# 4CS015 – Workshop #5 TO BE SUBMITTED

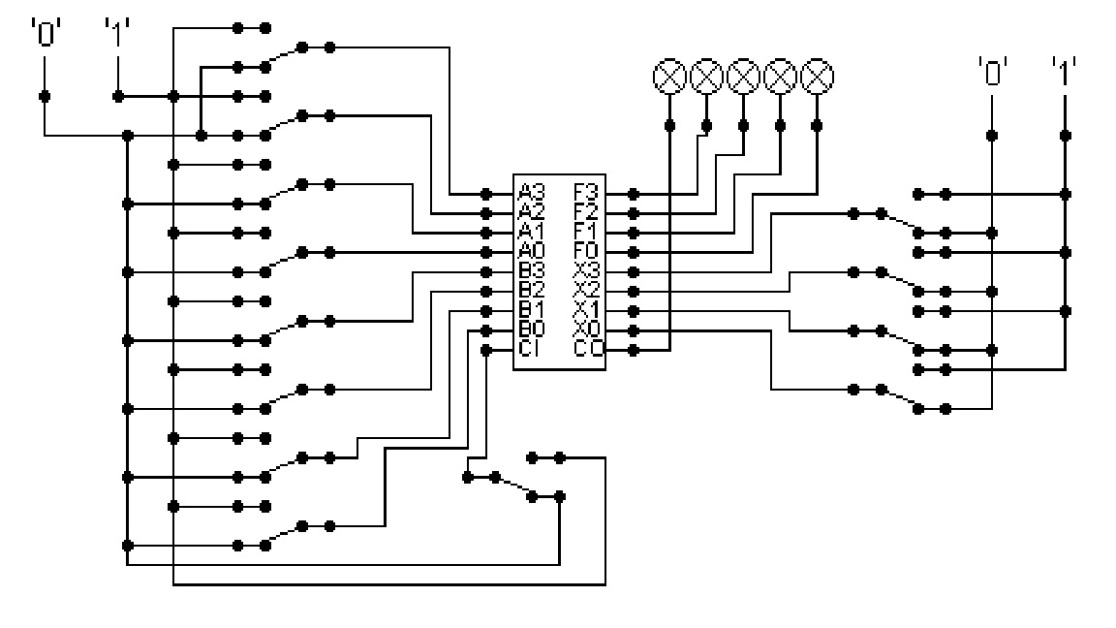
Name:

Student ID:

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

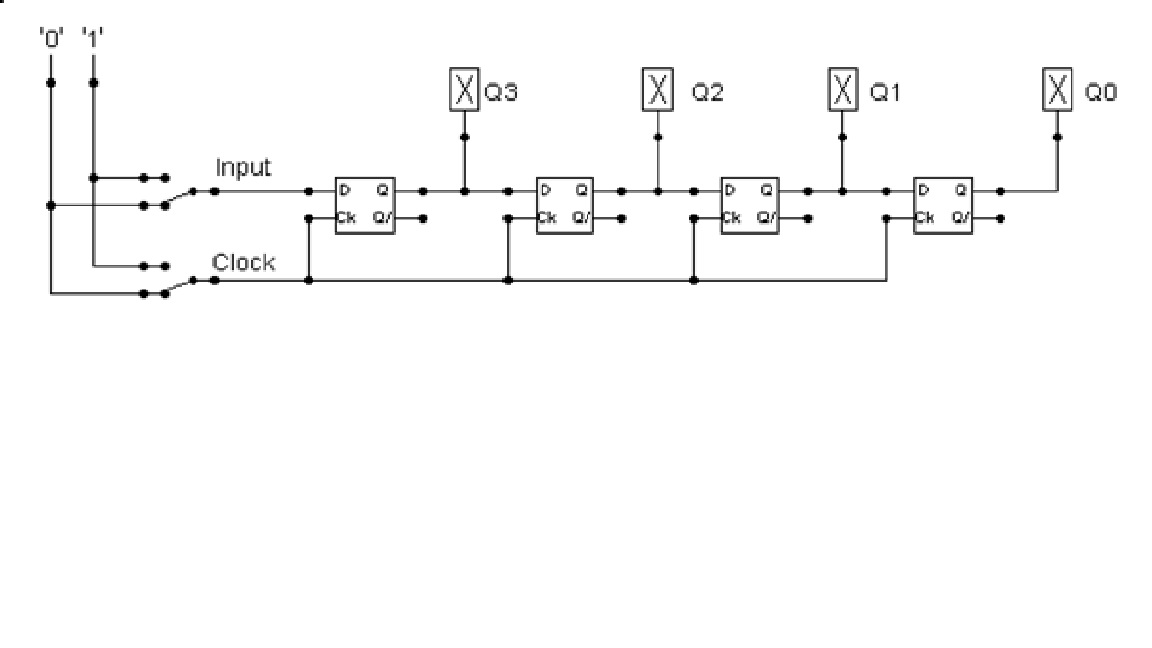
|  |  |
| --- | --- |
| FUNCTION | OUTPUT |
| AND (0000) |  |
| OR (0001) |  |
| XOR (0010) |  |
| NAND (0011) |  |
| NOR (0100) |  |
| NOT A (0101) |  |
| ADD (1010) |  |
| SUBTRACT (1011) |  |

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

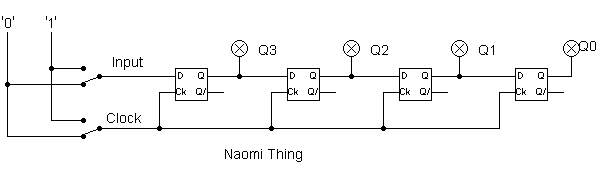
|  |  |  |
| --- | --- | --- |
| AND | OR | XOR |
| 1011  0100 (AND operation)  0000 | 1011  0100 (OR operation)  1111 | 1011  0100 (XOR operation) |

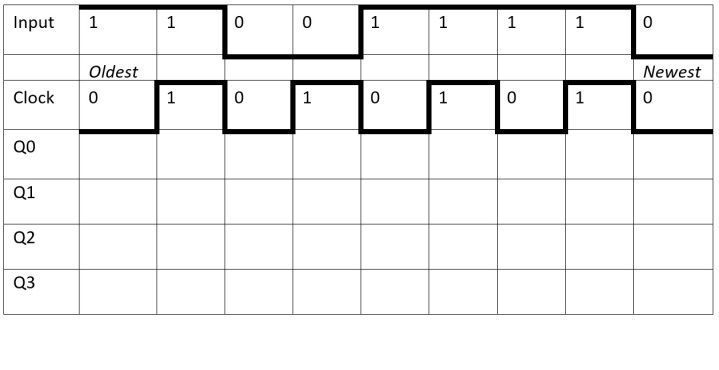
|  |  |  |
| --- | --- | --- |
| NAND | NOR | NOT A |
| 1011  0100 (NAND operation)  1111 | 1011  0100 (NOR operation)  0000 | 1011  0100 (NOT A operation)  0100 |

|  |  |
| --- | --- |
| ADD | SUBTRACT |
| 1011  0100 (ADD operation)  1111 | 1011  0100 (SUBTRACT operation)  0110 |

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)***





=

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Clock | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 |
| Input | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 |
| Q3 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 |
| Q2 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 |
| Q1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| Q0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |

Timing Diagram:

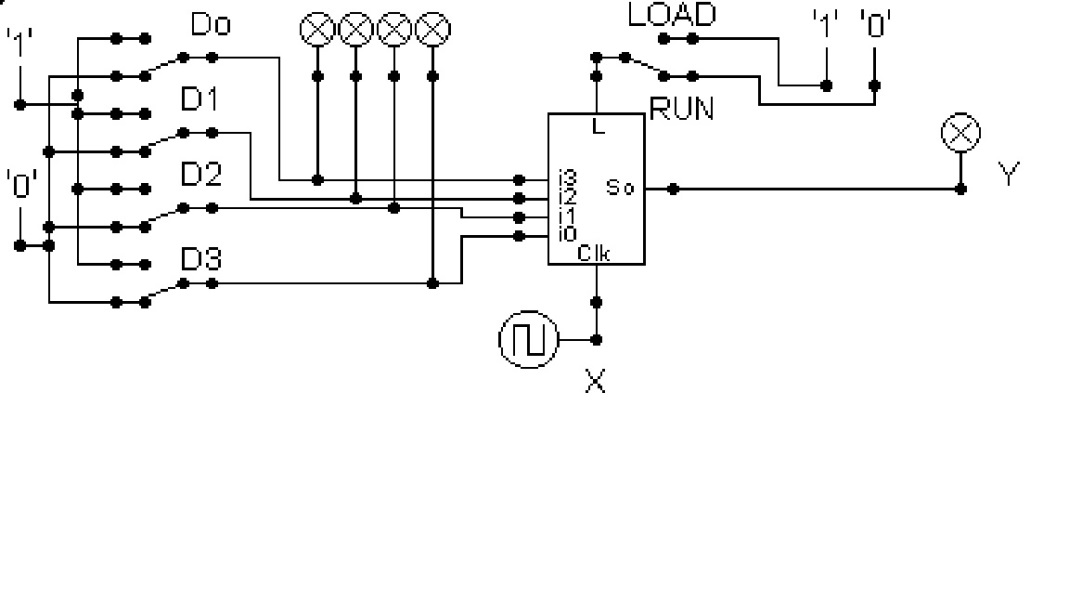
A picture containing calendar

Description automatically generated

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

= The circuit is a type of digital circuit that turns a serial input signal into a parallel output signal. This circuit is purposed to enable simultaneous transmission of numerous bits of data rather than sending them one after the other. This can speed up and improve the efficiency of data transmission, especially when it must go across noisy channels or over great distances. The circuit is also does other tasks which might carry out synchronization, data formatting, and error checking and correction.

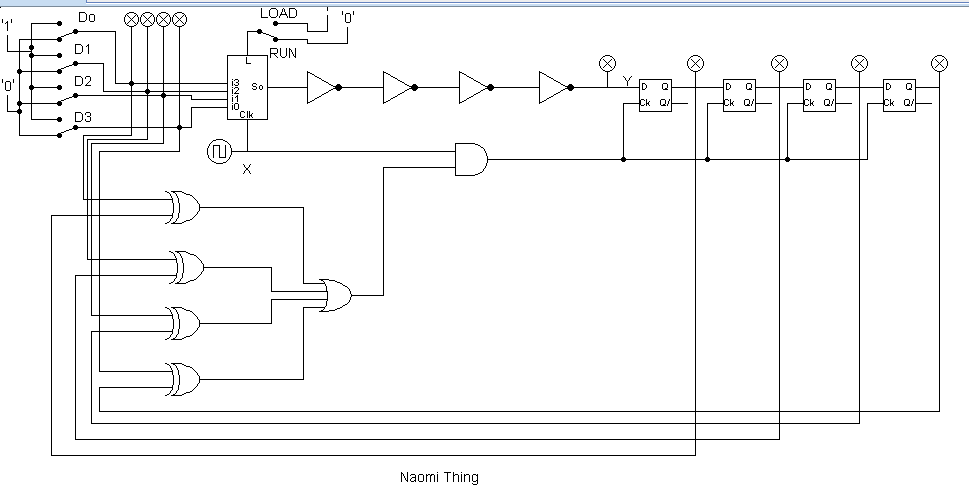
Parallel to Serial converter

Open the LogSim circuit **week5.cct** from the Logsim folder. It should look like this:  
  
  
  
Describe what this circuit does. ***(15 marks)***   
= This circuit is a type of digital circuit that transforms a parallel input signal into a serial output signal. Rather than transferring multiple bits of data concurrently, a parallel to serial converter circuit enables transmission of numerous bits sequentially. This is advantageous because a serial signal is simpler to send than a parallel signal in circumstances where the data must be delivered over long distances or across noisy channels.

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.

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

*Note: Save your GIF image when your output indicators match the input D0 - D3*. (35 marks)



Diagram, schematic

Description automatically generated