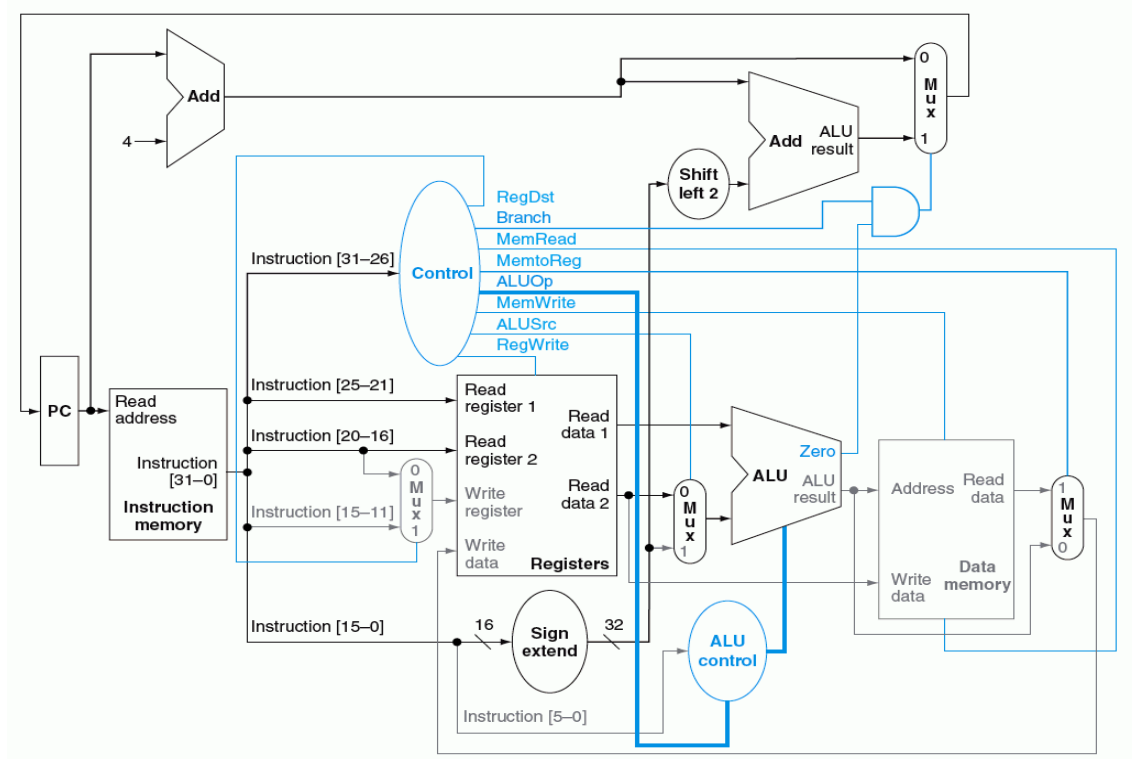
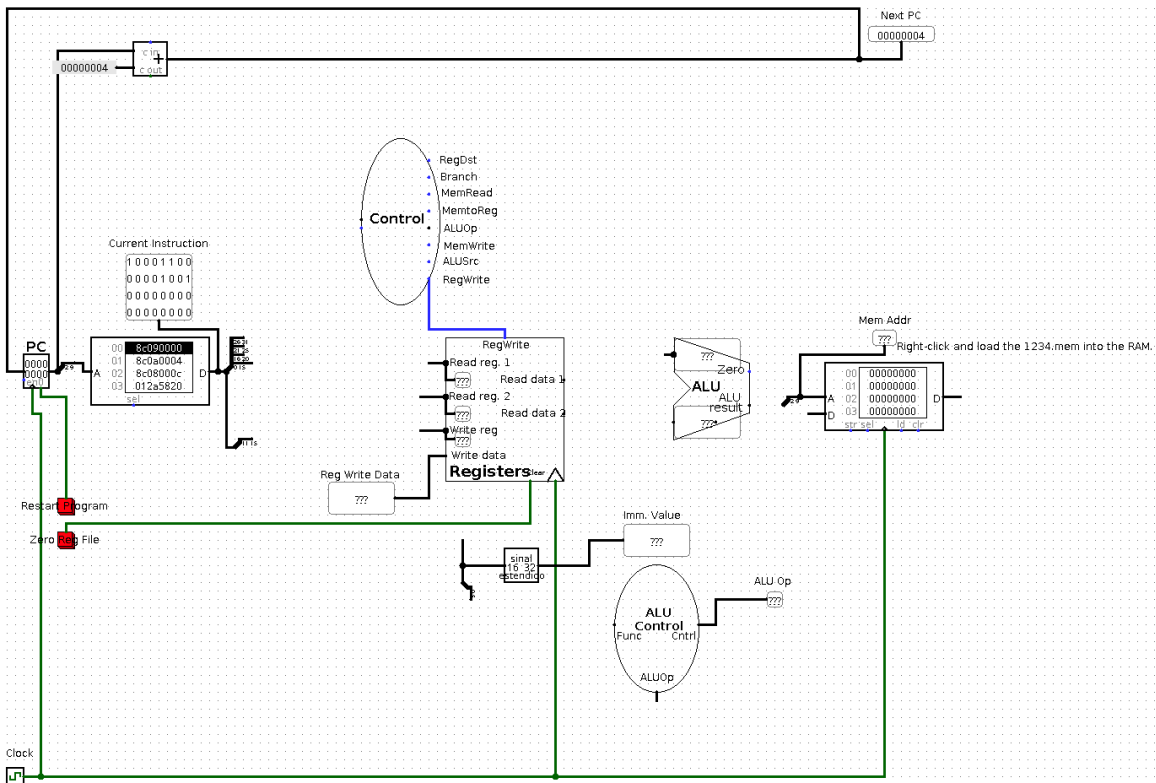


# Criando um Processador MIPS

Processador do nosso livro texto:



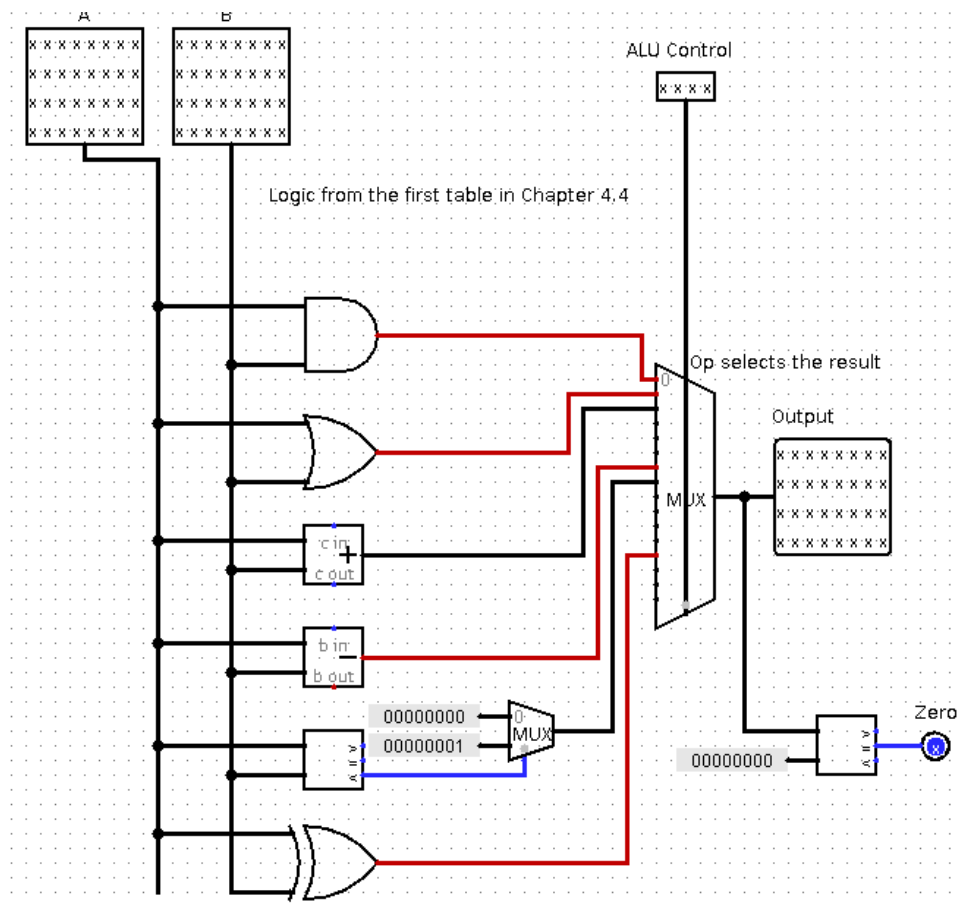
O que temos no logisim:



**Primeiro passo:** Verifique a similaridade entre os caminhos de dados. Estude o caminho de dados do MIPS visto em sala de aula. Sem o seu entendimento, você não conseguirá realizar a criação do seu caminho de dados no LOGISIM. Verifique todas as linhas de controle e o funcionamento das unidades de controle.

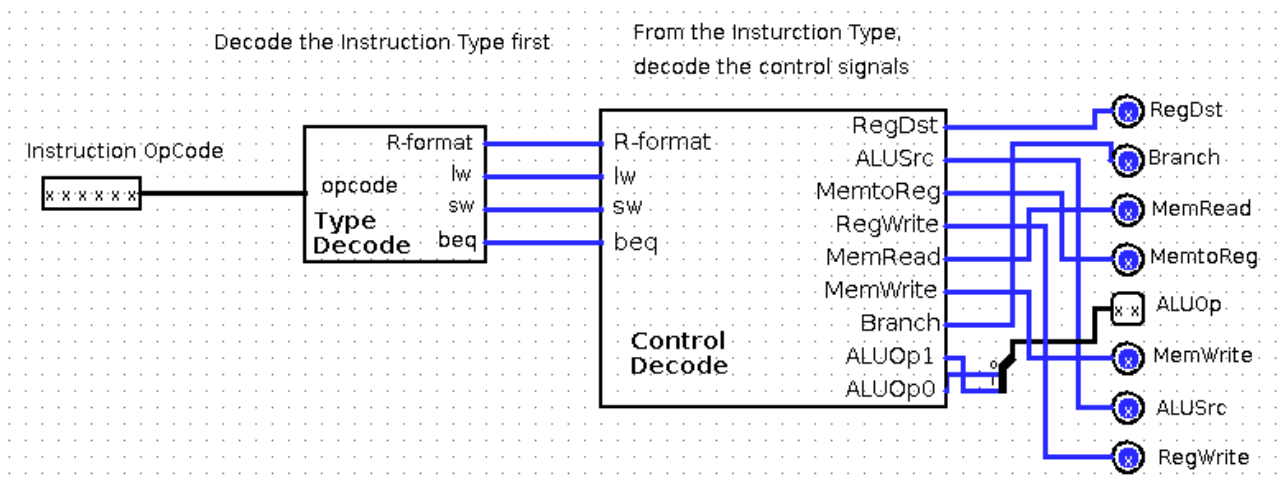
**Segundo passo:** Verifique todos os componentes do projeto inicial no LOGISIM. Inicie com a ALU. Clique com o botão direito em cima do desenho da ALU e depois em “ver alu”. Observe o funcionamento da ALU e como o controle de qual operação a ser realizada é feito. A ALU possui duas entradas (A e B) e calcula uma de 6 funções. Toas as 6 funções são calculadas, mas o MUX seleciona apenas uma para ir para a saída. A seleção é feita pela operação da ALU. Se o controle da ALU for 0010=2, a entrada 2 será selecionada para ir para a saída. A operação 0010 indica a soma entre  $A + B$ .

Compare o desenho com a seguinte tabela de controle da ALU:



Controle da ALU	Operação
0000	AND
0001	OR
0010	ADD
0110	SUB
0111	Set on less than
1100	NOR

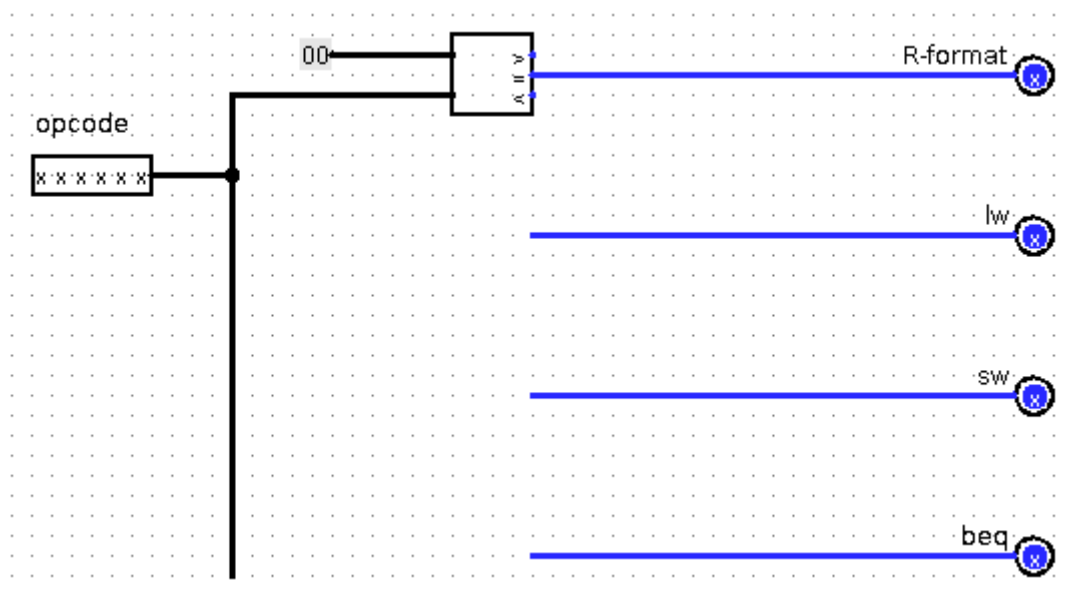
Agora volte para o projeto principal (clique duplo no “mips” no menu do lado esquerdo do Logisim) e verifique a unidade de controle.



A unidade de controle possui várias partes:

- 1) O OpCode como entradas
- 2) “Type Decode” que a partir do OpCode verifica se a instrução é do tipo R-format, lw, sw ou beq.
- 3) “Control Decode” que recebe o tipo da instrução e cria os sinais de controles corretos para executar a referida instrução.

**Tarefa 1:** Finalizar a lógica combinacional para realizar o “Type Decode”.



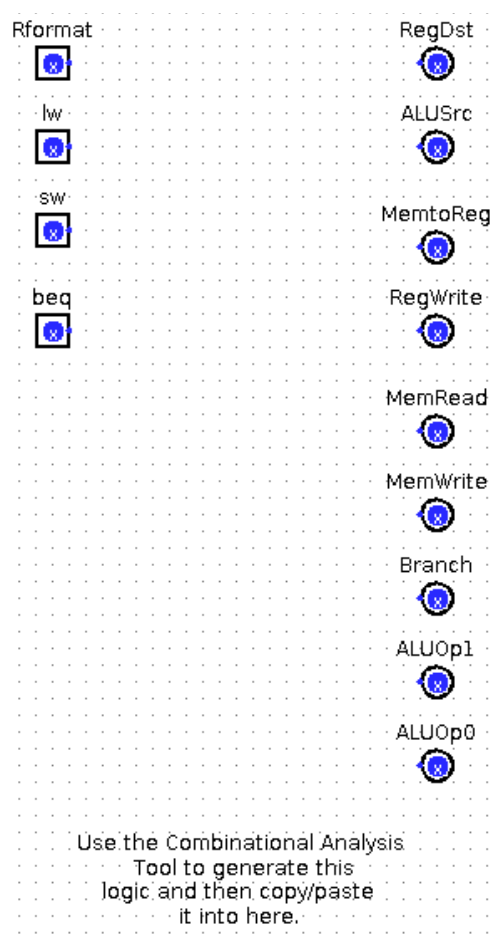
Verifique que foi utilizado um “comparador” e uma constante para verificar se a instrução é do tipo R-format.

Entrada	R-format	Lw	Sw	Beq
Op5	0	1	1	0
Op4	0	0	0	0
Op3	0	0	1	0
Op2	0	0	0	1
Op1	0	1	1	0
Op0	0	1	1	0

Complete a unidade “Type Decoder” adicionando mais comparadores para detectar as instruções corretas.

**Verificação:** Ao finalizar o circuito você precisa testá-lo. Para fazer isso, vá no menu “Simular” do logisim e habilite “Habilitar Simulação”. Utilize o indicador para modificar a entrada do opcode e verifique as saídas. Faça isso com calma para encontrar todos os possíveis errors.

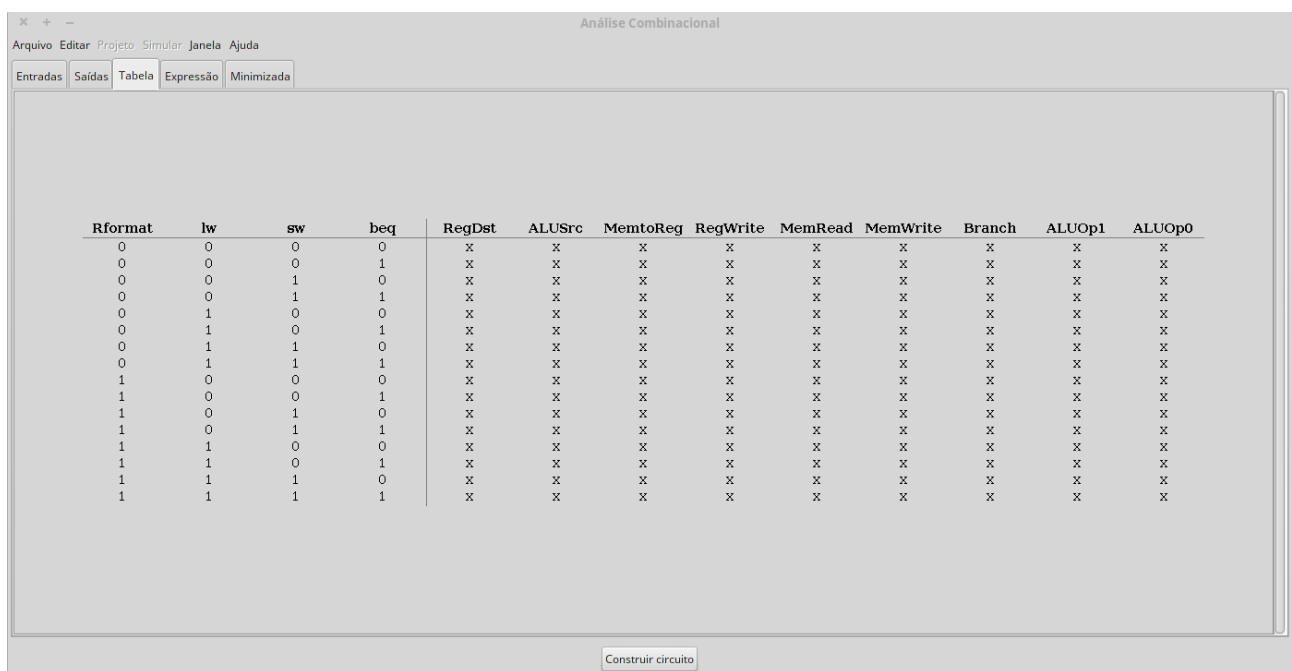
**Tarefa 2:** Finalizar a unidade de controle.



Para criar a lógica combinacional para a unidade de controle, primeiro você deve entender e preencher a tabela abaixo com os seguintes sinais de controle:

Saída	R-format	Lw	Sw	Beq
RegDest	1	0	X	X
ALUSrc				
MemToReg				
RegWrite				
MemRead				
MemWrite				
Branch				
ALUOp1				
ALUOp2				

Após preencher a tabela, utilize a Ferramenta “Combinational Analysys” para gerar o circuito correspondente. Para utilizar essa ferramenta, vá no menu esquerdo e clique com o botão direito no projeto “control-decode” e depois em Analisar Circuito. A seguinte tela irá aparecer:



The screenshot shows the 'Análise Combinacional' window with the 'Tabela' tab selected. It displays a truth table with 16 rows and 13 columns. The columns are labeled: Rformat, lw, sw, beq, RegDst, ALUSrc, MemtoReg, RegWrite, MemRead, MemWrite, Branch, ALUOp1, and ALUOp0. The rows represent all possible combinations of the four control signals (Rformat, lw, sw, beq) from 0 to 15. The values for the other signals are marked with 'X'.

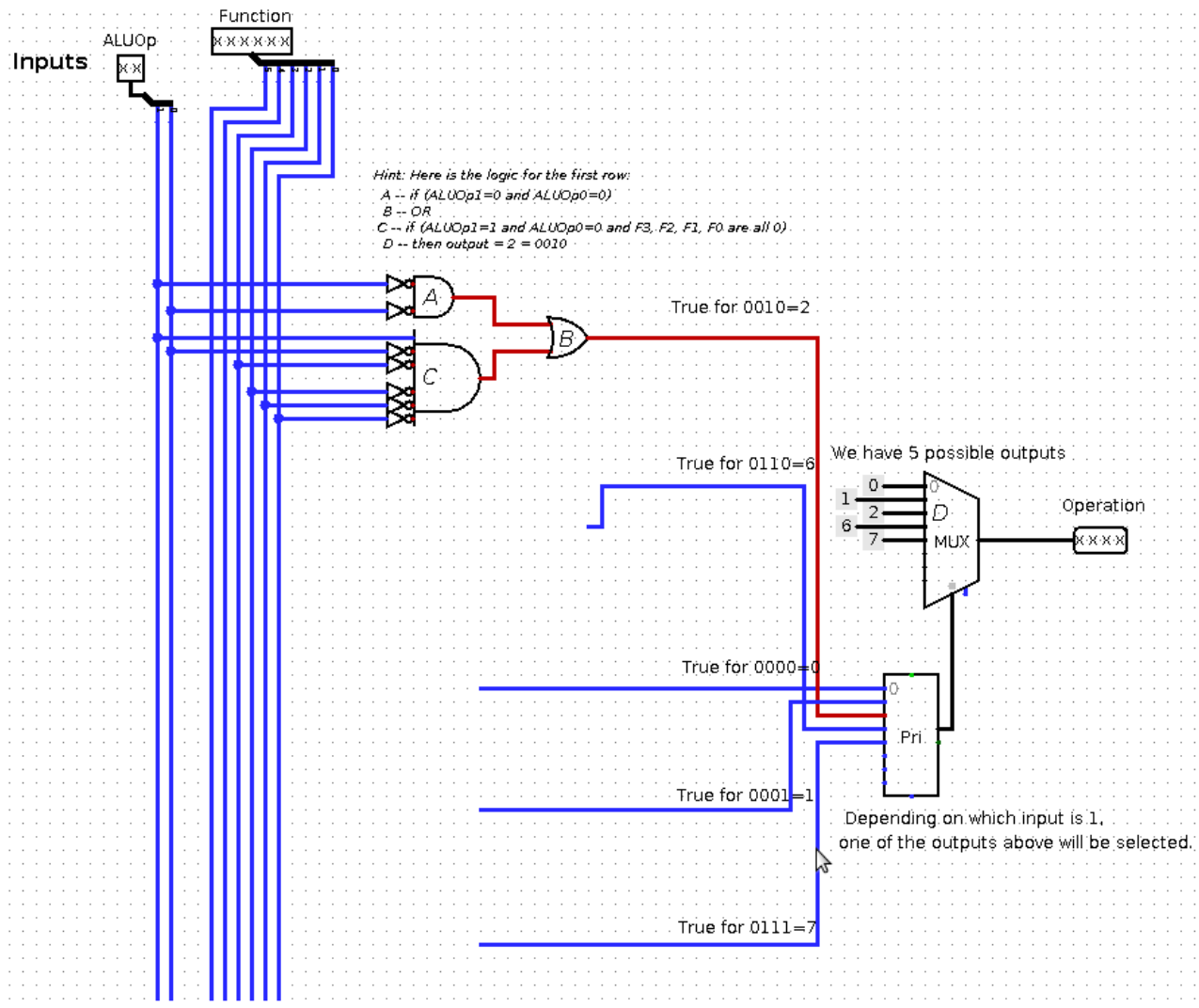
Rformat	lw	sw	beq	RegDst	ALUSrc	MemtoReg	RegWrite	MemRead	MemWrite	Branch	ALUOp1	ALUOp0
0	0	0	0	X	X	X	X	X	X	X	X	X
0	0	0	1	X	X	X	X	X	X	X	X	X
0	0	1	0	X	X	X	X	X	X	X	X	X
0	0	1	1	X	X	X	X	X	X	X	X	X
0	1	0	0	X	X	X	X	X	X	X	X	X
0	1	0	1	X	X	X	X	X	X	X	X	X
0	1	1	0	X	X	X	X	X	X	X	X	X
0	1	1	1	X	X	X	X	X	X	X	X	X
1	0	0	0	X	X	X	X	X	X	X	X	X
1	0	0	1	X	X	X	X	X	X	X	X	X
1	0	1	0	X	X	X	X	X	X	X	X	X
1	0	1	1	X	X	X	X	X	X	X	X	X
1	1	0	0	X	X	X	X	X	X	X	X	X
1	1	0	1	X	X	X	X	X	X	X	X	X
1	1	1	0	X	X	X	X	X	X	X	X	X
1	1	1	1	X	X	X	X	X	X	X	X	X

Verifique na Aba Entrada e Saída as entradas e saídas do circuito. A aba Tabela, preencha a tabela verdade com os dados da tabela de controle acima preenchida por você. Ao final, clique em construir circuito.

**Verificação:** Uma vez construído o circuito, você deverá testá-lo. Faça da mesma forma como fizemos na aula anterior. Verifique se todos os sinais gerados estão de acordo com o que foi projetado. Não esqueça de conferir se o que você projetou está realmente correto.

### Tarefa 3: Finalizar o Controle da ALU.

Verifique a unidade de controle da ALU. Ela possui 8 entradas divididas em: i) 2 entradas para ALUOp e 6 entradas provenientes do campo “function” da instrução. Verifique as entradas você mesmo.



Tente entender o funcionamento da unidade de controle. Não comece a implementá-la sem antes entender seu funcionamento. Utilize a tabela abaixo para entender o circuito e seu objetivo.

ALUOp1	ALUOp0	F5	F4	F3	F2	F1	F0	Operation
0	0	X	X	X	X	X	X	0010
0	1	X	X	X	X	X	X	0110
1	0	X	X	0	0	0	0	0010
1	X	X	X	0	0	1	0	0110
1	0	X	X	0	1	0	0	0000
1	0	X	X	0	1	0	1	0001
1	X	X	X	1	0	1	0	0111

Note as constantes (0, 1, 2, 6 e 7) como entrada do MUX e o codificador de prioridade como entrada de seleção do MUX.

**Verificação:** Teste o funcionamento completo da unidade de controle. Não esqueça de adicionar comentários a sua unidade de controle. Isso também será avaliado.

Conecte a saída ALUOp da Unidade de Controle central com a Unidade de Controle da ALU. Teste novamente o seu circuito, mas agora interagindo com ambas unidades de controle.