

TCES 430 Project Draft

Allysen Arntsen, Riley Ruckman, and Travis Shields

Components

For our project, we chose to create the simple calculator design. To do this, we needed to use our TM4C123GH6PM microprocessor, a four-digit 7-segment display, a number pad, a breadboard, and wiring as needed. We connected the number pad on ports A and D of the microcontroller, and defined buttons A, B, C, and D for addition, subtraction, multiplication, and division, respectively. The 7-segment display is on ports B and E.

Draft Operation and Progress

We have made a significant amount of progress on this project overall. It is still not yet entirely functional, but we do have large portions of it working. We have found that it works generally as expected with four-digit inputs, but it does not yet function properly with fewer digits. We know that this issue is a result of using a counter variable to get inputs, but using an operator as the delimiter has posed numerous problems. It is also currently impossible to enter the same digit twice consecutively, this is because the only way to monitor for new input is to check if the key variable has changed. We have plans to fix this by implementing interrupts upon key presses. For now, it requires a four-digit input, followed by an operator, followed by another four-digit input. At this point, all four operations appear to work as expected as far as they've been coded. Division is not yet capable of displaying decimal values, and it instead just results in the whole part of the quotient. We also have not yet handled the error case in which the result is too large to fit all four digits on the display. Because of this, results for large addition or multiplication operations appear to be giving an incorrect result because they display a portion of the multi-digit product or sum rather than its entirety or informing the user that it cannot express values that large. We will be able to further test these operations once we have resolved the issue of the initial input requiring four digits before it will proceed. Then, we will be able to better test our individual functions with smaller and more reasonable values. Once we have the error message working as well, we can test it using large multiplicands. We have sections of code prepared that we are in the process of testing to allow the smaller operands to be inputted, and it is just a matter of continuing to test, debug, and refine those portions of code before we will have the majority of the calculator completely operational.