Rodrigo Becerril Ferreyra
CECS 301 Section 03
Lab 5
Due 14 May 2020

1 Introduction

In this lab, we were tasked with creating a state machine that would control various logic (supplemental logic) that would be used to test and exercise all addresses of a 32 Kib × 16 b memory. Specifically, we were only tasked with creating the state machine, as the supplemental logic was given, and the memory was obtained by using the Vivado IP Catalog library included in most versions of Vivado (I am using Vivado version 2016.2). The state machine runs a total of four times, and fills each address of the memory with

1. the address number.
2. the bitwise NOT of the address number.
3. hex value `0x5555`.
4. hex value `0xaaaa`.

The contents of the address is checked after each pass. If an error occurred in the process of loading the addresses, then an error bit is raised. This can help with debugging the memory by tracing which register was being loaded at the time the error bit was raised.

2 State Machine

The state machine was the main part of the lab, and the piece of the circuit that performed the most work. It consists of eight states, five inputs (including `clk` and `rst`), and four outputs. The basic function of the state machine is documented by a state transition diagram. The state machine was programmed using a modified Moore model, which means that the outputs are not dependant on the inputs.