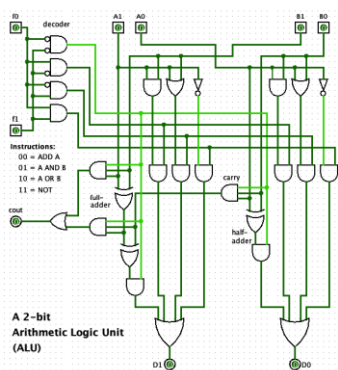


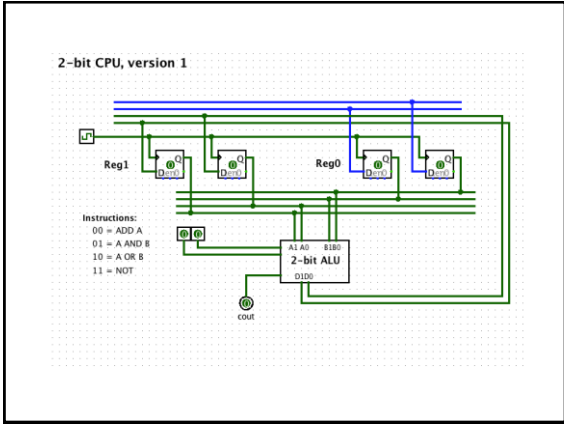
Digital Logic VI: Designing a 2-bit CPU

CMSC 313
Sections 01, 02

2-bit CPU: Version 1

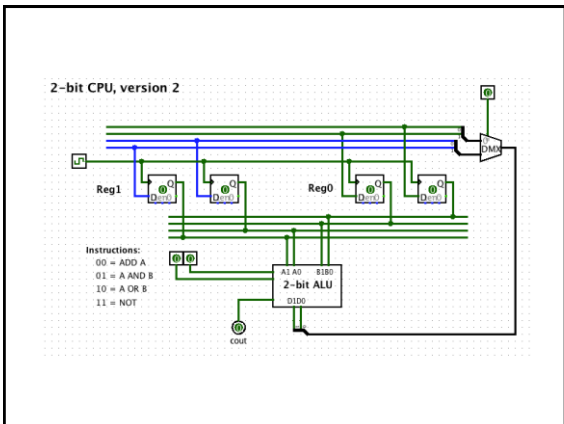
- 2-bit ALU in sub-circuit
- Connect two 2-bit registers to 2-bit ALU
- Output of ALU stored in Register 1





2-bit CPU: Version 2

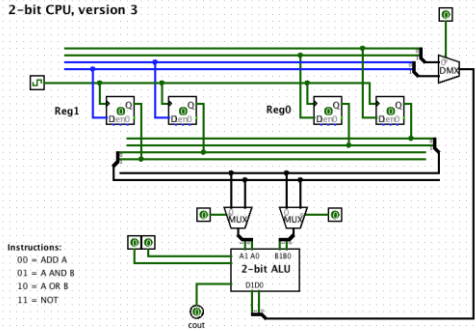
- Use DEMUX to select destination register
- Use Logisim wire bundles



2-bit CPU: Version 3

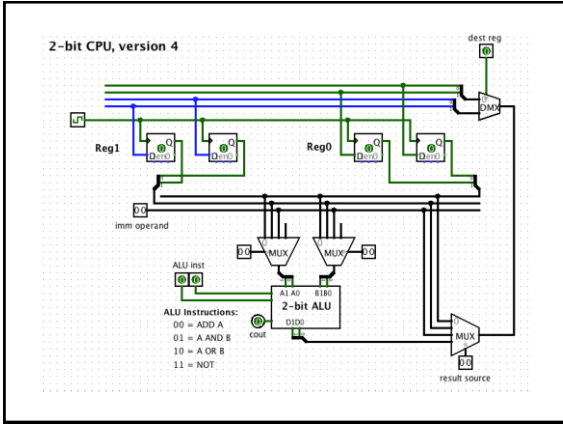
Use MUX to select input to each ALU "port".

2-bit CPU, version 3



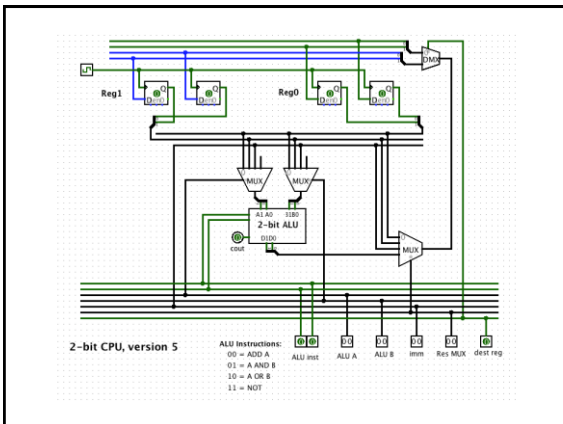
2-bit CPU: Version 4

- Simplify "data bus" using wire bundles
- Add immediate operand to data bus
- Use result MUX to select input to DEMUX for destination register. Input may be:
 - Register 0
 - Register 1
 - Immediate Operand
 - ALU output



2-bit CPU: Version 5

Consolidate controls to a "control bus"



2-bit CPU: Version 6

Use 8-bit "instruction code"

i7	i6	i5	i4	i3	i2	i1	i0
0							

- i7: 0 if ALU instruction, 1 otherwise
- i6 i5: ALU instruction
- i4: operand 1 register (Reg 0 or Reg 1)
- i3 i2 i1: 0rx = operand 2 is Reg r
1xy = immediate operand xy
- i0: destination register

2-bit CPU: Version 6

Use 8-bit "instruction code"

i7	i6	i5	i4	i3	i2	i1	i0
1	0	0	0				

- i7: 0 if ALU instruction, 1 otherwise
- i6 i5 i4: 000 = move, others not implemented
- i3 i2 i1: 0rx = source operand is Reg r
1xy = immediate operand xy
- i0: destination register

Instruction Decoder

MUX for ALU port B

$$B1 = i3$$

$$B0 = \overline{i3} i2 \overline{i1} + \overline{i3} i2 i1$$

$$= \overline{i3} i2$$

i3	i2	i1	B1	B0	
0	0	0	0	0	} Reg 0
0	0	1	0	0	
0	1	0	0	1	} Reg 1
0	1	1	0	1	
1	0	0	1	0	} Imm
1	0	1	1	0	
1	1	0	1	0	
1	1	1	1	0	

Instruction Decoder

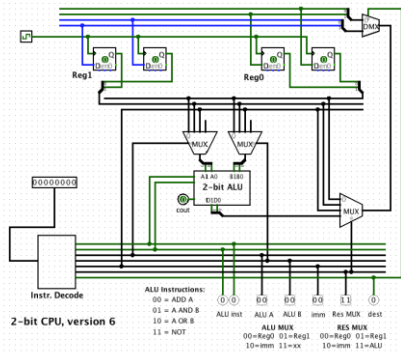
Result MUX control

$$M1 = \overline{i7} + i3$$

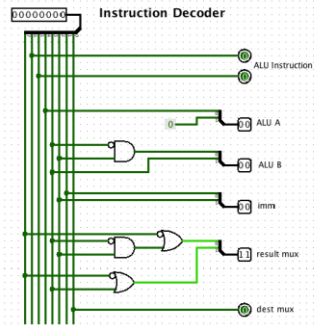
$$M0 = \overline{i7} + \overline{i3} + i2$$

i7	i3	i2	i1	M1	M0
0	0	0	0	1	1
0	0	0	1	1	1
0	0	1	0	1	1
0	0	1	1	1	1
0	1	0	0	1	1
0	1	0	1	1	1
0	1	1	0	1	1
0	1	1	1	1	1
1	0	0	0	0	0
1	0	0	1	0	0
1	0	1	0	0	1
1	0	1	1	0	1
1	1	0	0	1	0
1	1	0	1	1	0
1	1	1	0	1	0
1	1	1	1	1	0

ALU
Reg0
Reg1
Imm

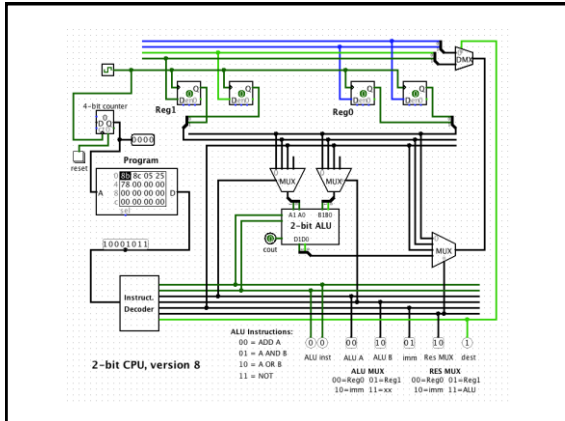


2-bit CPU, version 6



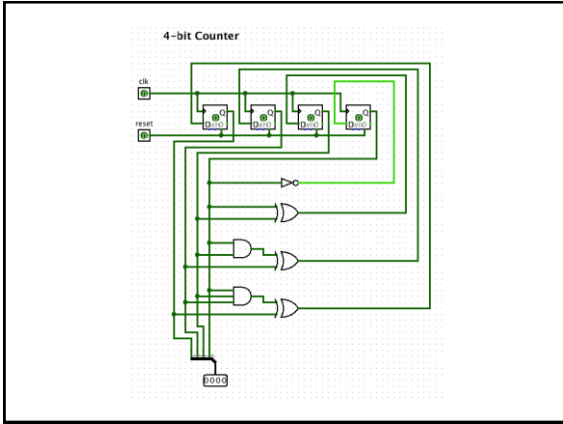
2-bit CPU: Version 8

Added 4-bit counter which automatically advances Program ROM to next instruction.



2-bit CPU: Version 9

Implement 4-bit counter from scratch.



2-bit CPU: Version 10

Implement Program ROM from scratch.

