Jetic Gū
Columbia College

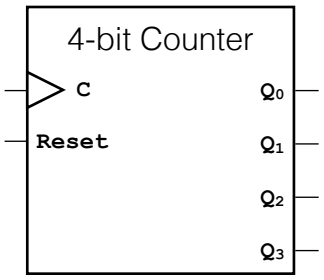
This assignment is due on 11 April 2021.

Please remember to write your name and student number.

Please submit a single PDF for each assignment. Handwritten submissions and proprietary formats (e.g. Pages or MS Word) will not be accepted. The Lab portion must be submitted separately.

Assignment 5

1. Given 16-bit operand 00110101 10101100, use a single operation (with or without specific operand) covered in class:
   A. to clear all odd bit positions to 0?
   B. to set the leftmost 4 bits to 1?
   C. to complement the least significant 8 bits?
2. Synchronous binary counter
   A. Design a counter using a 4-bit synchronous binary counter and an AND gate to design a counter that counts from 0000 through 1100 (you cannot use additional gates nor functional blocks, remember to include the CLK signal source).



   B. Design a 5-bit counter using the above 4-bit synchronous binary counter, an AND gate and a single D flip-flop (remember to include the CLK signal source).
3. A register cell is to be designed for an 8-bit register $R0$ that has the following register transfer functions:
   $\overline{S_1} \cdot \overline{S_0} : R0 \rightarrow R0 \oplus R1$
   $\overline{S_1} \cdot S_0 : R0 \rightarrow R0 \vee R1$
   $S_1 \cdot \overline{S_0} : R0 \rightarrow \overline{R0 \oplus R1}$
   $S_1 \cdot S_0 : R0 \rightarrow R0 \wedge R1$
   Here, $\vee$ denotes bitwise OR operation, while $\wedge$ bitwise AND operation. Design the circuit for a register bit (D flip-flop).
4. Use D flip-flops and gates to design a counter with the following repeated binary sequence: 0, 2, 1, 3, 4, 6, 5, 7.

5. Given two 8-bit registers $R1$ and $R2$, two 8-bit wires with new parallel data inbound for $R1$ and $R2$, and 2-bit mode selector S_1 and S_0, show the diagram of the hardware that implements the register transfer statement.

   $C_1: R2 \leftarrow R1, R1 \leftarrow R2$

S_1	S_0	**Register Operation**
0	0	No change
0	1	Perform C_1
1	0	Load parallel data
1	1	Clear both registers to 0

Lab 4

You must complete the following assignment and submit a PDF of instructions enough to replicate your results, and required documentation. You will also need to upload LogicWork circuit design files as specified, and your own library file. Then upload a single ZIP file to student portal.

1. Save the library and circuit files we created in class containing the following designs in the final ZIP file:

   (1) Implement 4-bit Ripple Counter in LogicWorks (`circuit1-1.cct`);

   (2) Implement 4-bit Synchronous Counter in LogicWorks (`circuit1-2.cct`);

   (3) Implement 3 digit BCD counter using 4-bit binary counters (`circuit1-3.cct`);

   (4) Provide 10 CLK ticks' timing diagram for (3).

2. Circuit design (`circuit2.cct`).

   You are to design a datapath with a 4-bit register array (4 GPRs inside) that can perform certain functions. The datapath takes input $F_{2:0}$ from the control unit for function selection, $Add^1_{1:0}$ and $Add^0_{1:0}$ for register selection, $Ip_{3:0}$ for value input, $Op_{3:0}$ for value output.

$F_{2:0}$	**Register Operation**
000	No change
001	Clear a single register to 0, selected by $Add^1_{1:0}$
010	Perform register transferring, assign the value of register at address $Add^0_{1:0}$ to register at $Add^1_{1:0}$.
011	Load value from $Ip_{3:0}$ into a single register, selected by $Add^1_{1:0}$
100	Output value from a single register to $Op_{3:0}$, selected by $Add^1_{1:0}$
101	Output inverted value from a single register to $Op_{3:0}$, selected by $Add^1_{1:0}$
110	Perform addition of 2 registers, selected by $Add^1_{1:0}$ and $Add^0_{1:0}$, store the output to register with address $Add^1_{1:0}$. (Use the adder-subtractor functional block)
111	Perform subtraction of 2 registers, selected by $Add^1_{1:0}$ and $Add^0_{1:0}$, store the output to register with address $Add^1_{1:0}$. (Use the adder-subtractor functional block)

   A. Write down the sequence for all necessary inputs for computing 4 + 5 - 7. You will need to load number 4, 5, 7 into the datapath, then perform the necessary calculation, and finally output the results to $Op_{3:0}$. (1pt)

B. You can use any functional blocks we've discussed in class. The grading criteria for this assignment is as follows:

1pt: register array clearing function works properly ($F = 000{,}001{,}010$)

1pt: register array transferring function works properly ($F = 011$)

2pt: register array output function works properly ($F = 10X$)

2pt: register array arithmetic function works properly ($F = 11X$)