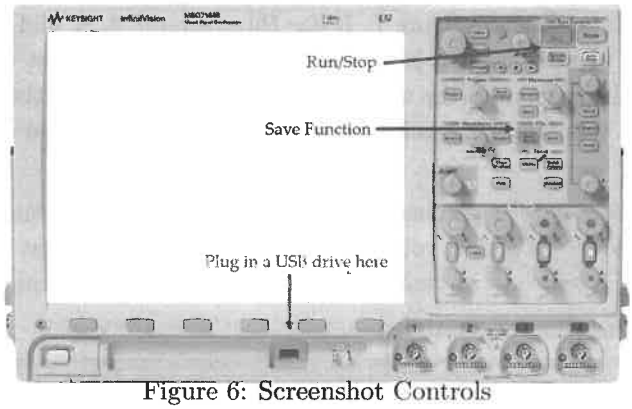

Figure 6: Screenshot Controls

A well-annotated screenshot that shows analog and digital representations of the clock. A good figure will label all signals on the scope, shows 5 periods of the wave, and takes up most of the screen. Annotate the following physical characteristics of the clock waveform:

- High Level Voltage
- Low Level Voltage
- Frequency
- Duty Cycle (percentage of time HIGH relative to one period).

3.4 Mystery Flip Flop Circuit 1

You will now examine the first of two mystery D-flip-flop-based circuits. In the subsequent pages there is space for you to take notes as you explore the behavior of the circuit. Your deliverable for this part of the Lab will be a short (read, concise is better than long!) write up of what the circuit does, with evidence to support your arguments. When debugging a circuit, it is quite helpful to keep running notes as you try different experiments and observe different outcomes.

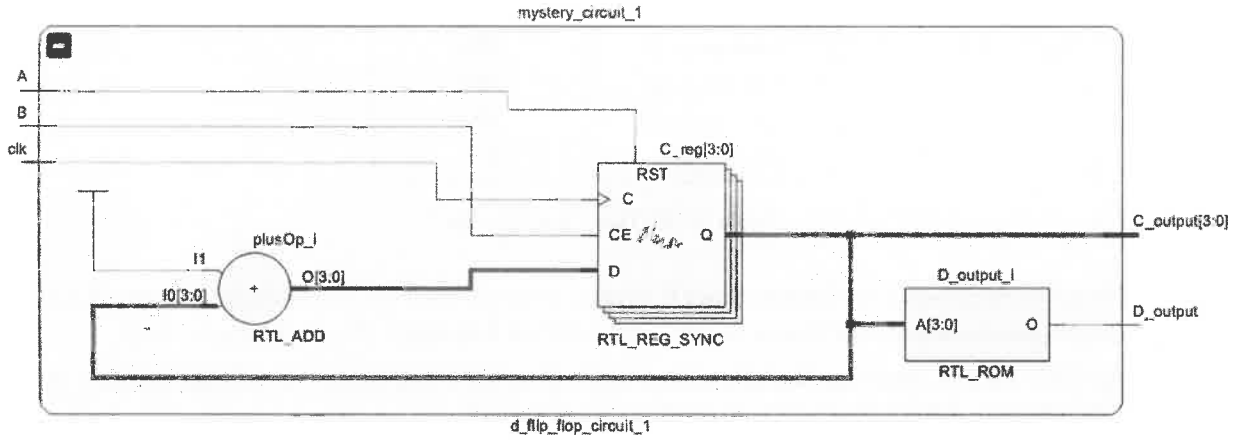

Figure 7: Mystery Circuit #1

The schematic for the circuit is shown in figure 7. It has three inputs: `A`, `B`, and `CLK`, and two outputs: `C_output` and `D_output`. Your goal is to identify what this circuit does and to think about why it might be useful. Observe that `C_output` has four bits, indexed from 3 (the most significant bit) down to 0 (the least significant bit) and should be interpreted collectively. On this diagram, multi-bit signals (busses) are drawn with heavy lines, and single bits with thin lines.

Without playing with the hardware implementation of this circuit—just looking at this circuit diagram—what do you think that this circuit does? (It's ok if you aren't totally sure, or if you get elements of this wrong—just get your initial thoughts on paper. It's also ok to annotate the schematic or draw timing diagrams!).

Deliverable 2: Circuit 1 Function *(2 points)*

In the space below, briefly describe what you think this circuit does.

Not totally sure what this circuit does but it is strange that there are 3 input bits but essentially 8 output bits. Since B is labelled CE maybe it enables the C output. And maybe A is for resetting C to 0000. Otherwise C increments by 1 at each clock rise?

Now consider the hardware implementation. Frequently, we will use tables to keep our signal and socket names organized. As it turns out, there is a LOT of bookkeeping in digital electronics, and it all needs to be just right for our circuits to work! `A` and `B` are slide switch inputs. `C_output` and `D_output` are connected to Pmod socket JB (the upper right socket) so you can measure the outputs with the digital oscilloscope. The mapping used is shown in Table 1.

RTL Port	Basys3 IO
`A`	SW1
`B`	SW0
`CLK`	(Internal)
`C_output(0)`	JB1
`C_output(1)`	JB2
`C_output(2)`	JB3
`C_output(3)`	JB4
`D_output`	JB7

Table 1: IO Used for Circuit 1

Connect the digital oscilloscope to `C_output` and `D_output` (you will need two 6-pin headers). If you have questions about how to make these connections, please ask a member of the Lab staff! We are happy to help!

Turn analog CH1 off, but leave the digital probe connected to the clock as before. Set all the slide switches on your board LOW (down). Adjust your scope to show 20 clock periods.

What does setting `B` (the right most switch) HIGH do? What is `B` connected to in the RTL schematic?

Deliverable 3: Screen capture *(5 points)*

With the B input asserted, capture a scope trace showing 20 periods of the clock. Submit your annotated screenshot as part of your submission. In the space below, what B does.

B enables the counting of clock cycle when it is high.

What does setting A HIGH do? What is A connected to in the RTL schematic?

Deliverable 4: Function of A *(2 points)*

Experiment with the A input, while monitoring the scope trace. Describe below what A does.

A sets all of the outputs low

Deliverable 5: Input Priority *(2 points)*

Determine whether A or B has priority — that is, do you observe that A overrides B, or that B overrides A? Note your conclusions below

A overrides B because even when B is high, all of the outputs go low. And when A is high and I set B high, nothing happens.

Deliverable 6: Behavior Truth Table *(2 points)*

Fill out this behavioral truth table. This is like a logical truth table, except it captures behavior. The first element is filled in for you.

A	B	Circuit Behavior
0	0	Circuit holds its current state
0	1	Circuit counts up on clock rise (Carry clear)
1	0	All outputs go low
1	1	All outputs go low

Table 2: Circuit 1 Behavior

An RTL ROM (*Read Only Memory*) is a block that contains combinational logic. What logic function is implemented inside that box (connected to `D_output`)?

Deliverable 7: ROM Table *(5 points)*

Fill out the truth table, then summarize your results in the space provided. Be sure to indicate what function D performs.

C3	C2	C1	C0	D
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	1

Table 3: ROM Truthtable

D = ~~C3~~ $C_3 C_2 C_1 C_0$ which is on and operation on all four C outputs. ~~this~~ D is high if all 4 bits of C are high, on low otherwise

Now that you have had an opportunity to explore the behavior of this circuit, describe what this circuit does at a high level (frame your discussion of in terms of `CLK`, `A`, `B`, `C_output`, and `D_output`). Back up your arguments with evidence by referencing the circuit schematic and oscilloscope screenshots.

Deliverable 8: Circuit Description *(5 points)*

Describe the function of this circuit. Submit any necessary (annotated) screenshots to go with your description. (You may refer to the screenshot submitted for Deleverable 3)

At a high level, the circuit increments from 0000 to 1111, incrementing the count every other clock rise. At 1111, the count overflows and starts over at 0000. This behavior only occurs when B is high and A is low, IF A is high, the counter is reset ~~all outputs~~ to zero, and if B is low, the counter holds its current count

Deliverable 9: Circuit Utility *(2 points)*

Describe one potential use for this circuit.

This circuit can be used as a timer counting from 0 to 15 which corresponds with 30 clock cycles. It can be calculated how long this timer would take by $30 \times T$ where T is the clock period. The output would be D which happens every cycle.

3.5 Mystery Flip Flop Circuit 2

You will now examine the second of two mystery D-flip-flop-based circuits (see Figure 8). Your deliverable for this part of the Lab will be a short write up of what the circuit does, with evidence to support your arguments.

The circuit has two inputs, `X` and `CLK`, one primary output, `Z_output`. The intermediate flip-flops have their outputs mapped externally as well and will be referred to as `Y1_output` and `Y0_output` in this handout. Additionally, the X input has been copied (passed directly to) to `X_output`, so you can visualize the button push on the scope.

Without playing with the hardware implementation of this circuit—just looking at this circuit diagram—what do you think that this circuit does?

Deliverable 10: Circuit 2 Function *(2 points)*

In the space below, briefly describe what you think this circuit does.

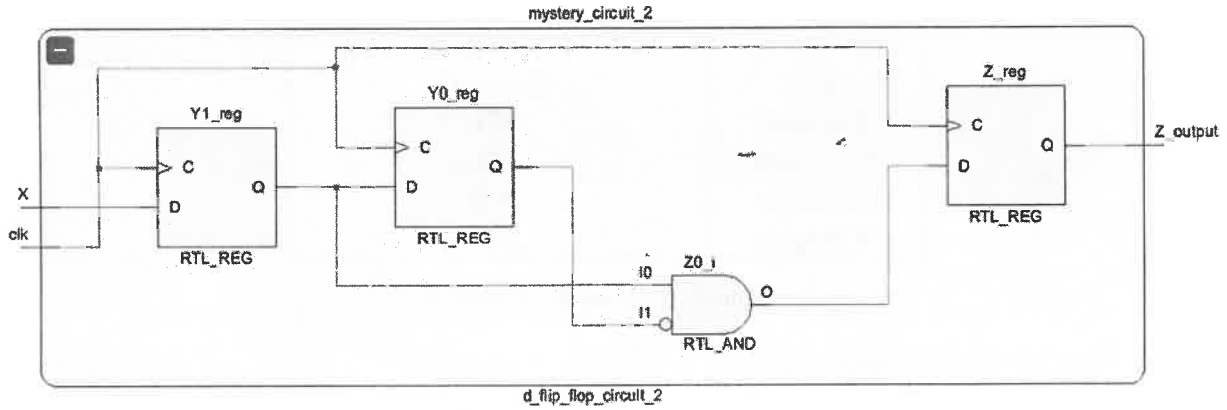

Figure 8: Mystery Circuit 2

When the button is pressed or released, Z will go high. However if button stays low or high, Z will remain static. Activity when button goes from pressed, to released.

Deliverable 11: Truth Table *(2 points)*

Fill in the truth table (4) for the logic gate labelled `Z0_i` in Figure 8. Note, the two inputs `I1` and `I0` are connected to `Y0_output` and `Y1_output` respectively, so you can observe them directly on your oscilloscope.

I0	I1	O
0	0	0
0	1	1
1	0	0
1	1	0

Table 4: Truth Table for Z0_i

Now consider the hardware implementation.

`X` is a pushbutton input. `X_output`, `Y1_output`, `Y0_output`, and `Z_output` are connected to the Pmod socket JC (the bottom right socket) so you can measure the outputs with the digital oscilloscope. The mapping used is shown in Table 5.

Connect the digital oscilloscope to `X_output`, `Y1_output`, `Y0_output`, and `Z_output`. Leave the analog oscilloscope on the clock.

When `X` is pressed, it applies a HIGH input to the system. When `X` is not pressed, it sends a LOW to the system.

RTL Port	Basys3 IO
`X`	BTNC (Center pushbutton)
`CLK`	(Internal)
`X_output`	JC1
`Y1_output`	JC2
`Y0_output`	JC3
`Z_output`	JC4

Table 5: IO Used for Circuit 2

Press and hold `X` for a few seconds, then release it. Observe the waveform that results, relative to the clock. When you press `X`, when does `Y1_output` change? When does `Y0_output` change? Try doing this for a few different length button presses.

Deliverable 12: Circuit 2 Screen Shot *(2 points)*

Arrange your scope trigger to clearly capture the behavior of this circuit. Submit an annotated screen capture showing which signals affect each edge of the output.

Deliverable 13: Circuit 2 Behavior *(2 points)*

In the space below, explain clearly and concisely how this circuit works.

When the button is pressed (goes from low to high), the 2 output stays low for one clock cycle and then goes high for one clock cycle, and stays low after.

Deliverable 14: Circuit 2 Utility *(2 points)*

In the space below, describe a situation in which this circuit might be useful.

This circuit would be useful when you want an input to be synced with the clock edge, with a delay rather than instantaneous.