# 4CS015 – Workshop #5 TO BE SUBMITTED

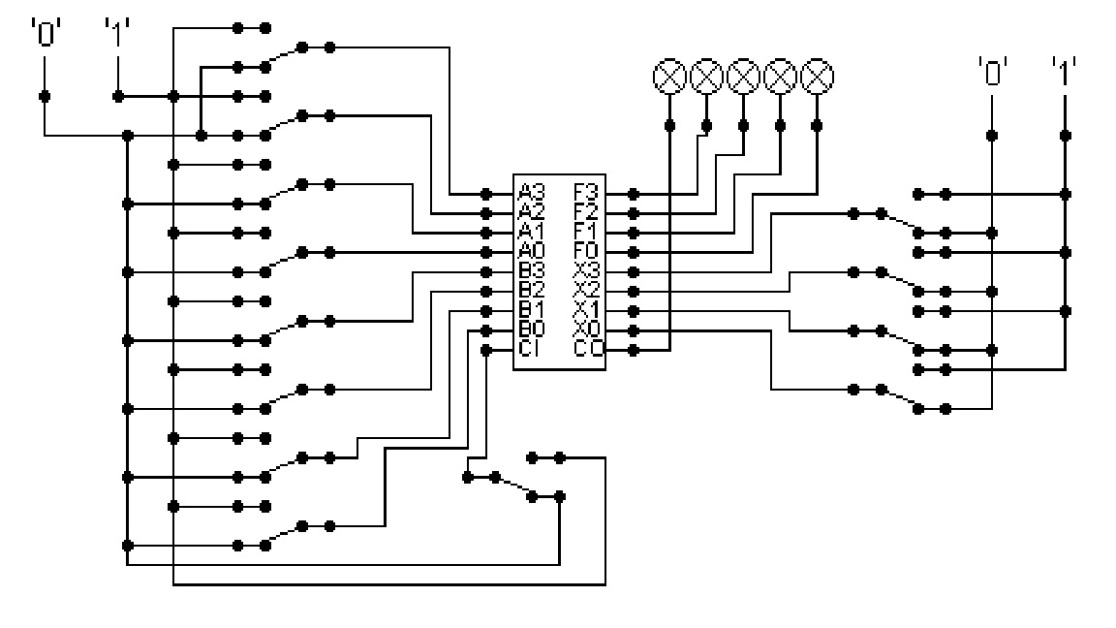
Name: Niraj Chaudhary

Student ID: 2332917

This is a marked workshop. It forms the second part of your portfolio. You will need to complete the workshop and then submit a copy of this document with a title that follows the following format (“DENNETT 1234567 wsp5.docx”), via CANVAS, by the deadline.

**Workshop tasks:**

Arithmetic Logic Unit:

Load the LogSim Arithmetic Logic Unit Circuit **alu.cct** from inside the logsim application (You'll find it in the logsim folder) (***You may need to right-click on the link to download the file instead of opening it in the browser)***. It should look like this:  
  
  
  
The circuit behaves like a simple arithmetic logic unit. The inputs A0-A3 represent a 4 bit binary number. Inputs B0-B3 represent another binary number. A0 and B0 are the least significant bits respectively. The following table details the functions supported by the chip. All other control lines = 0.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Function | AND | OR | XOR | NAND | NOR | NOT A | ADD | SUBTRACT |
| X3 – X0 | 0000 | 0001 | 0010 | 0011 | 0100 | 0101 | 1010 | 1011 |

Use A = 11 B=4, complete the following table in binary ***(15 marks)***:

A = 11 = 1011

B = 4 = 0100

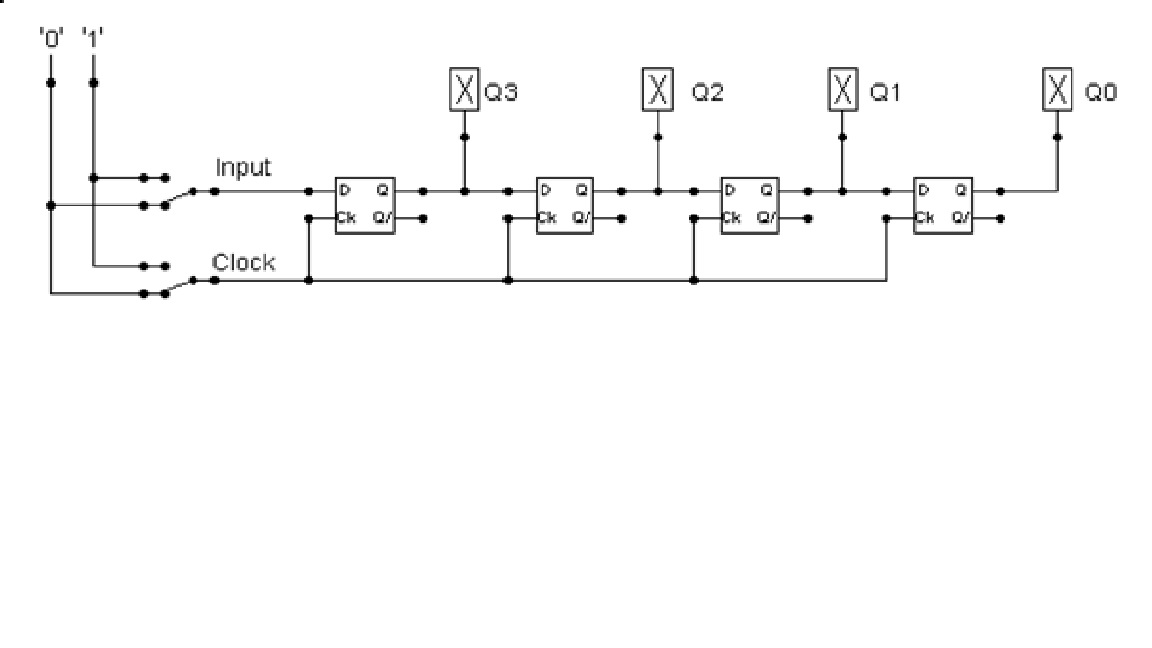
|  |  |
| --- | --- |
| FUNCTION | OUTPUT |
| AND (0000) |  |
| OR(0001) |  |
| XOR(0010) |  |
| NAND(0011) |  |
| NOR(0100) |  |
| NOT A(0101) |  |
| ADD(1010) |  |
| SUBTRACT(1011) |  |

The logical operations are bitwise. Manually prove each operation has returned the correct result by  ***(15 marks)***:  
Example:  1 0 1 1  
                 1 0 1 0 AND OPERATION  
                 1 0 1 0 RESULT

|  |  |  |
| --- | --- | --- |
| AND | OR | XOR |
| 1011  0100 AND operation  0000 | 1011  0100 OR operation  1111 | 1011  0100 XOR operation  1111 |

|  |  |  |
| --- | --- | --- |
| NAND | NOR | NOT A |
| 1o11  0100 NAND operation  1111 | 1011  0100 NOR operation  0000 | 1011  0100 NOT A operation  0100 |

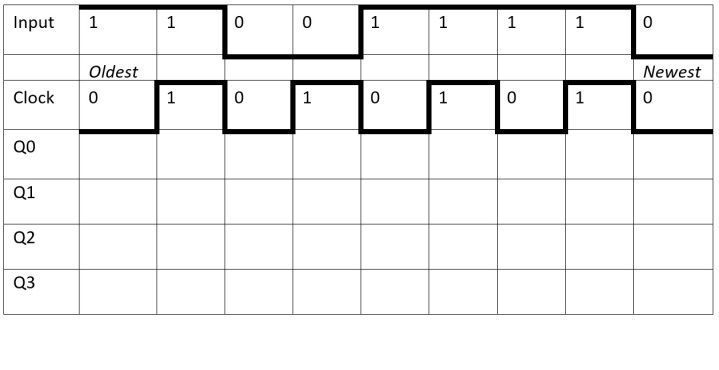
|  |  |
| --- | --- |
| ADD | SUBTRACT |
| 1011  0100 ADD operation  1111 | 1011  0100 SUBTRACT operation  0111 |

Serial to Parallel Decoder ***(30 marks)***:  


Build the circuit above and complete the following timing diagram by filling in the table spaces with ‘1’ or ‘0’. ***(15 marks)***

Diagram

Description automatically generated



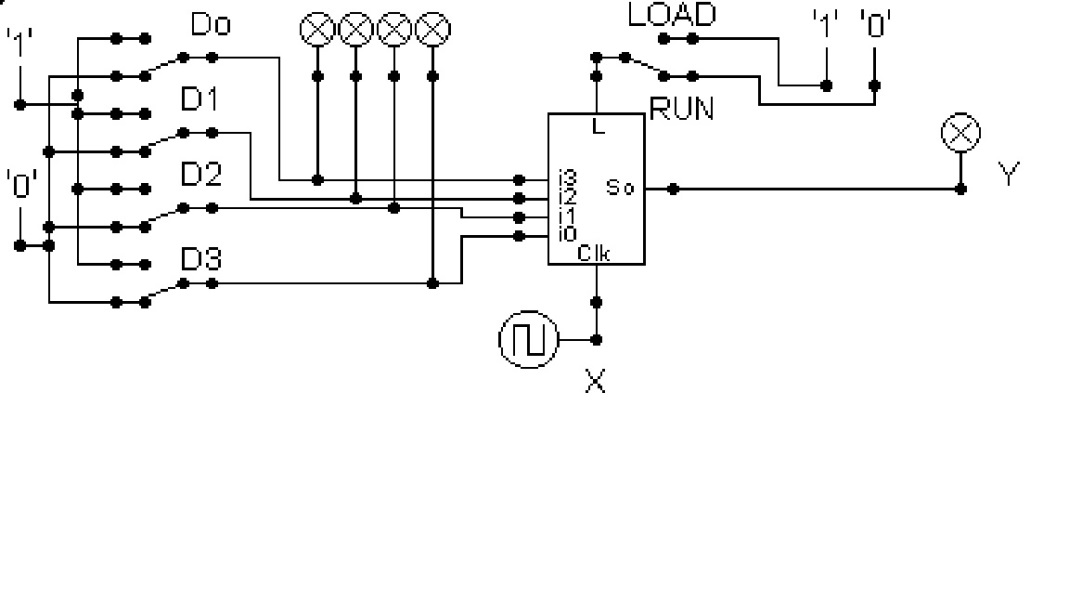
Describe what the circuit does. ***(15 marks)***

ANS: In the above figure, when the Shift registers that accept and output data concurrently via parallel and serial lines are known as Sipos. Each piece of data is first stored in a separate memory cell and then sent one at a time until all the data has been sent. The shift register's output pins are then where the data is provided in parallel form. A few applications for SIPO shift registers include digital storage, data capture, logic circuits, and communication systems. Following serial input, data are outputted here concurrently. The circuit accepts an input value throughout the course of a clock cycle and produces Q0, Q1, Q2, and Q3 as requested. When the clock value and the latest input value in the circuit described here are both zero. When the clock and the most recent input are both 0, all four output values are also 0. When the clock value and input value are both 1, all consequent values eventually turn zero, with the exception of Q3 (whose value comes out to be 1). When the circumstances change, the outputs change by 1 bit at a time. Similar to this, when the clock is set to 0 and input is set to 1, Q3 and Q2 turn on while Q1 and Q0 remain off. Additionally, when the clock value and input value are, respectively, 1 and 0, Q2 and Q1 turn on while the others turn off. In assumption, the input value is 1 and the clock value is 0.

Diagram

Description automatically generated

Parallel to Serial converter

Open the LogSim circuit **week5.cct** from the Logsim folder. It should look like this:  
  
  
  
Describe what this circuit does. ***(15 marks)***

Design and add to the above circuit an additional circuit that takes the Clock X and the Output Y and decodes Y into 4 output indicators so that they match D0 – D3. Insert the LogSim GIF output of your design in the space below.

Diagram, schematic

Description automatically generated

The highest marks will go to those who design the circuit such that it **AUTOMATICALLY** stops (not pauses) when the input to the circuit matches the output to the circuit

Diagram, schematic

Description automatically generatedAns: In this diagram, we make a circuit that automatically stops, when the input, input by the user matches the output of the circuit. Similarly, in this circuit, we used 4 NOT gates, ‘XOR’ gates, followed up by a ‘OR’ gate and a ‘AND’ gate. We have also used a 4 D – Type flip-flop for the execution of the circuit. In detail, in this circuit, input (1001 ) is given. Similarly, the given input is transferred into the D – Type circuit when the load is turned on. Moving on, to the run, when the run is turned on, the circuit is run and paused according to the given input.