CE 6303.001

HW5 Report

**Luke Allen** NetID: laa170930

**Yu Feng** NetID: yxf160330

09/28/2021

# Problem Description:

The problem we were tasked with for Homework 6 was to implement the same algorithm from Homework 5 in Verilog. The C code from Homework 5 implements the more efficient convolution procedure that mentioned in the MSDAP paper. We are to use this code as a reference while implementing the algorithm in Verilog.

# Algorithm Implementation:

## Modules:

For our implementation, we are only using one module which acts as a finite state machine.

## Inputs/Outputs:

The inputs and outputs of our MSDAP module are as follows:

reset: Resets the memory and inputs of the module to 0

clk: clock signal

inData: input Data

validData: Tells the module whether or not to use the data on the inData bus

inCoeffSign: sign of the current coefficient

readyForData: tells the controller it is ready for new data

readyForCoeffSign: tells the controller it is ready for new coeff sign

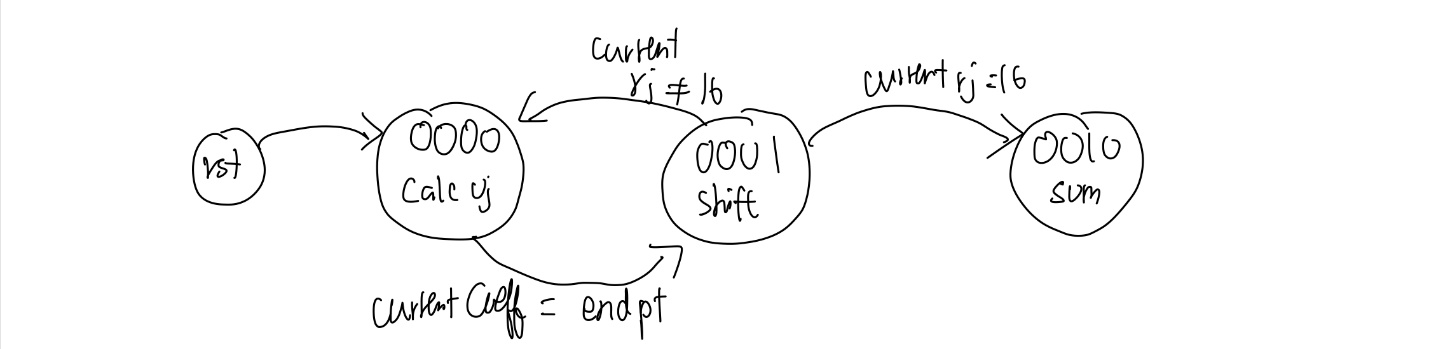
currentUjOut: current calculated Uj

overallResult: Outputs the calculated result

status:

## Finite State Machine:

The main module of our code implements the following finite sate machine:



*Figure 1: Diagram demonstrating the operation of our finite state machine.*

Sate 0000 – Calculate Uj: Performs the summation step of calculating Uj

State 0001 – Shifting: Performs the necessary shifting.

State 0010 – Ready: Output is calculated and ready.

## Simulation Results:

Shown below are our simulation results:

Graphical user interface

Description automatically generated

*Figure 2: Testbench Simulation Results Waveform*