 AND 3	REG_4 = 2 & 3= **2**
ORR REG_5, Reg_2, REG_0	REG_5 = Reg_2 OR REG_0	REG_5 = 6 |3 = **7**
ORI REG_6, Reg_3, 12	REG_6 = Reg_3 OR 12	REG_6 = 2 OR 12 = **14**
XORR REG_7, Reg_2, REG_0	REG_7 = Reg_2 XOR REG_0	REG_7 = 6 ^ 3 = **5**
XORI REG_8, Reg_6, 15	REG_8 = Reg_6 XOR 15	REG_8 =14 ^15 = **1**
NEG REG_9, REG_3	REG_9 = -REG_3	REG_9 = **-2**
ADDR REG_{10}, Reg_7, REG_7	REG_{10} = Reg_7 + REG_7	REG_{10} = 5 + 5 = **10**
ADDI REG_{11}, Reg_1, 3	REG_{11} = Reg_1 + 3	REG_{11} = 6 + 3 = **9**
SUBR REG_{12}, Reg_6, REG_2	REG_{12} = Reg_6 - REG_2	REG_{12} = 14 - 6 = **8**
SUBI REG_{13}, Reg_8, 5	REG_{13} = Reg_8 - 5	REG_{13} = 1 - 5 = **-4**
NOT REG_{15}, REG_8	REG_{15} = NOT REG_8	REG_{15} = ~1 = **14**
ANDR REG_1, REG_{12}, REG_{13}	REG_1 = REG_{12} AND REG_{13}	REG_1 = 8 & 12 = **8**
NEG REG_5, REG_5	REG_5 = -REG_5	REG_5 = **-7**
NOP	Do Nothing	No Change
ADDR REG_{14}, REG_5, REG_2	REG_{14} = Reg_5 + REG_2	REG_{14} = -7 + 6 = **-1**
XORR REG_7, REG_7, REG_{14}	REG_7 = Reg_7 ^ REG_{14}	REG_7 = 5 ^ -1 = **10**
ORI REG_2, REG_2, 3	REG_2 = Reg_2 | 3	REG_2 = 6 | 3 = **7**
SUBI REG_{13}, REG_{13}, 12	REG_{13} = REG_{13} - 12	REG_{13} = 12 - 12 = **0**
STOP	CPU should cease execution	CPU should cease execution

LOAD REG_{15}, 13	This line should not be executed because the CPU should have STOPped already	This line should not be executed because the CPU should have STOPped already

The test program is just an **example program.** Your CPU should function on **any program** given to it.

Making Fixes to Your Solution

After you make changes to your solution you will need to reload your circuit in the grader circuit. If you don't it won't see the updates. To reload your circuit, select your circuit in the grader, right-click it and select Reload Library.

Submitting

Submit to

Logisim Homework 3 on GradeScope.

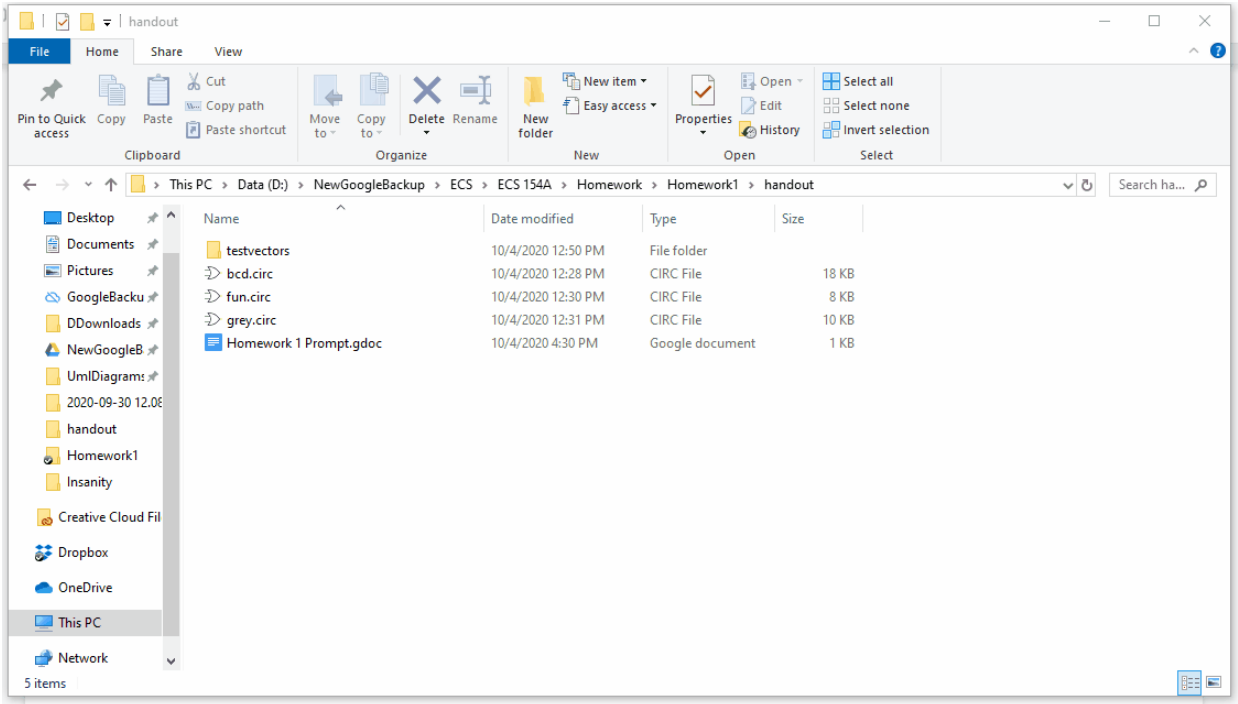
What to Submit

Submit a zip file that contains the following .circ files

1. CPU.circ

Inside of each .circ file leave a comment with you and your partner's names.

Make sure that you submit a zip that contains the files and **NOT** the **folder containing the files.** Check out the animation below for what to submit.


handout
File
Home
Share
View
Pin to Quick access
Copy
Paste
Cut
Copy path
Paste shortcut
Clipboard
Move to
Copy to
Delete
Rename
Organize
New folder
New item
Easy access
New
Properties
Open
Edit
History
Open
Select all
Select none
Invert selection
Select
This PC › Data (D:) › NewGoogleBackup › ECS › ECS 154A › Homework › Homework1 › handout
Search ha...
Desktop
Documents
Pictures
GoogleBacku
DDownloads
NewGoogleB
UmlDiagrams
2020-09-30 12.08
handout
Homework1
Insanity
Creative Cloud Fil
Dropbox
OneDrive
This PC
Network
Name
Date modified
Type
Size
testvectors
10/4/2020 12:50 PM
File folder
bcd.circ
10/4/2020 12:28 PM
CIRC File
18 KB
fun.circ
10/4/2020 12:30 PM
CIRC File
8 KB
grey.circ
10/4/2020 12:31 PM
CIRC File
10 KB
Homework 1 Prompt.gdoc
10/4/2020 4:30 PM
Google document
1 KB
5 items