FINAL PROJECT REPORT

Name:	Padmanabha Nikhil
UnityID:	pbhimav
StudentID:	200535343

Delay (ns to run provided provided example): 14350
Clock period: 100ns
cycles": 1435

Logic Area: 143940.5809 (um^2)

Memory: N/A

1/(delay.area): 4.84×10^{-10} ($ns^{-1}.um^{-2}$)

Delay (TA provided example. TA to complete)

1/(delay.area) (TA)

Abstract

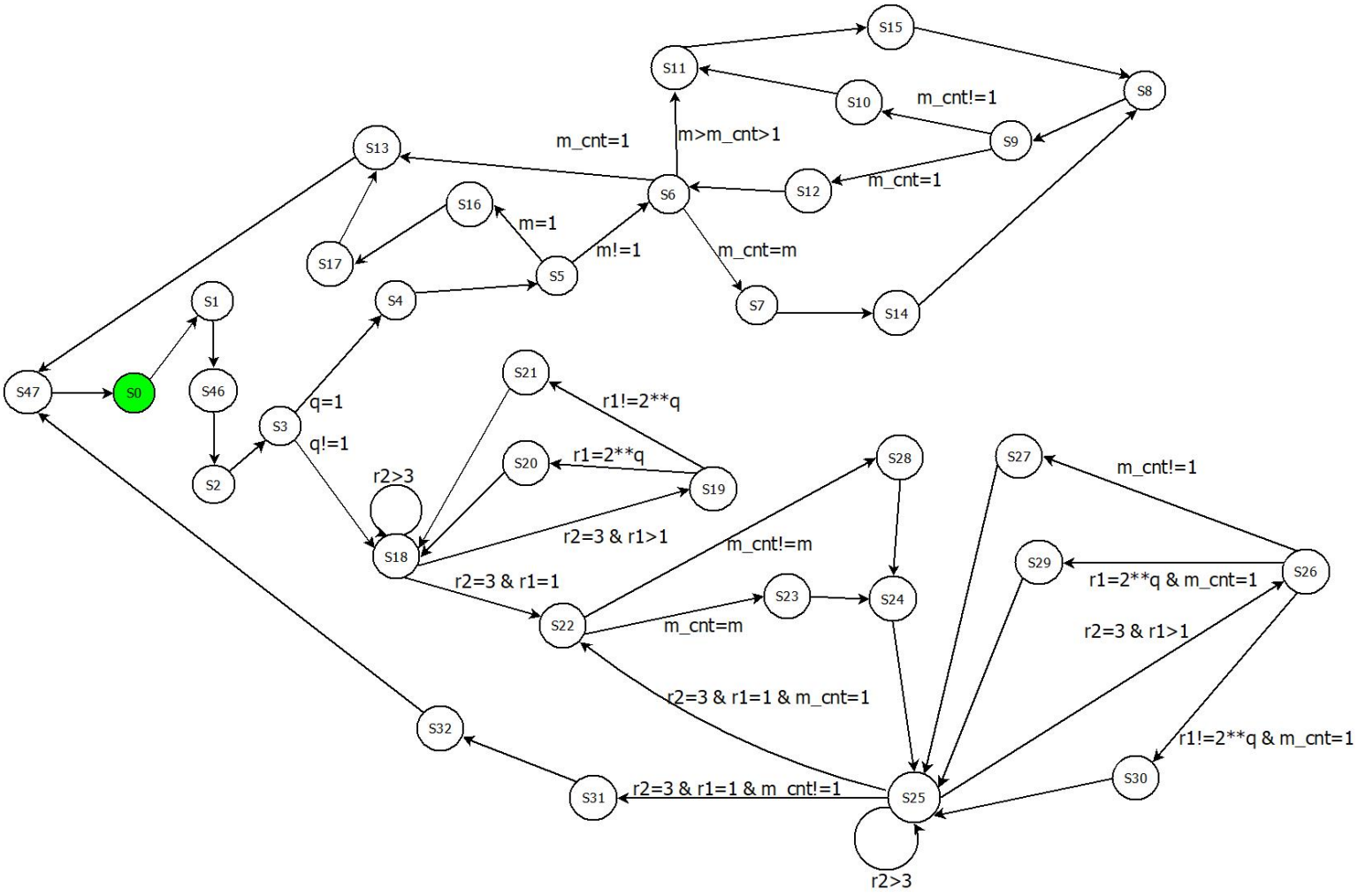
Quantum computing is one of the most trending technologies in the modern world that employs qubits, which may represent several states at once due to superposition. This characteristic enables quantum computers to do certain computations quicker than conventional computers. The strength of quantum computers is humungous and lies in the fact that n qubits can represent states up to the 2n. The project is to conduct an experiment on quantum emulators where we multiply **m** number of (2^q X 2^q) **q gate** matrices with a (2^q X 1) **initial state vector** matrix. The **q** represents number of qubits employed by the the quantum computer. The q gate and the q state vector matrices consist of complex numbers in form of IEEE754 Double Precision Floating point numbers. The matrices elements are stored in two different SRAMs named as **q_state_input_sram** for the final and **q_gates_sram**. The other two SRAMs are q_state_output_sram for the final state vectors of the output of q state values and scratchpad_sram for storing intermediate values in matrix multiplication.

1. Introduction

- The data flow in the project consists of 12 multiplexers and 8 Design Ware Multiply and Accumulate Blocks.
- The cycles achieved (Test 6) are 1435ns
- The report consists of the high-level design (FSM) and technical Implementations

2. Micro-Architecture

- High level architecture drawing, and description of data flow



3. Interface Specification

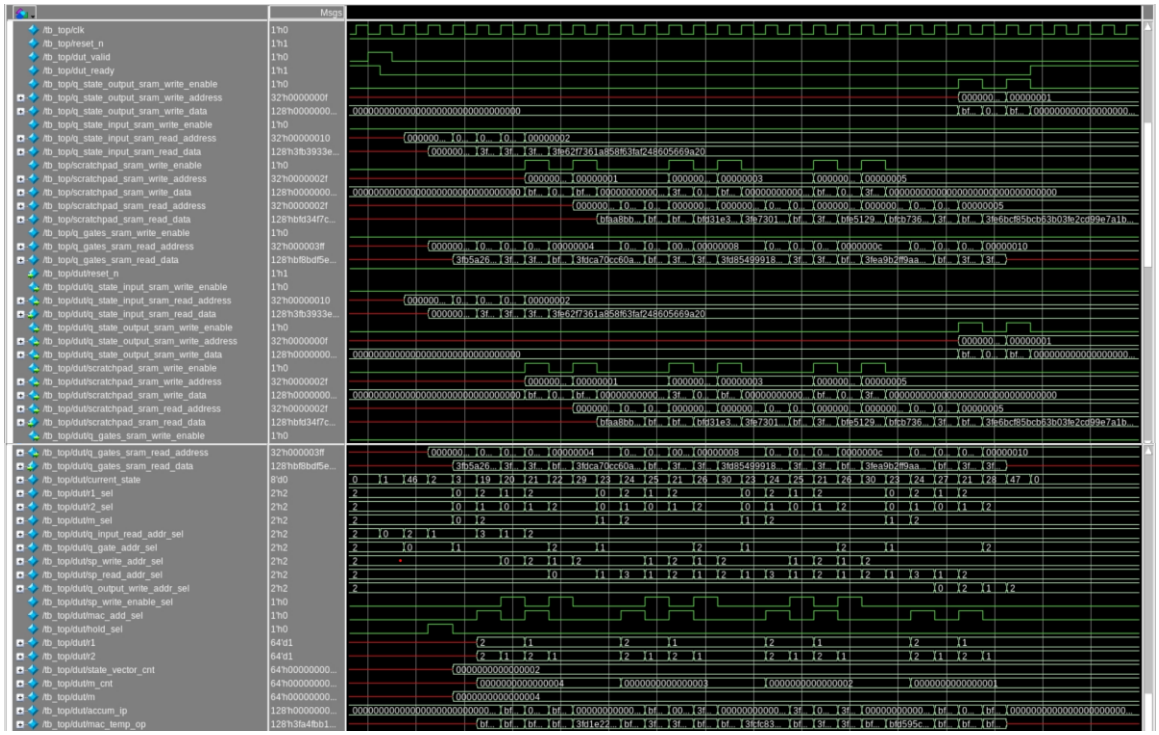
Control lines (select lines for mux)	Width	Function
r1_sel	2	Controls the count of q gate matrix left
r2_sel	2	Controls the count of initial state vector matrix left
m_sel	2	Controls the count of the number of matrices left parameter
q_input_read_addr_sel	2	Controls the address of the q_state_input_sram for data read
q_gate_addr_sel	2	Controls the address of the q_gate _sram for data read
sp_write_addr_sel	2	Controls the address of the scratchpad_sram for data write
sp_read_addr_sel	2	Controls the address of the scratchpad_sram for data read
q_output_write_addr_sel	2	Controls the address of the q_state_output_sram for data write
sp_write_enable_sel	1	Controls the write enable for the scratchpad_sram
mac_add_sel	1	Controls the multiply and accumulate output
q_output_write_enable_sel	1	Controls the write enable for the q_state_output _sram
hold_sel	1	Controls the value of the ‘q’ and ‘m’ parameters

4. Technical Implementation

A module of Design Ware Multiplier and accumulator is being used for the operator matrix multiplication. Each (2^q X 2^q) q_gate matrix will be multiplied with the (2^q X 1) q_state_input matrix to obtain the final state vector (q_state_output) matrix.
The design of this module will contain 4 MAC units for the multiplication of all operator matrices and 4 MAC units for multiplication of the operator matrix product and q_state_input matrix.

The subsequent product from each matrix will be stored in the scratchpad SRAM for the next multiplication. The scratchpad SRAM will be updated after every operator matrix multiplication the last matrix multiplication which will be written in the q_state_output_sram.

The test run (Test-6) was executed for 1435 cycles. Below is the timing report for the test case 1 (q=1; m=4) of Test 6:



5. Verification

- The design has been tested with 6 different tests each with different q & m parameters.
- The design has also been synthesized and it has been made sure that the slack is met. The area and timing reports have been submitted with the report.

6. Results Achieved

- Throughput, area, power/energy (if applicable), etc.
- Area: 143940.5809
- Clock Period: 100ns

7. Conclusions

The project has allowed me to gain knowledge about the basics how the quantum computing works with the qubits. The quantum computing emulator project could play a major role to the advancement of quantum computing research by providing a reliable and efficient platform for simulation and experimentation. As quantum computing advances, the emulator will serve as a basic tool for both researchers and fans, boosting creativity and knowledge in this exciting subject.