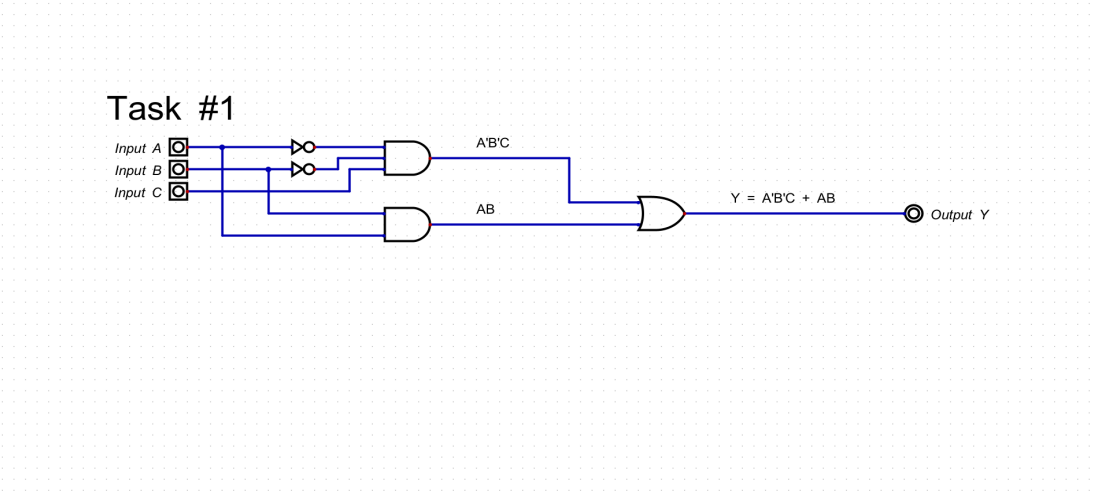
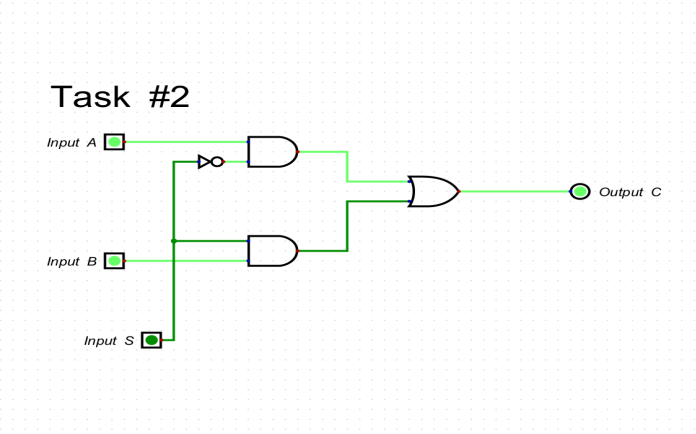
0. Selecting your CPU bitwidth (8-bit, 16-bit, or 32-bit): 8-bit

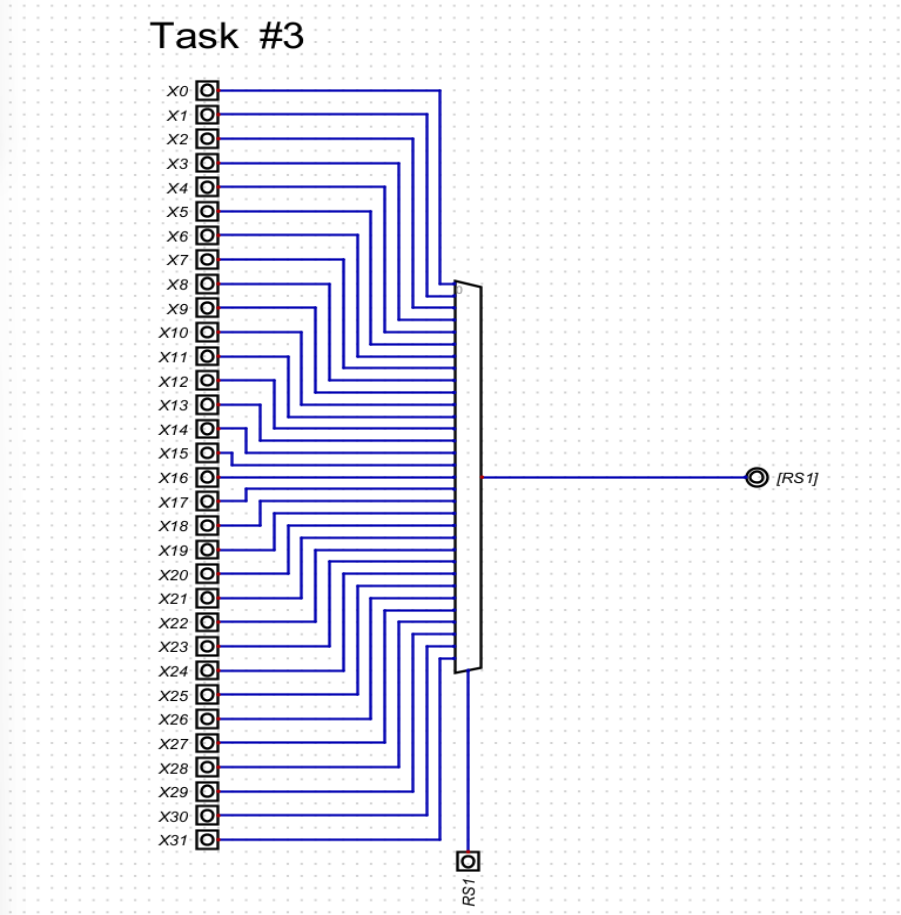
1. Given the following truth table, a) write a Boolean equation in the sum-of-products form; b) minimize the Boolean equation; c) implement the design from the Digital and experiment with the input (A, B, and C) from the truth table. Check that the output Y matches the truth table.
    a. Y = A'B'C + ABC' + ABC
    b. Y = A'B'C + ABC' + ABC
       = A'B'C + AB(C'+C)
       = A'B'C + AB(1)
       Minimized boolean equation Y = A'B'C + AB
    c. Circuit Screenshot:



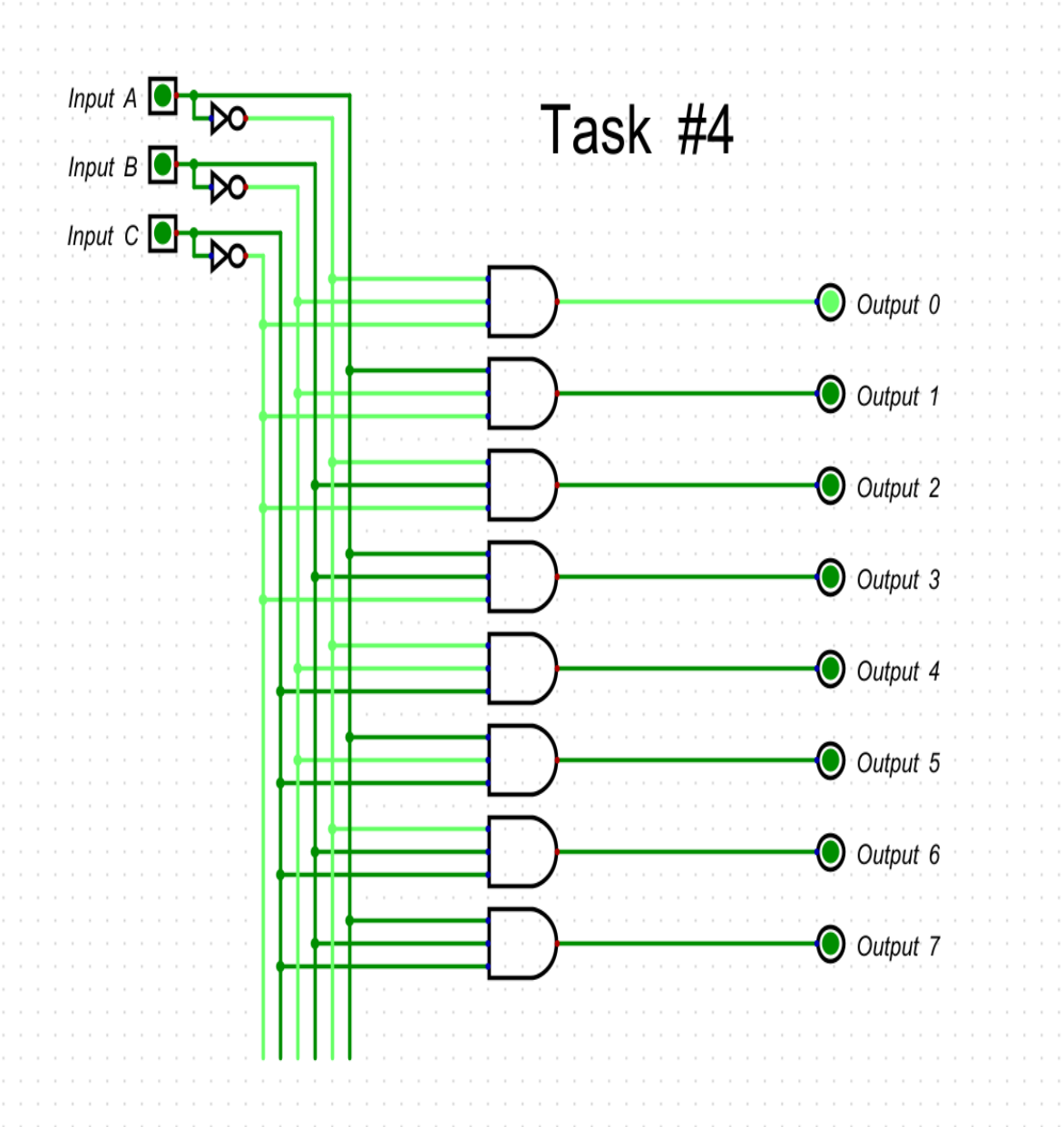
2. Design and simulate the 1-bit multiplexer from Digital based on the logic from slide 63 of Appendix A. See below. Add input and output to the logic and simulate the logic. Change the input values to check the output value changes according to the behavior of the multiplexer. Save the design in a file.
    a. Circuit Screenshot:



3. Create a new circuit and design a 32-1 Multiplexer using the Digital-provided multiplexer (ComponentsàPlexeràMultiplexer). The multiplexer is used to read the value of one of the 32 registers (X0, …, X31), the register to read is RS1 and the value is represented as [RS1]. (RS1 is one of the source operands of most instructions, e.g. add rd, rs1, rs2). The bitwidth of the input and output of the multiplexer should be the same bitwidth you selected for your CPU design. The multiplexer is used to read one of 32 registers; thus, you need to set the correct number of Selector bits for the multiplexer. After that, add input and output, the input should be named as X0, X1, …, X31, and the selector should be named as RS1 and the output should be named as [RS1]. Task 1 (20%) 2 (20%) 3 (20%) 4 (20%) 5 (20%) Total Correctness (components, input/output, bitwidth, connection, width of control, etc), 60% Organization (I/O labeled and positioned correctly, all straight wires and T junction, dot used correctly, readability etc), 40% Total 3 Configure the multiplexer in the circuit configuration window (Control+ Mouse Click). After that, add all the inputs (X0, …, X31, RS1) and output [RS1], and configure the right data bits for each of the input/output components. This new circuit should be named as "32-1-mux". Simulate the Multiplexer by setting different value of RS1, then updating the selected register input to see whether the output is changed while you simulate.
    a. Circuit Screenshot:



4. Design and simulate a 3-8 decoder from Digital based on the logic from slide 66. See below. Add input and output to the logic, simulate the logic, change the input values, and check which output line is set and whether it is set according to the behavior of a decoder (use the following truth table to check. For inputs, A, B and C can be considered as 10, 11, and 12, which are considered as the bit position of a certain address). Note Command/Ctl-C and Command/Ctl-V work very conveniently for adding the same components.
    a. Circuit Screenshot:



5. Create a new circuit and design a 5-32 decoder using the Digital-provided decoder (Components → Plexer Decoder). The decoder is used to set the Write-Enable bit of one of the 32 registers (XO, ..., X31) for writing a value to a register. The register to write is RD (RD is the destination operand of instructions, e.g. add rd, rs1, rs2). Configure the decoder in the circuit configuration window (Control+ Mouse Click) to have the correct number of selector bits. After that, add the input (RD) and output [WEO, WE1, ... WE31], and configure the right data bits for each of the input/output components. This new circuit should be named as "5-32-decoder". Simulate the decoder by setting different value of RD to see whether the correct output bit is set while you simulate.
    a. Circuit Screenshot: