Discussion & conclusion

In this project, we were given a unique code to print using a 7-segment display. In part one of the project we built a circuit that did so and we called it a combinational circuit.

In the discussion and conclusion included in my project phase 1, I go in-depth about 7-segment displays; how to construct various combinational circuits using truth tables, k-maps, Boolean algebra and so on. Therefore, for this the discussion and conclusion about the project phase 2, I'll only talk about the newer additions to the project.

Combinational circuits cannot hold information as it does not have any memory. This also means there is no state. However, sequential circuits do have memory, therefore they can interact with states based on inputs. This is done by utilizing previous input, clock and a memory element.

Flip-Flops are the memory elements we learned about in this course. There are three types of flip-Flops: JK flip-flop, T flip-flop, D flip flops. These flip-flops take in a clock pulse, input and gives out an output and a inverted output.

These flip-flops cycle through states of 0 and 1. We can make them hold many information by using a lot of flip flops. For example: 3 flipflops can store 3-bit info.

We cycle through states by giving a specific input function to the flip flops. The minimum inputs that are necessary to generate a "next state" when the correct state is known are plotted down in an excitation table.

I used a JK-flipflop for my memory element and wanted to cycle through it's states in this order: (001, 010, 011, 100, 101, 110) which I was able to by creating a state table and utilizing JK Flip flop excitation table.

Excitation table for ~~Jk~~ Jk FlipFlop
_____

Q	next Q	J	K
0	0	0	x
0	1	1	x
1	0	x	1
1	1	x	0

After that I found out the input for Jk Flip Flop
functions by utilizing K-maps.

Afterwards a circuit was created, which
cycled through states as mentioned
(also shown in the state diagram).

Then I attached the output of the Flip Flop to
my previously built combinational circuit in
this fashion: $C \rightarrow Z$, $B \rightarrow Y$, $A \rightarrow Z$ in
(shown in diagram)

Finally, when the simulation is turned on.
It automatically ~~of~~ cycles through the
code "ES1462" without user input.