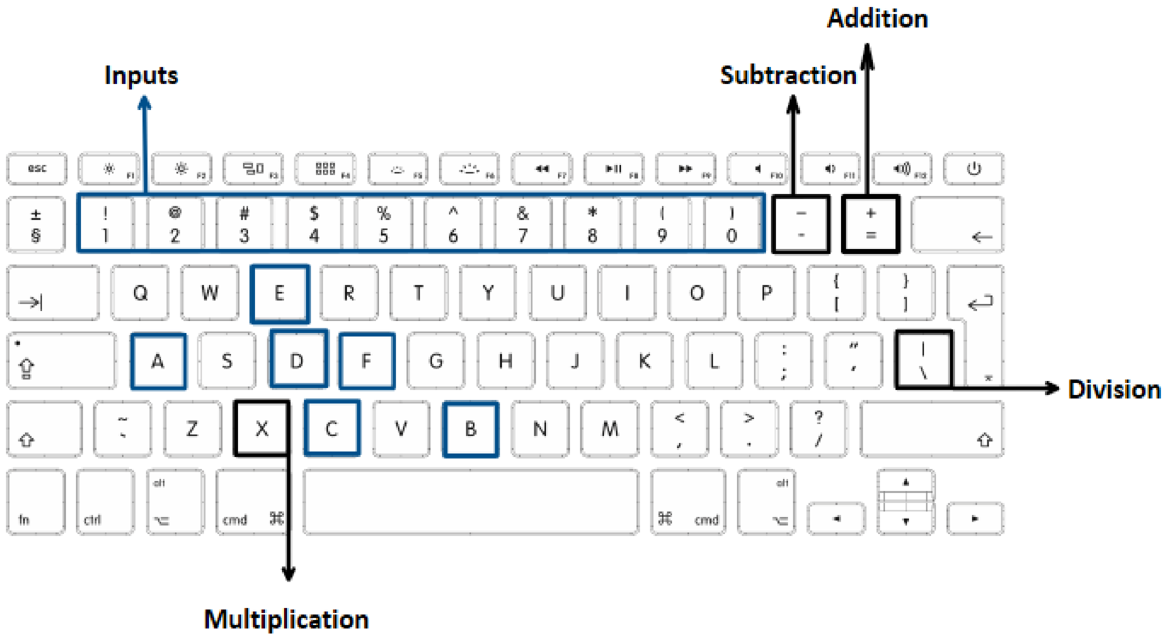
8 BIT SIGNED CALCULATOR

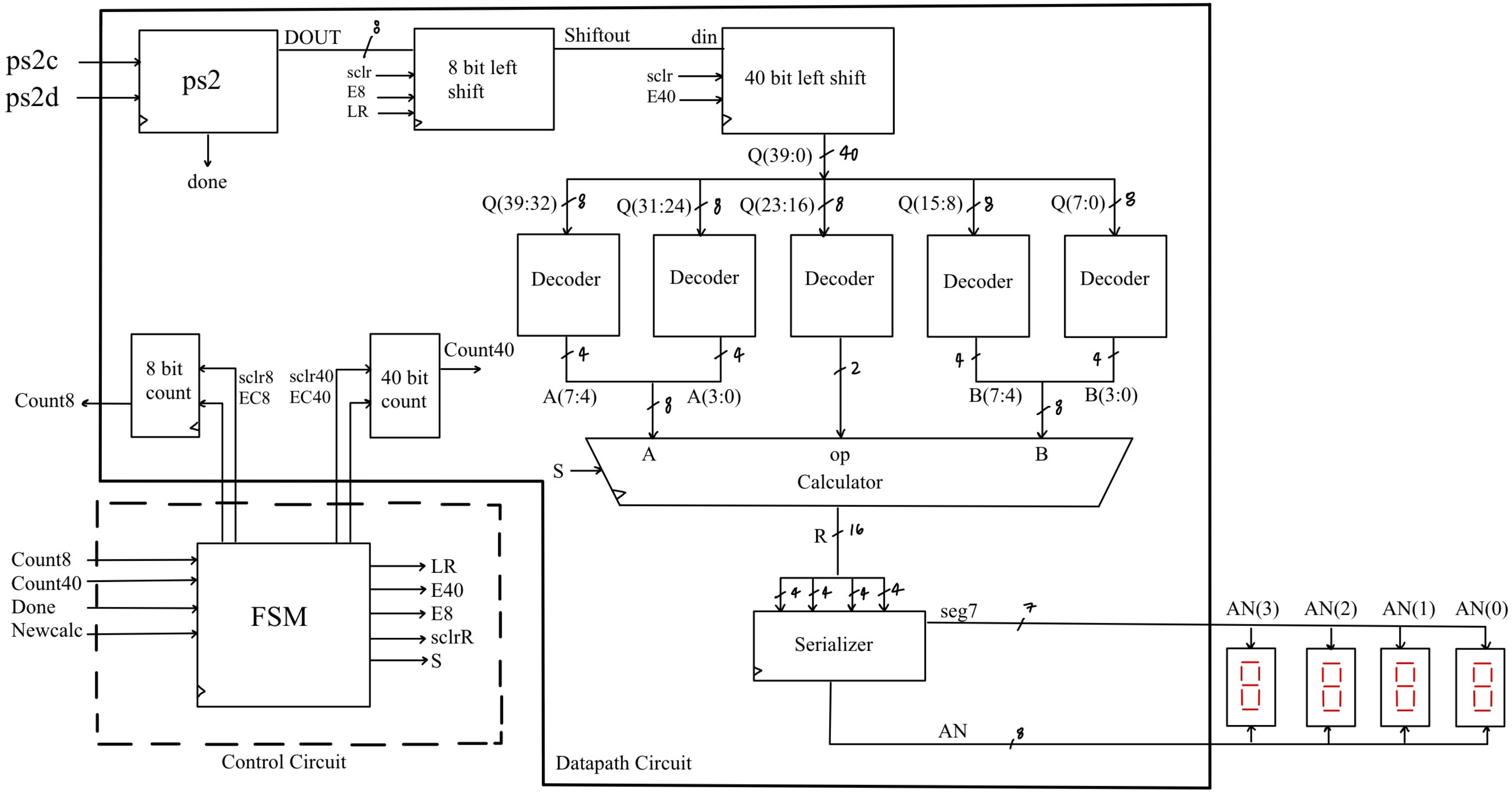
Andrew Fergan, Austin Nierporte, Paige Seyfarth, Francesca Cipriano

Calculator Overview

- This calculator takes five key strokes that equate to two 8-bit signed numbers and an operation.
- Inputs are entered on a keyboard as hexadecimal numbers and outputted as a 16-bit signed number on 4 seven segment displays.
- For the calculator to function properly, enter a number as hexadecimal followed by the operation and the second hexadecimal number.
- When entering a number, any other key stroke besides 0-F is assigned the value of 0.
- When entering the operation, if any other key besides the designated operations is pressed, the operation defaults to multiplication.
- To enter a negative number, type in the 2's complement of it

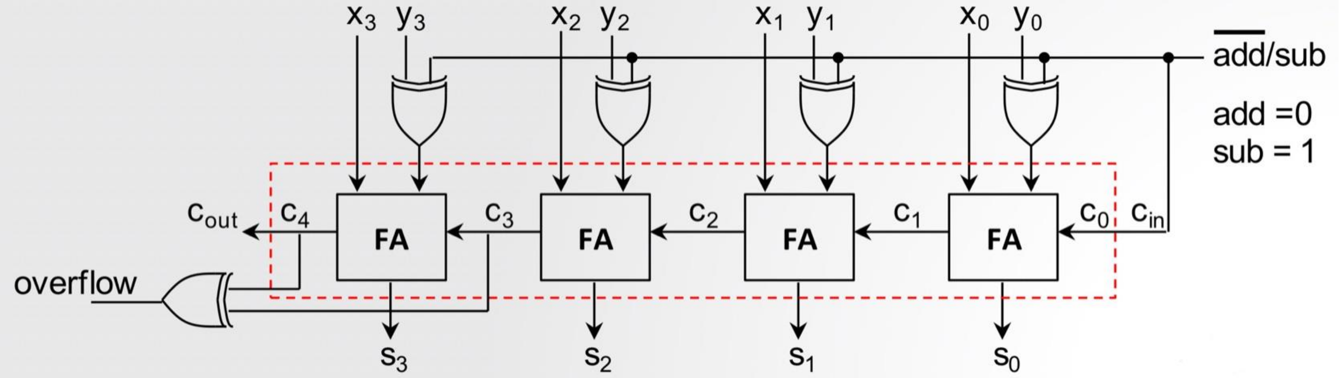


Top File Layout



Add/Sub

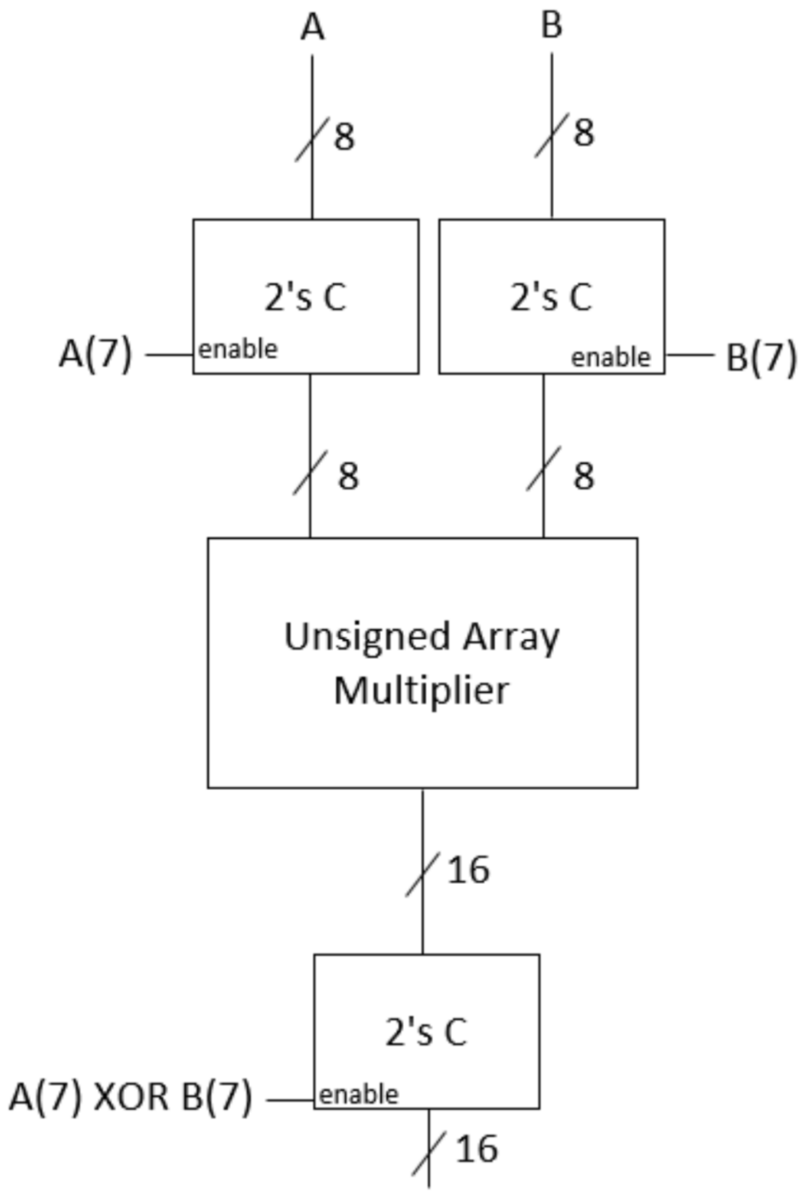
- Entering the add-sub are two 9 bit inputs that are sign extended to avoid overflow.
- We have two add/sub functions, but one functions as an adder and the other functions as a subtractor. The only difference is the adder carries in a value of 0 while the subtract or carries in a value of 1.
- In order to use the addition function, the “+” button must be hit on the keyboard. To use the subtraction function, the “ – ” button must be hit on the keyboard.



This is a 4-bit adder/sub, and we used a 9-bit adder/ sub

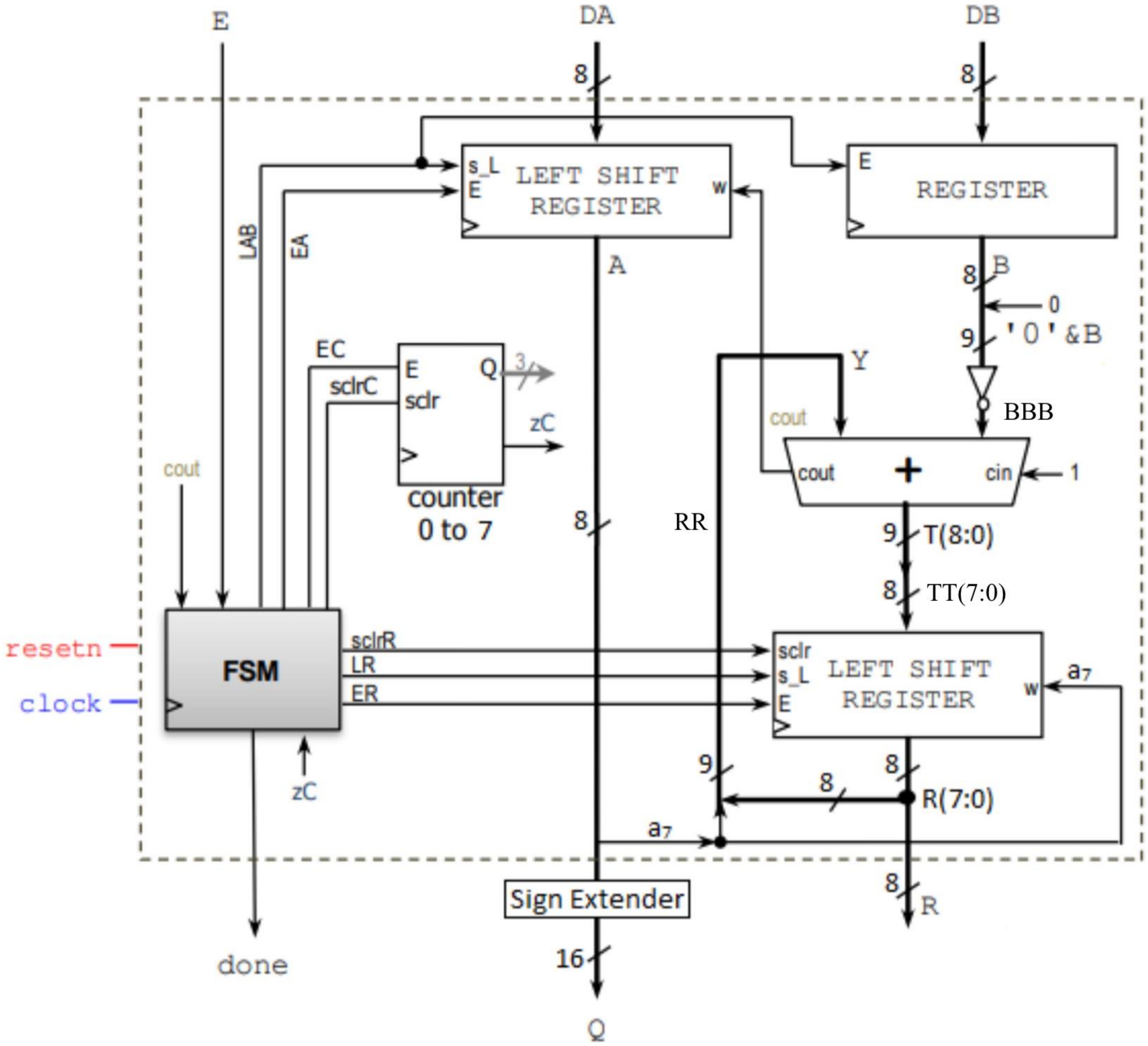
Multiplier

- We used the multiplier from the lab that was unsigned and instead of changing it, we used 2’s compliment to turn it into a signed function.
- Our 2’s compliment component works by looking at the value of the enable. If the enable is 1, then 2’s compliment will be performed. If the enable is 0 then the original signal is passed through.
- The multiplication function is used with the “x” key on the keyboard.



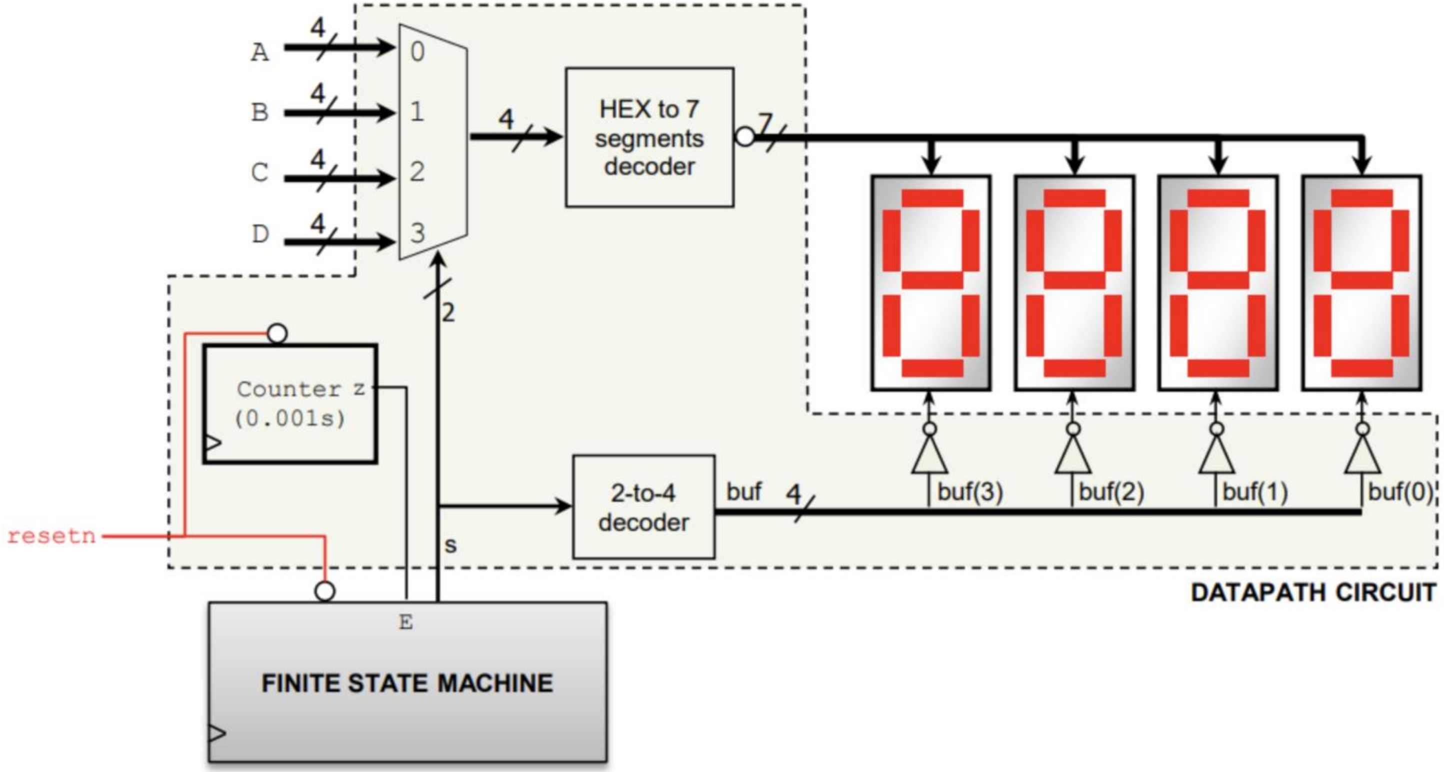
Divider

- We used the unsigned divider from the lab and changed it to a signed divider using 2’s compliment (similar to the multiplier).
- The divider used a clock, shift registers, a counter, and a FSM to control the circuit
- To use this function the user must press the “/” key on the keyboard



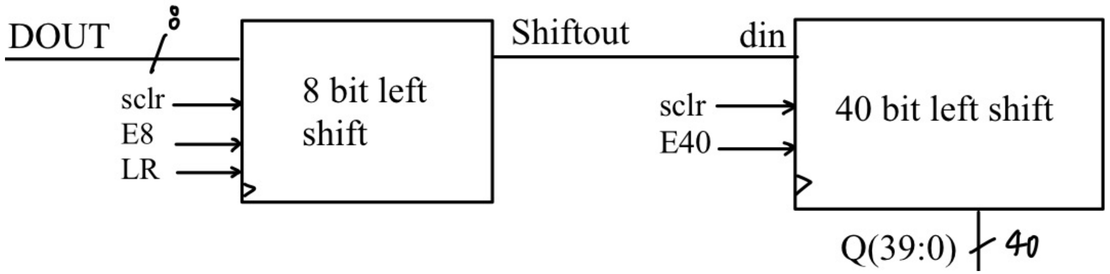
7–Segment Serializer

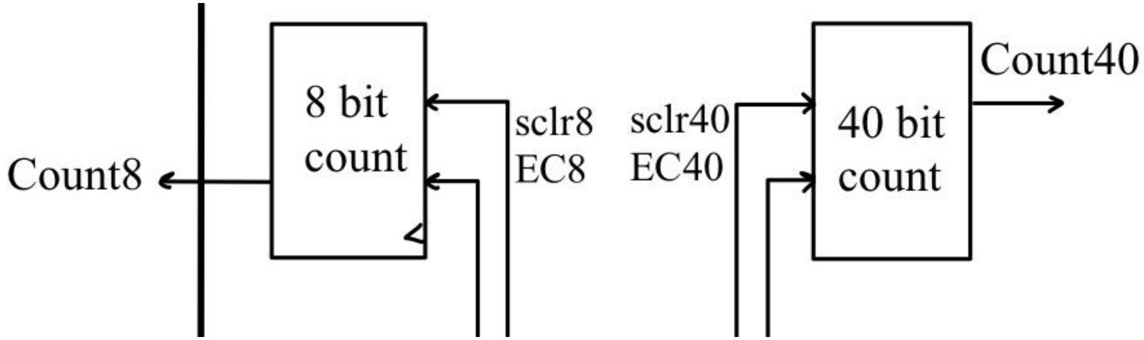
- We used Professor Llamocca's Unit 7 Notes
- Once hexadecimal numbers are input with an operand, the output of the calculator is divided into 4 buses. Each bus is then used as an input for the serializer
- By using 4, seven-segment displays we are able to display the 16-bit calculation



Left shift register and counter

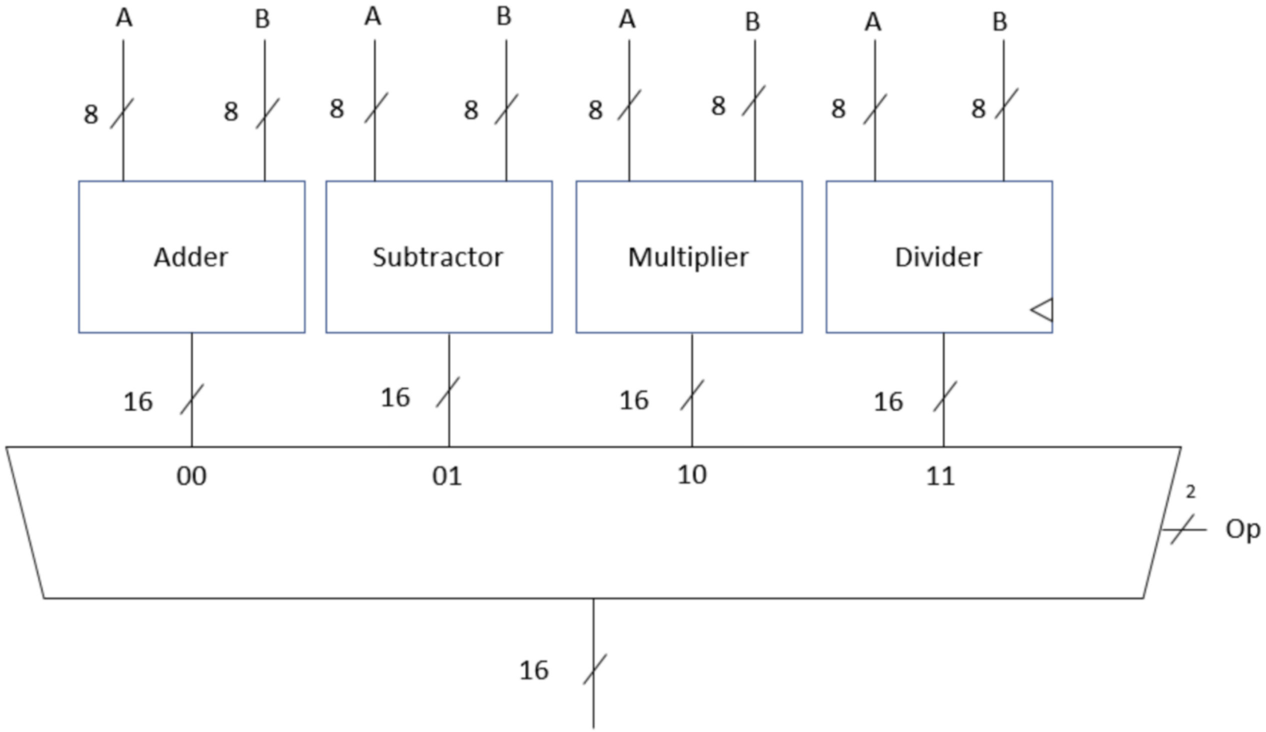
- This calculator consists of 2 left shift registers and 2 counters.
- These are working during the state 2 of the finite state machine making sure all the bits are counted for so the calculator can work properly.
- As the bits shift out of the 8 bit register, they get moved into the 40 bit register.
- The counters are responsible for counting how many bits are getting shifted out.





Encoders and calculator

- The encoders encode the data from the 40 bit-register as either the hexadecimal input or the operation.
- The output from the encoders are the inputs to the multiplexer.
- The part of the circuit that acts as a calculator takes the 4 operations and connects them through a multiplexer. The multiplexer takes the multiple input signals and synthesizes them to a single output.
- The output of the multiplexer is determined by the operation that is being inputted.



FSM

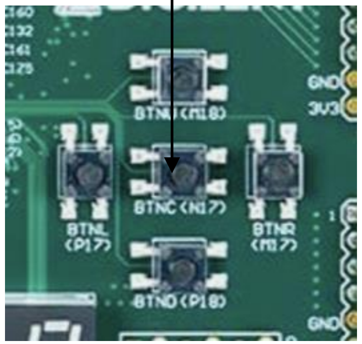
This Finite State Machine consists of three different states.

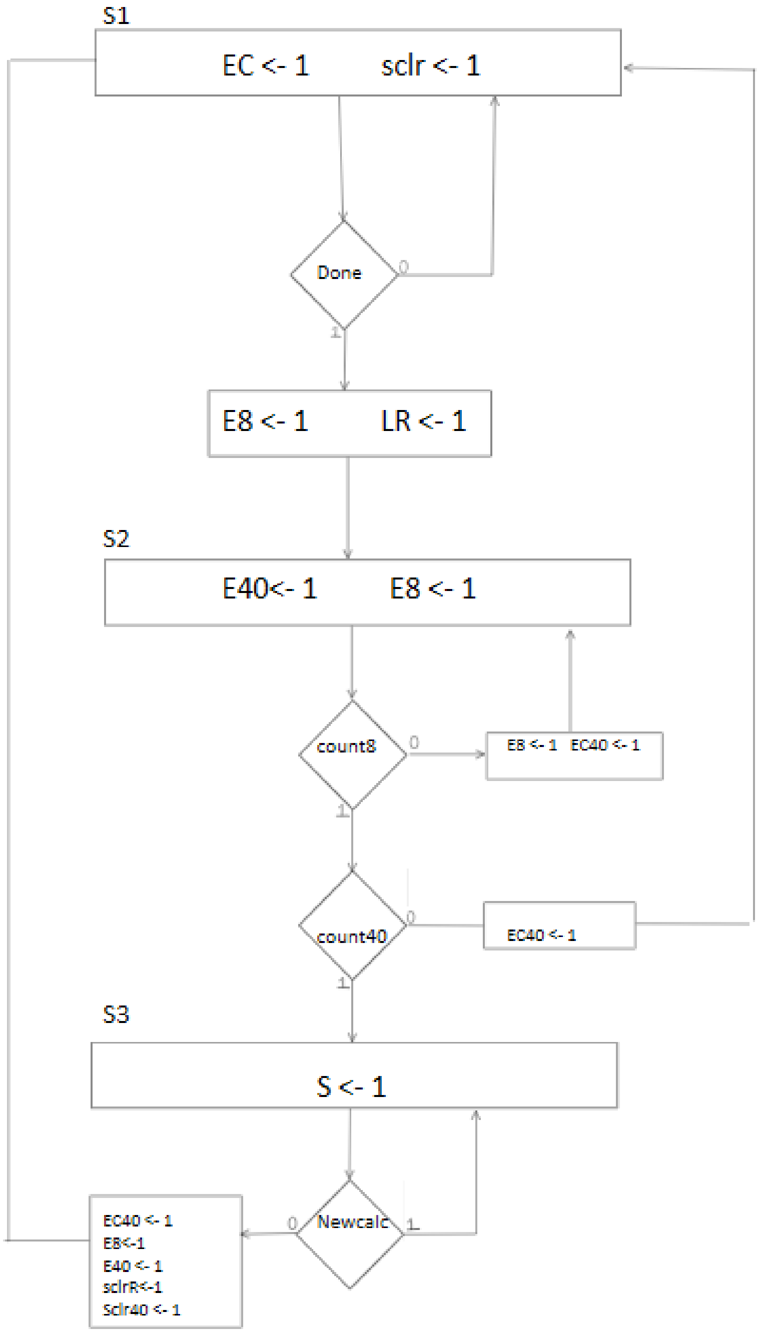
State 1 is responsible for waiting for an input. If an input is detected the input is loaded into the 8-bit register.

State 2 is responsible for detecting all 40 bits from the 5 input keys. When state 2 counts to 8 bits, it returns to state 1 each time until all 40 bits are accounted for.

When state 3 is hit then the calculation process is complete and the middle button on the Nexys board to start a new calculation. If you messed up and want to reset, then press the reset button

New calculation





QUESTIONS?