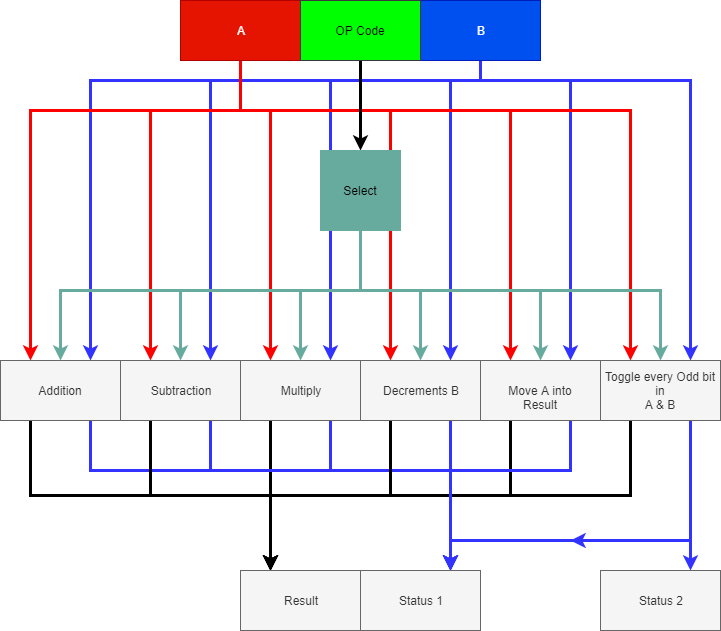
This final project implements a multitude of arithmetic calculations, which include: addition, subtraction, multiplication, decrement by one, move A into result, swab A and B, and costume operation, which is toggle every odd bit in A and B. The code deals with signed integers, standard logic vectors, and flag bits. The VHDL code is converted and transferred so it can be run from SD card and a C code is written to assign values to the registers.



**ALU:**

Takes in 3 inputs and outputs 3 signals. It takes in Register A, Register B and OP code value. A and B go is used in each case. OP code is used as a select register, and in VHDL is shown through a case statement. Once an OP code is received, the ALU chooses the correct statement and outputs values.

**Addition:** Take signed value of A and B and added them together. The added value is placed into a 33 bit signal to account for any overflow if the two values reach the max negative or positive number. Then value is placed into result. Status bits are read from the 33 bit register and placed into the status 1 bit.

**Subtraction:** Subtraction is the same as addition.

**Multiply:** Takes value A and multiplies it with B. The output register needs to be a 64 bit signal. Values A and B stayed signed and flag bits are set and read from the ALU Result vector.

**Decrements B:** Takes B and decrements it by 1 and places it into the ALU result.

**Move A into Result:** Takes A and places it into the ALU result.

**Toggle every Odd bit in A & B:** Takes A and B and XOR them both with x”AAAAAAAA” to toggle every odd bit of each register. Then they are placed into the first 32 bits and second 32 bits of ALU result. The status bits are read from the toggle A and B signals. A status bits are placed into Status 1 and B status bits are placed into Status 2.

**Conclusion:**

I overall enjoyed the project. The hardest part was figuring out errors that showed up in the VHDL code and then waiting on compiling. One thing that I missed in the beginning was that everything you made a change to your wrapper, you had to re-analyze the file in platform designer. This just created a lot of waiting for compilation. The Linux part of the project was a bit harder to understand. It was straight forward once it was understood how the components worked. The major frustrating point was waiting for everything to compile, to only find an error 4 minutes in.