

Project 2 Report

Date 11/07/2018

Name : Emil Prisquilas Peter

Unity Email Address: epeter@ncsu.edu.

Part A

Timing Analysis

1. What is the original SPI bit rate?

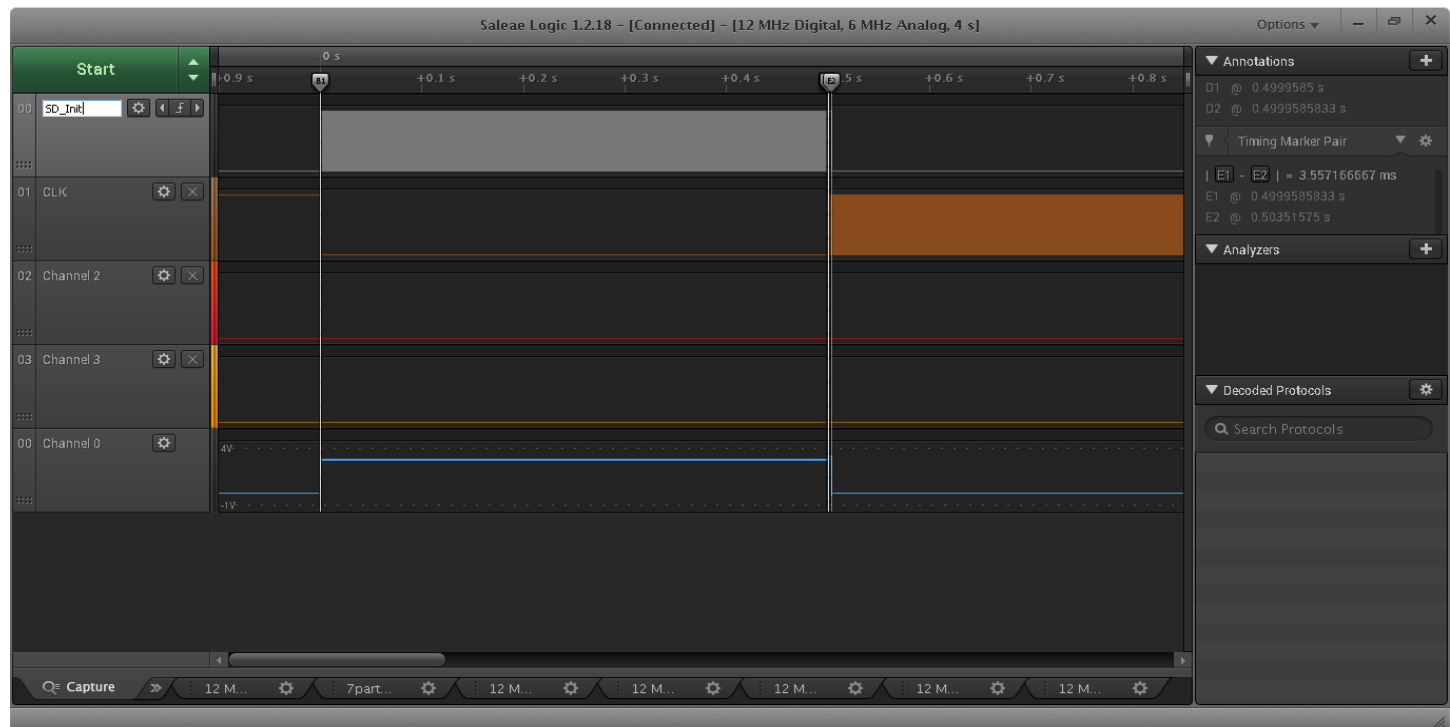
The original SPI bit rate is 1.5MHz. The value loaded in SPI1_BR is 0x04

2. What is the fastest SPI bit rate which works for your μ SD card?

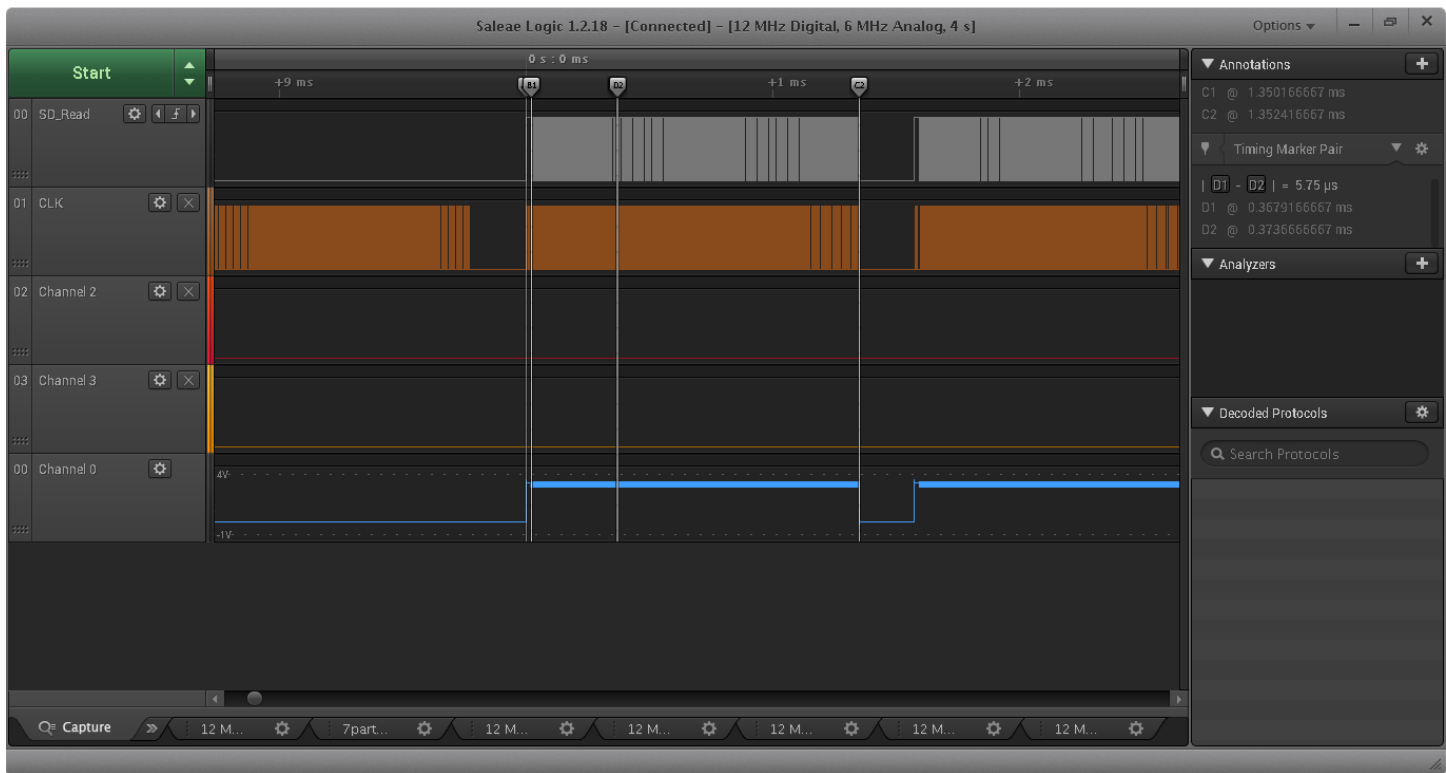
The fastest SPI bit rate which works for my SD card is 24MHz. The value loaded in SPI1_BR is 0x01.

5. Logic analyzer screen shot showing SPI and debug signals for during SD_Read or SD_Write operation (*not* SD_Init), marked with segments (Compute, I/O-SD, I/O-SPI, Other).

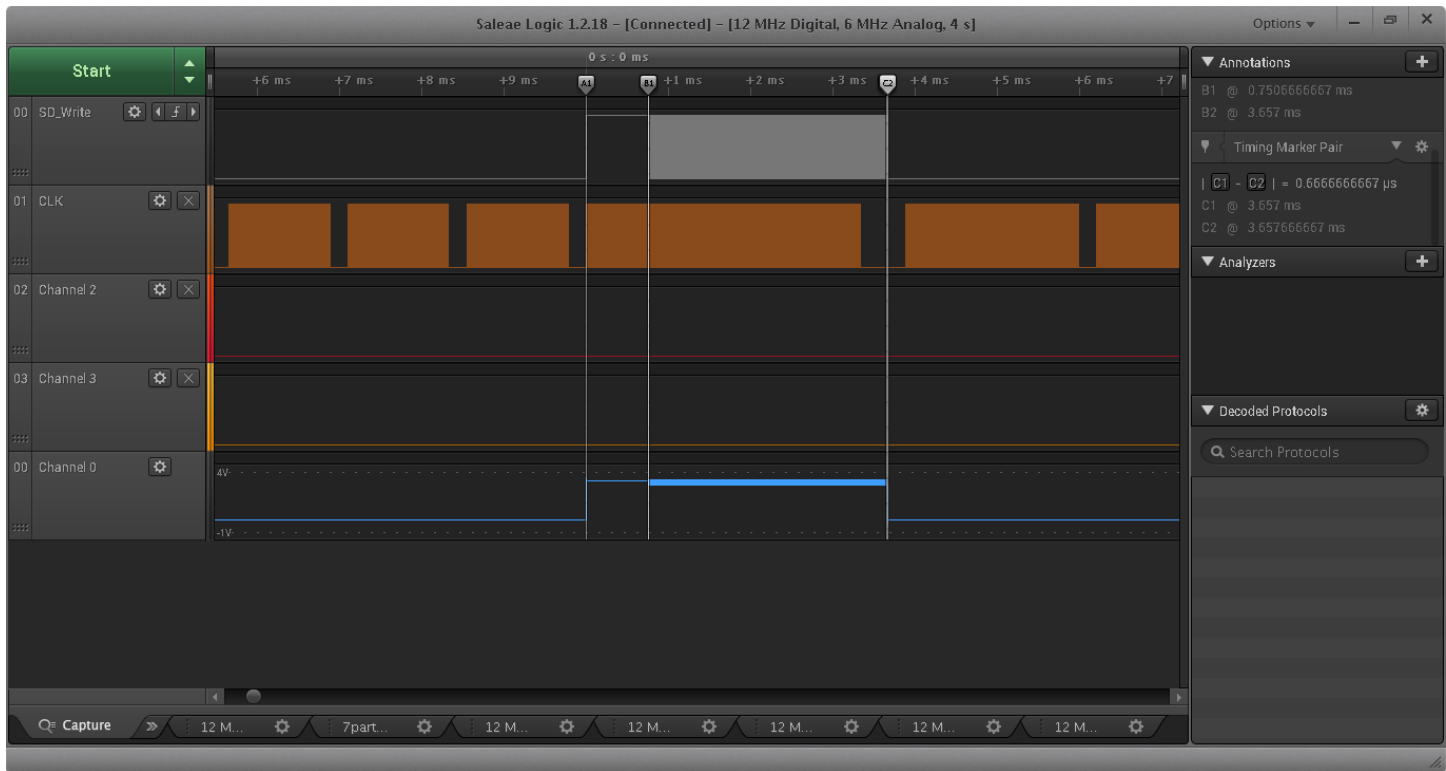
SD_INIT:



SD_Read:

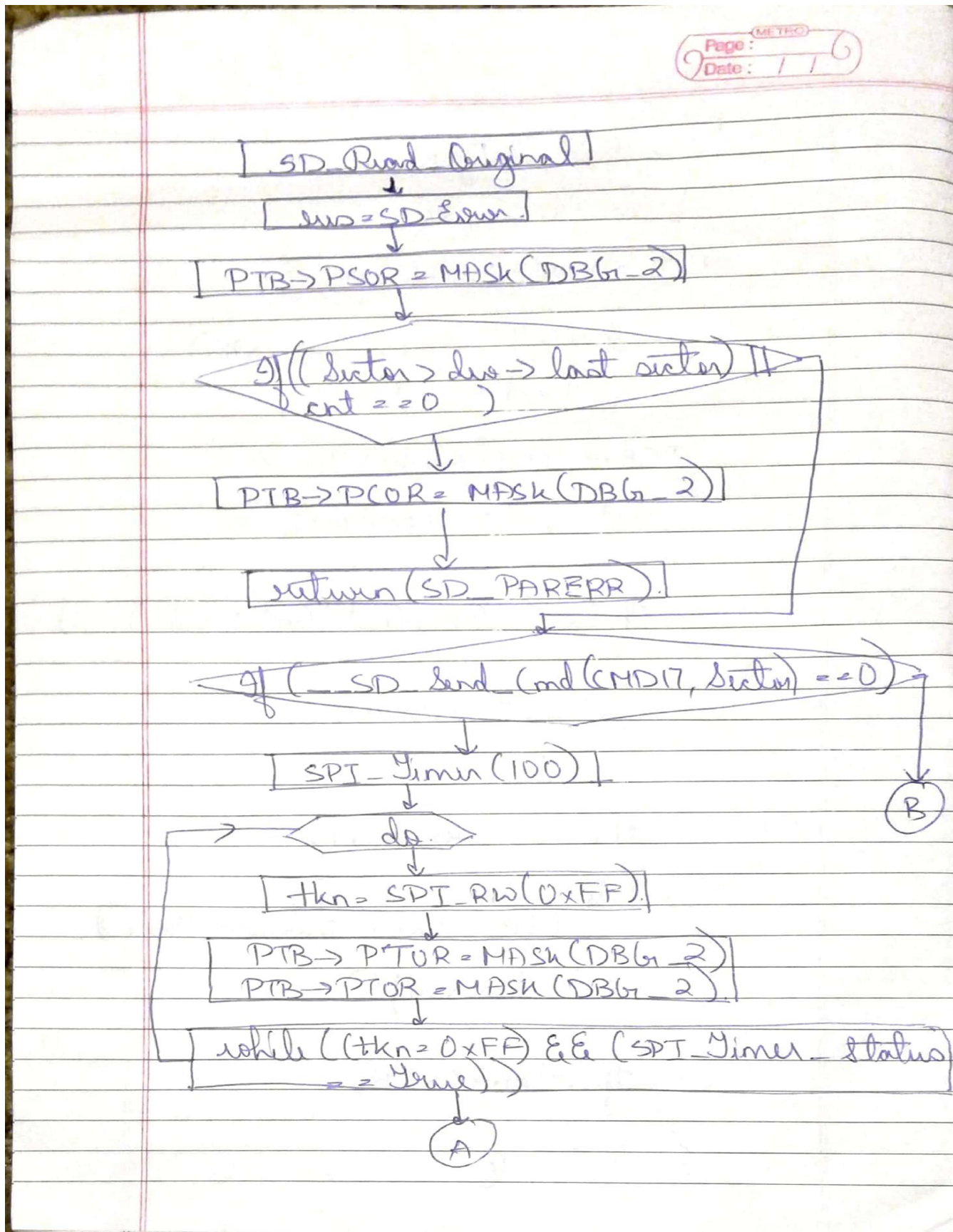


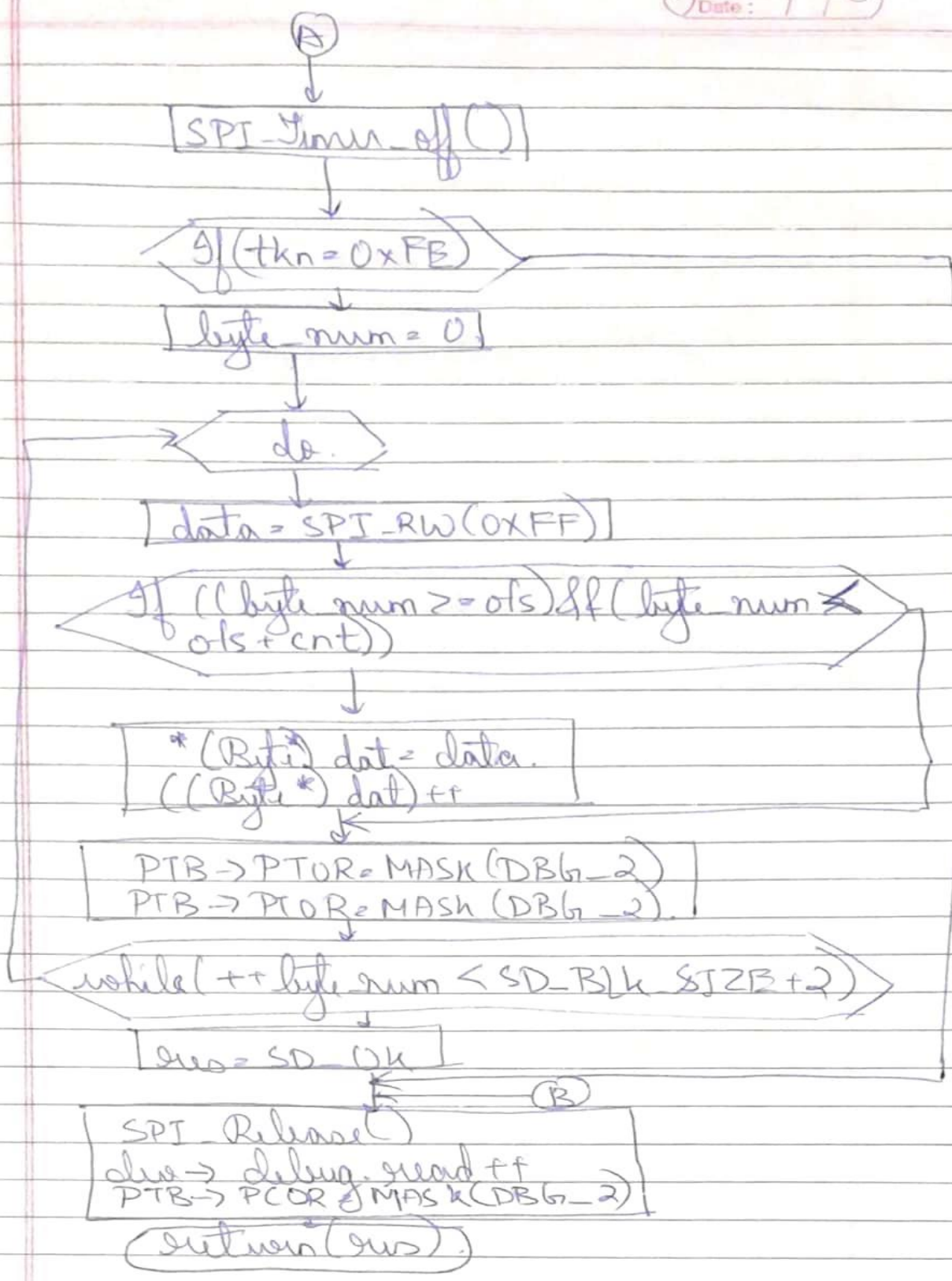
SD_Write:



Code Structure Analysis

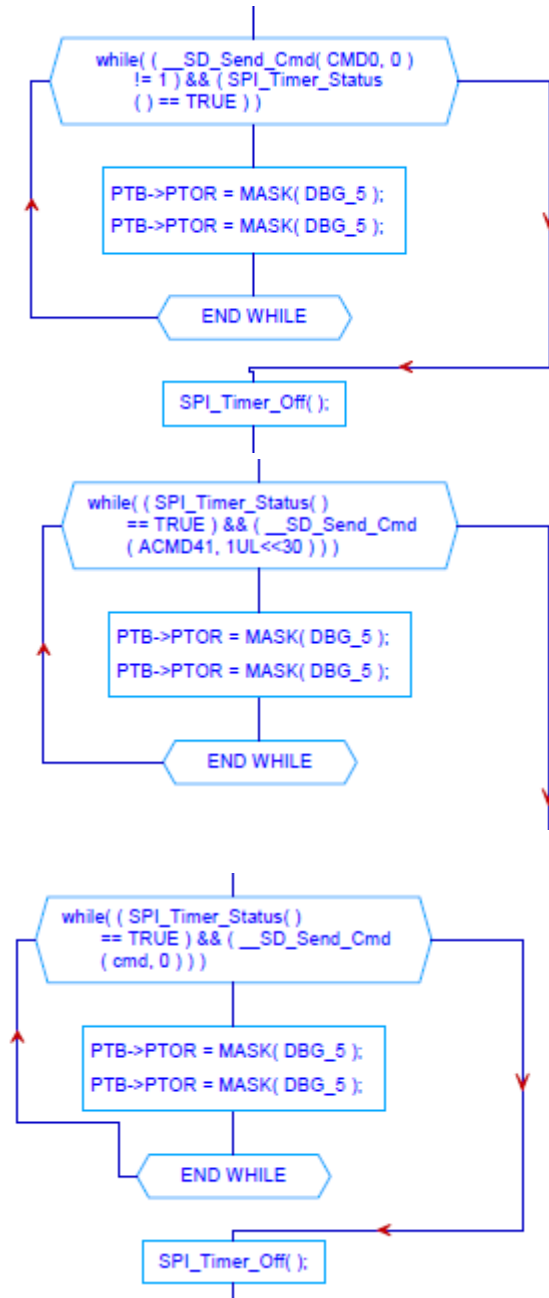
6. Control flow graph for SD_Read.



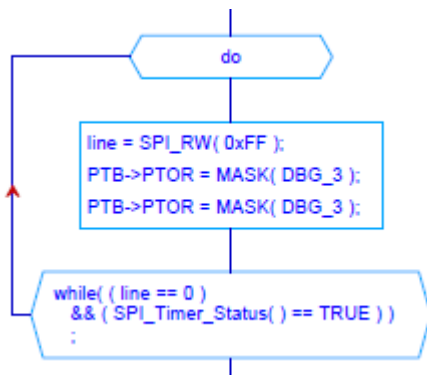


7. Control flow graphs highlighting operations which may block (i.e. repeat a loop an unknown number of times).

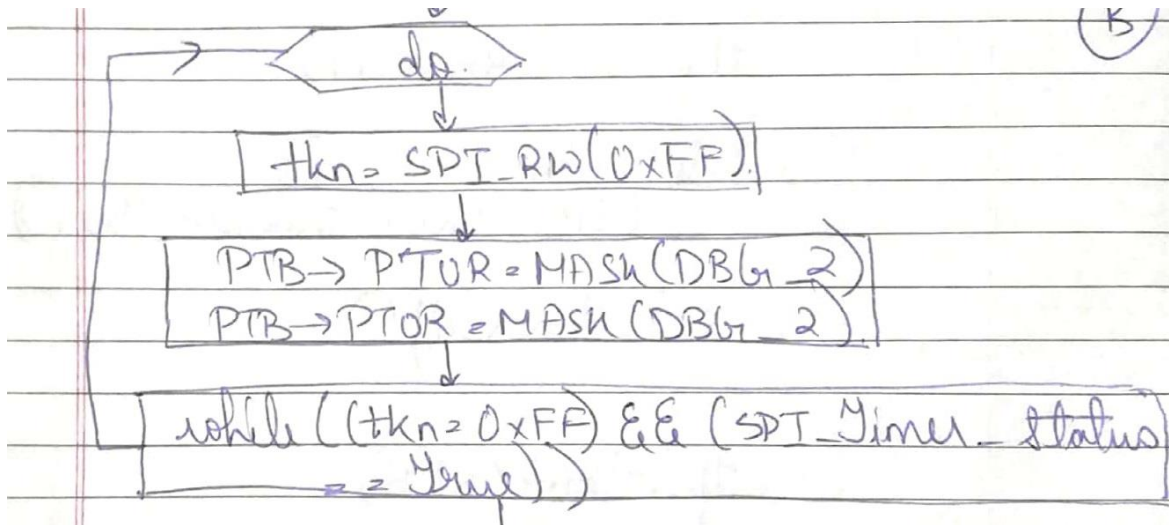
a. SD_Init



b. SD_Read



c. SD_Write



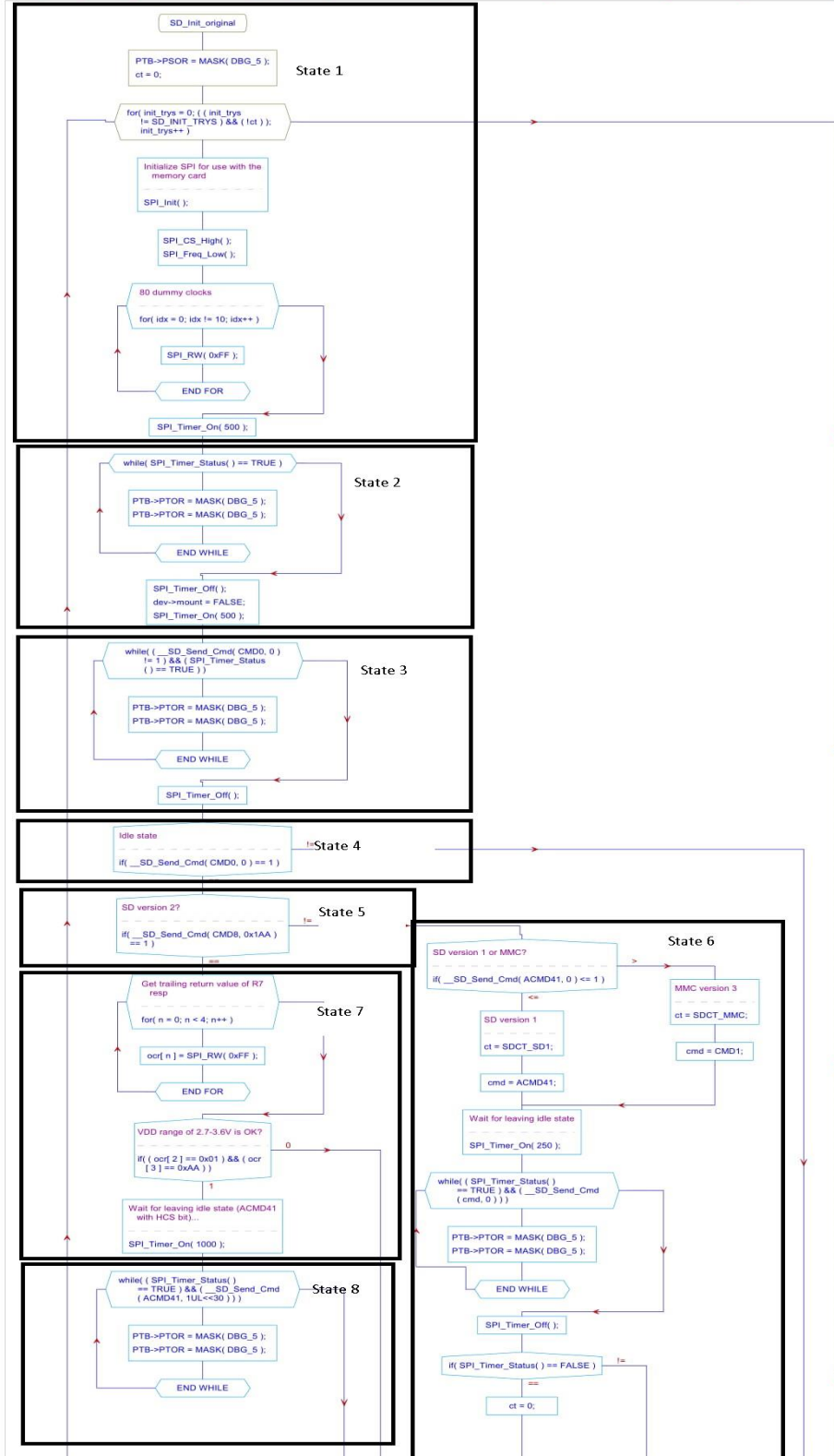
Code Transformation

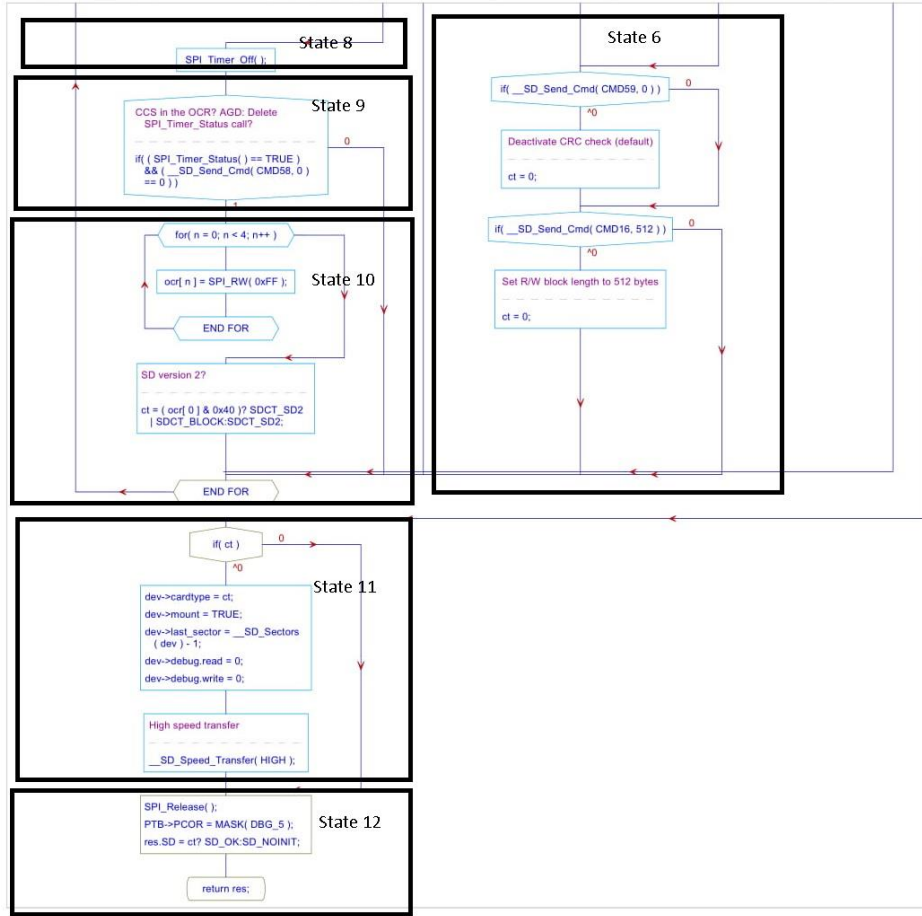
8. CFG with FSM states overlaid

a. Marked-up control flow graphs allocating code to states

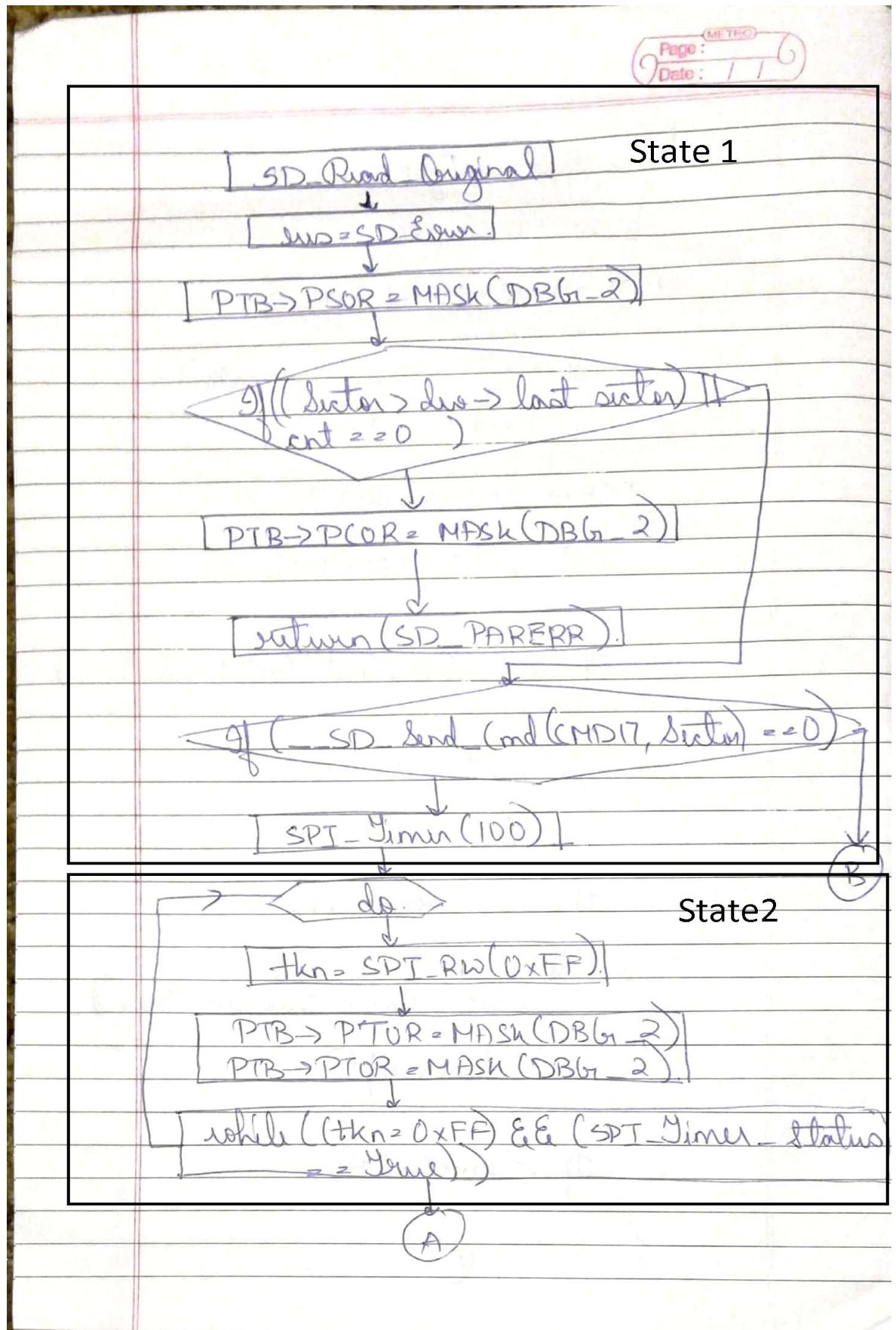
i. SD_Init

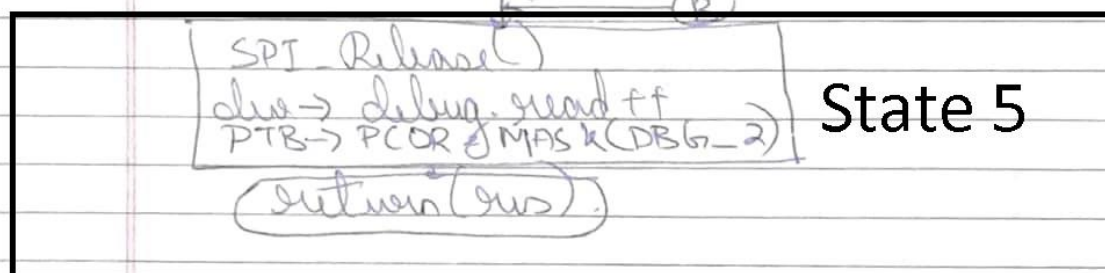
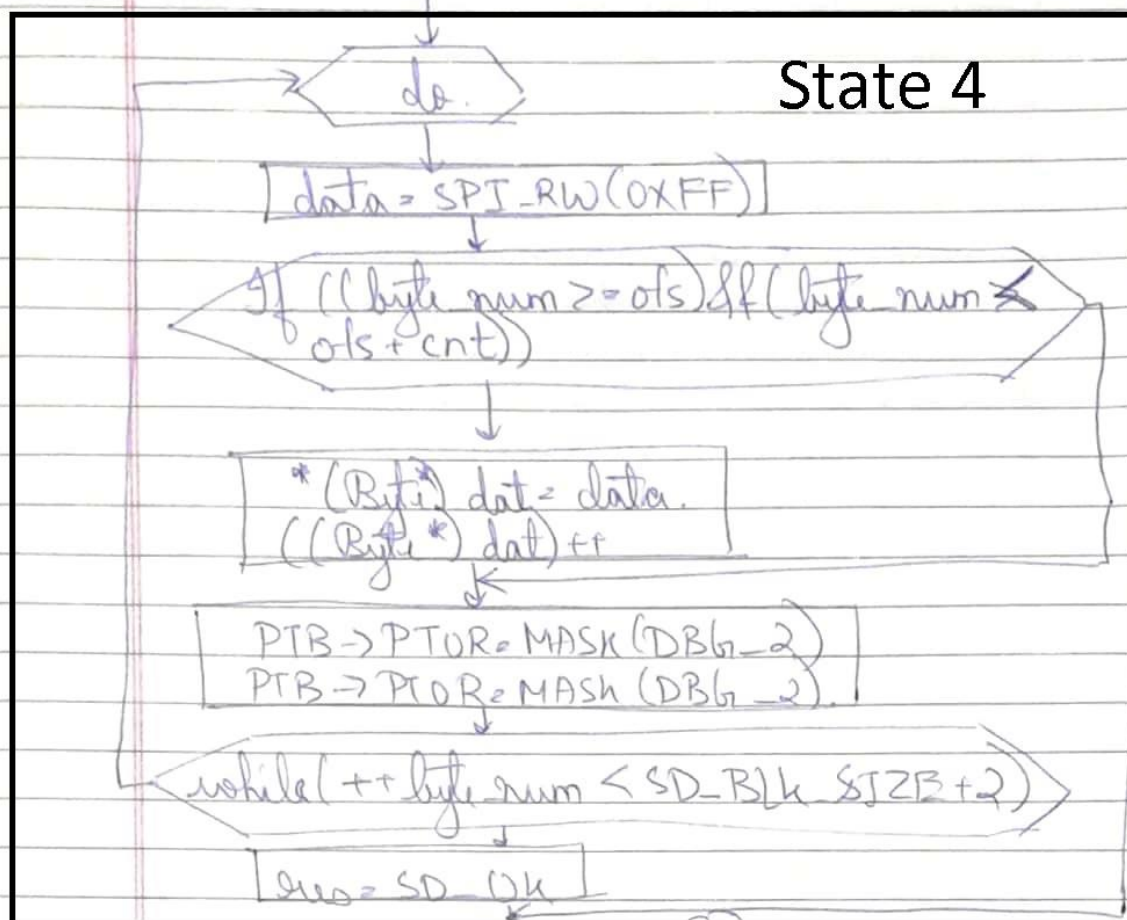
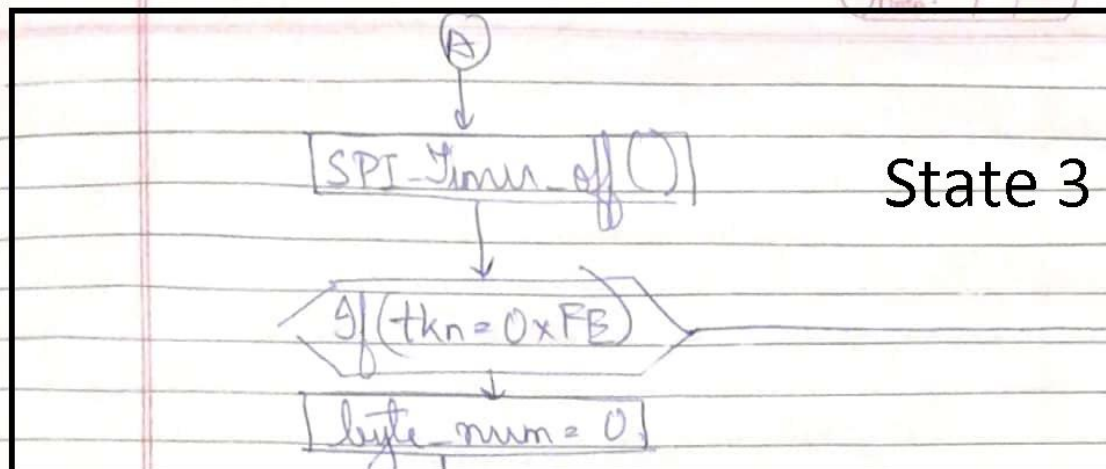
Page 1 [1, 1] function: SD_Init_original file: sd_io.c Date: [16:09:2017] Time: [13.59.05]





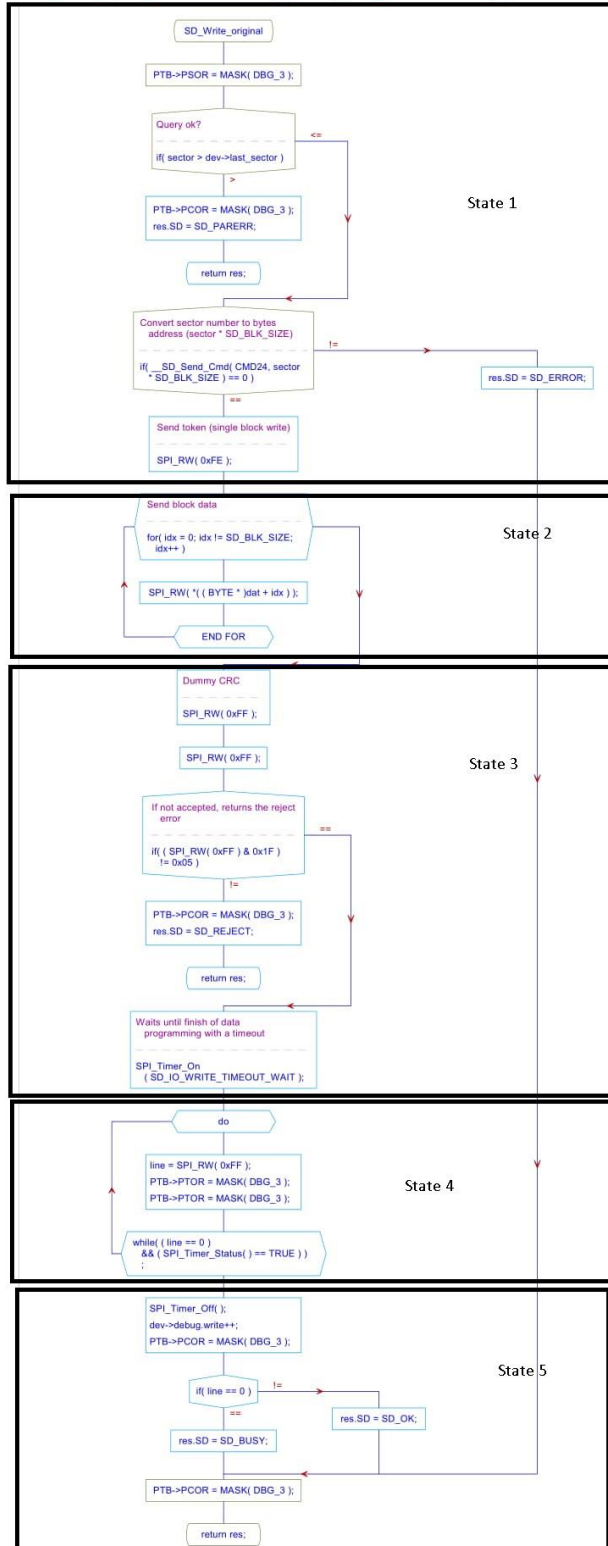
ii. SD_Read





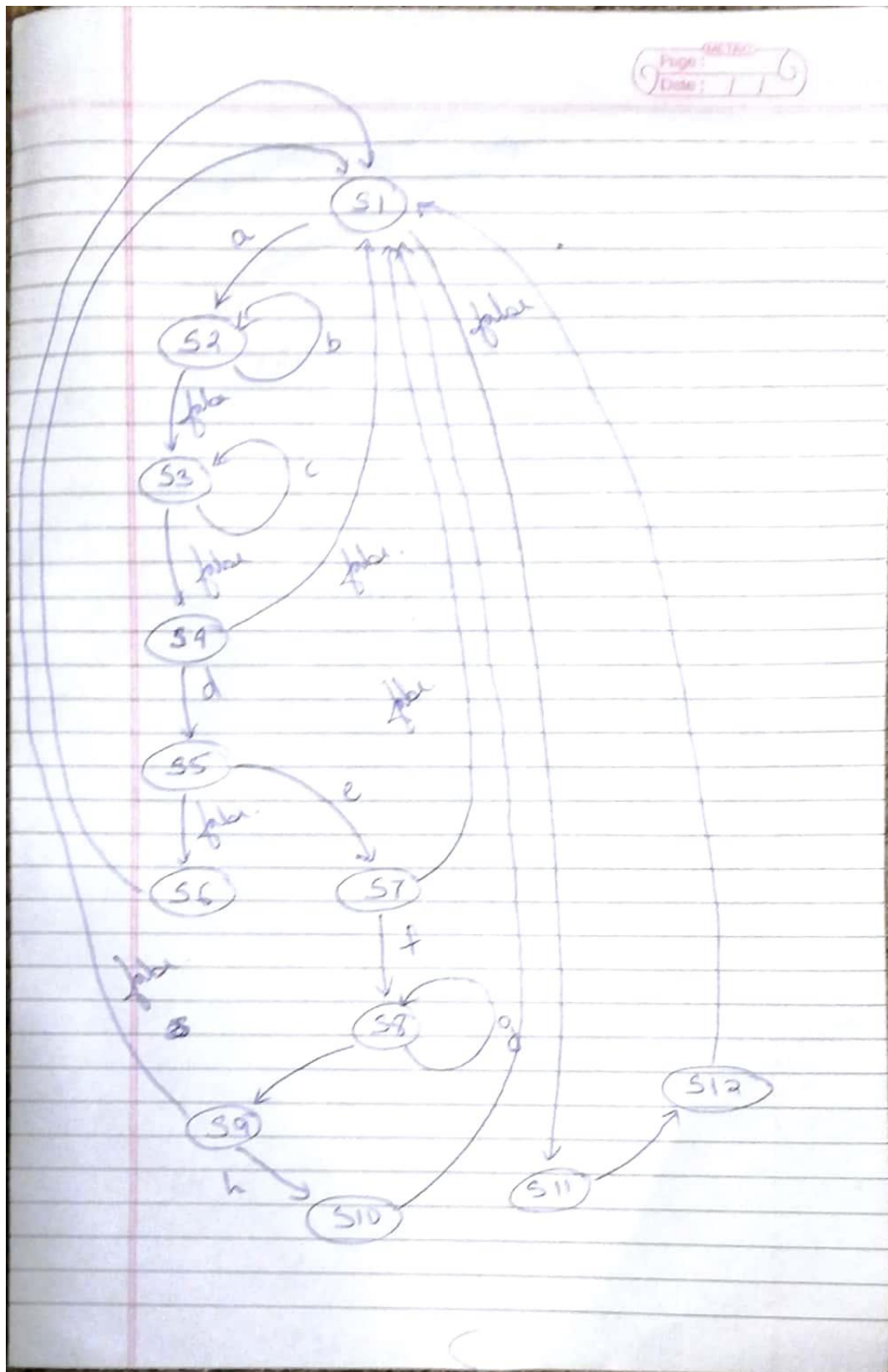
iii. SD_Write

Page 1 [1, 1] function: SD_Write_original file: sd_io.c Date: [16:09:2017] Time: [14.01.17]

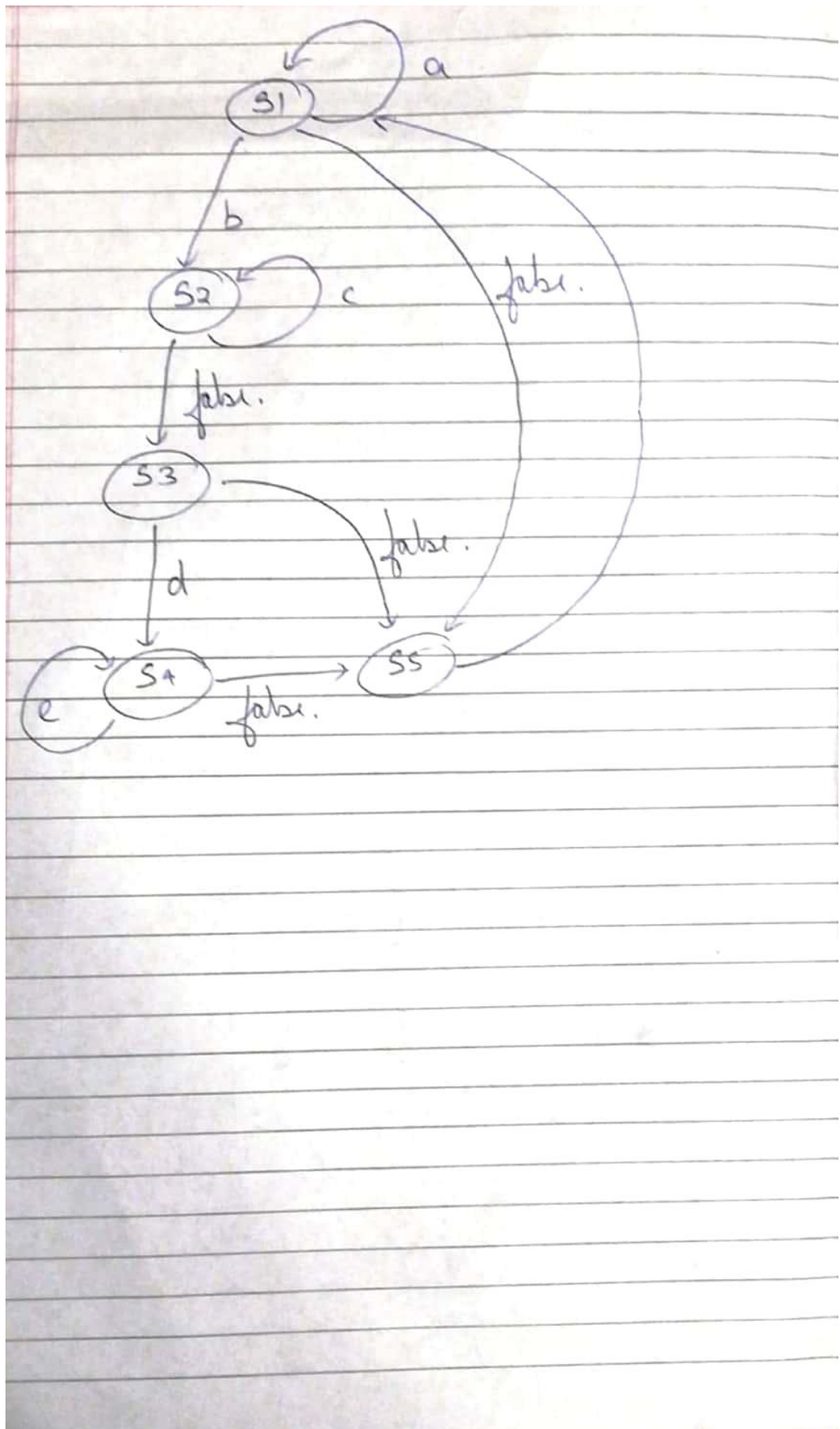


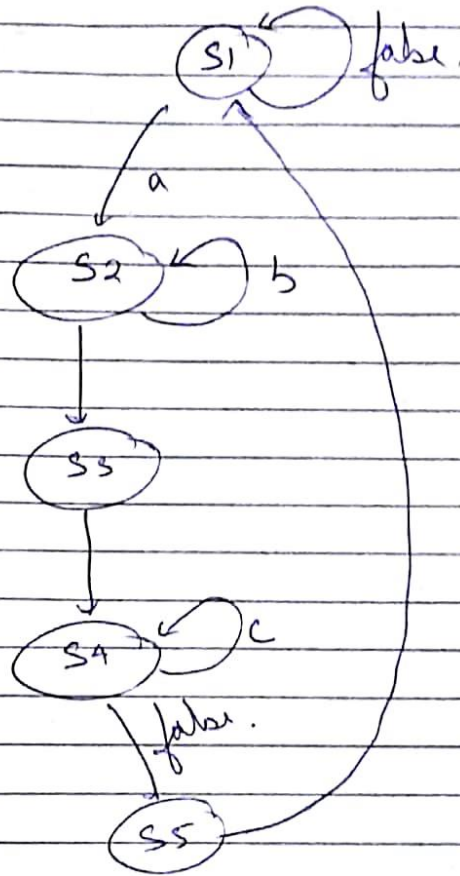
9. FSM diagrams

- i. State diagram with both states and transitions labeled
SD_Init



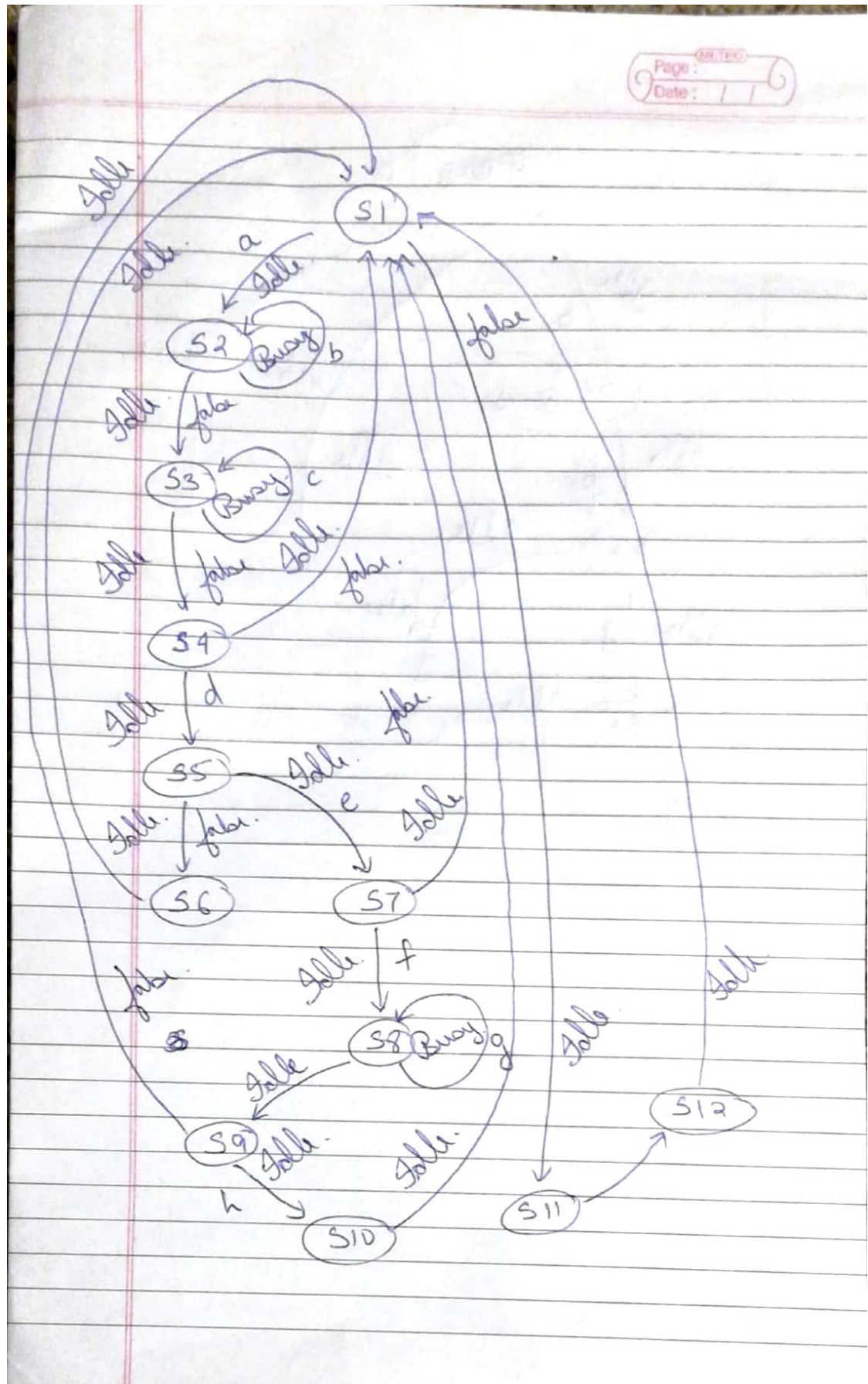
ii.



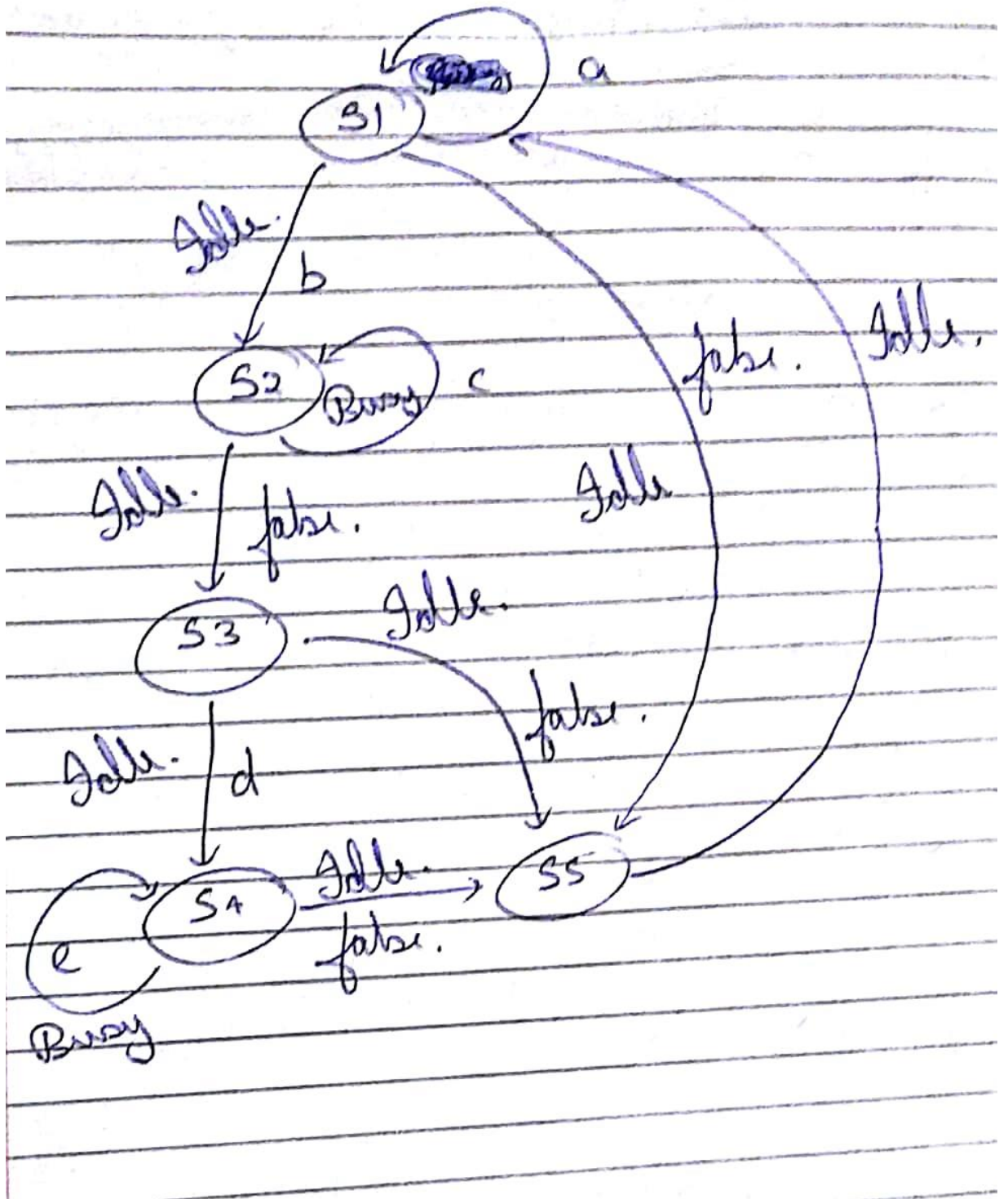


b. State diagram showing which states are busy and which are idle.

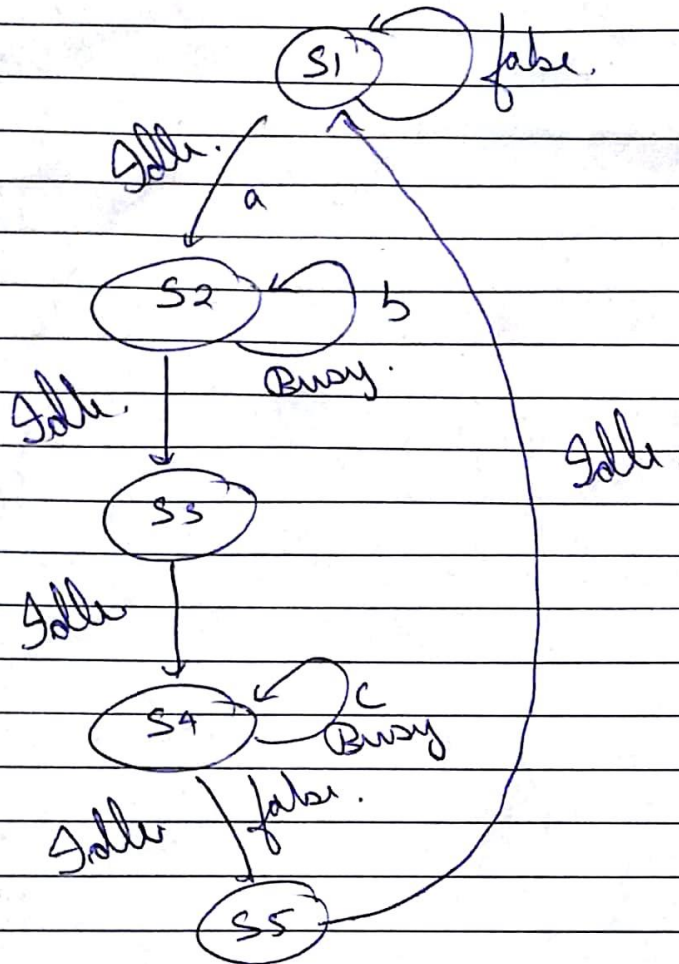
i. SD_Init



ii. SD_Read

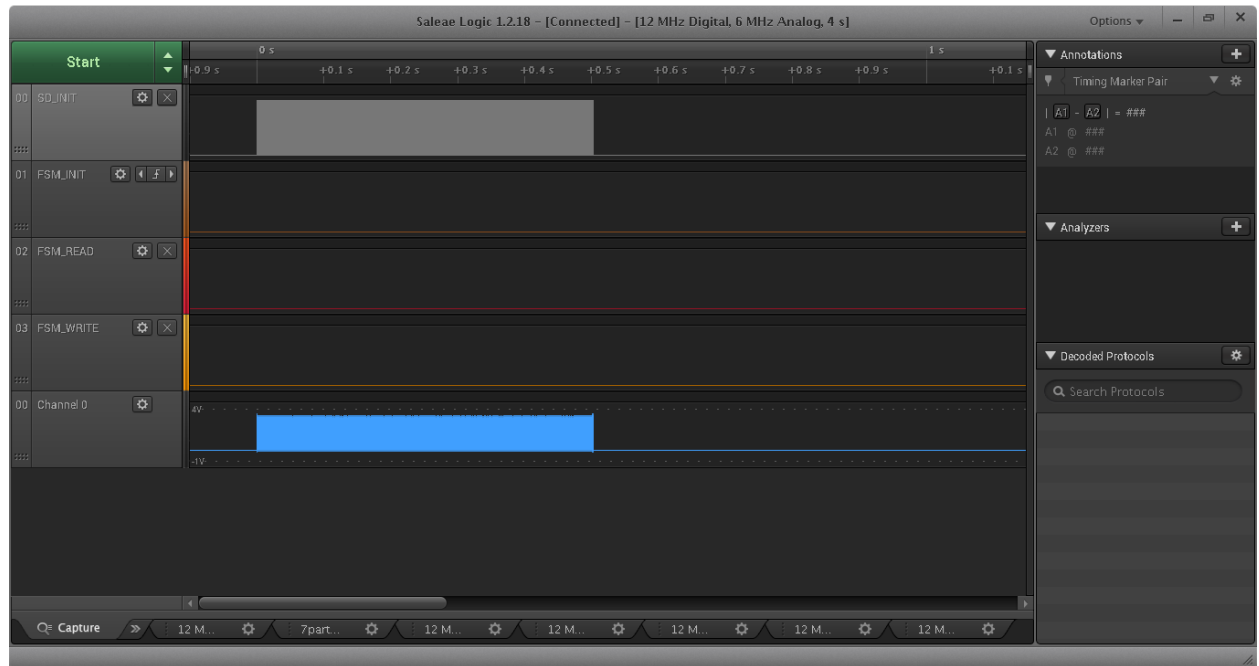


iii. SD_Write

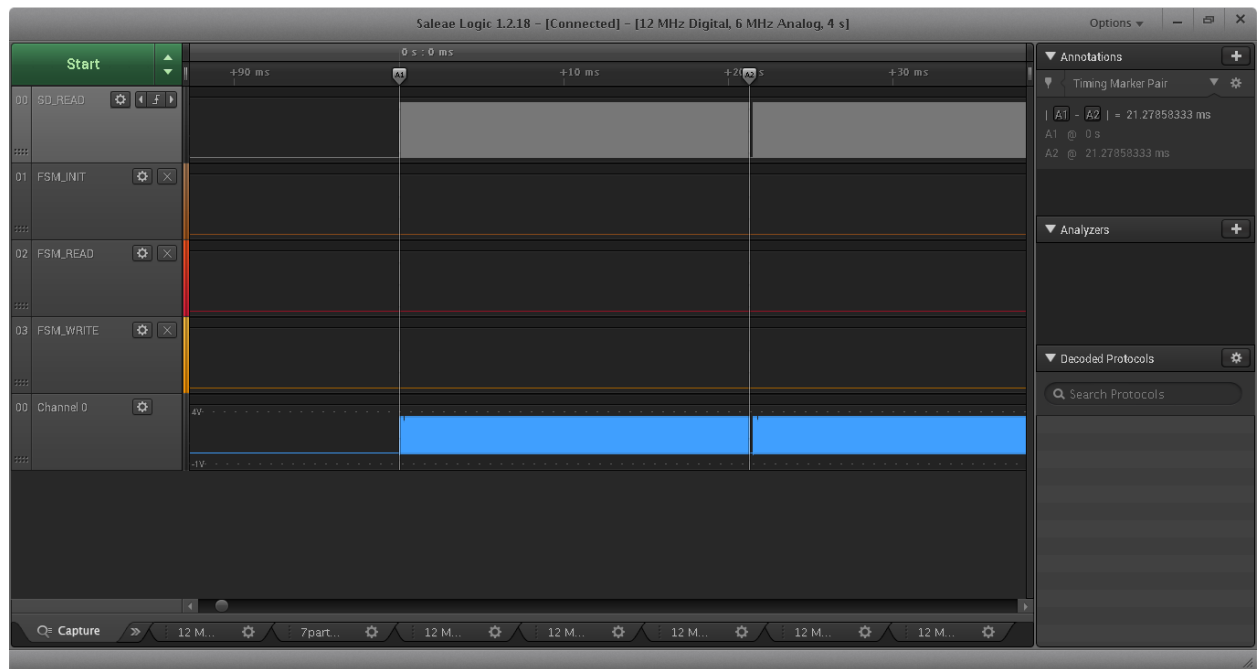


10. FSM Verification

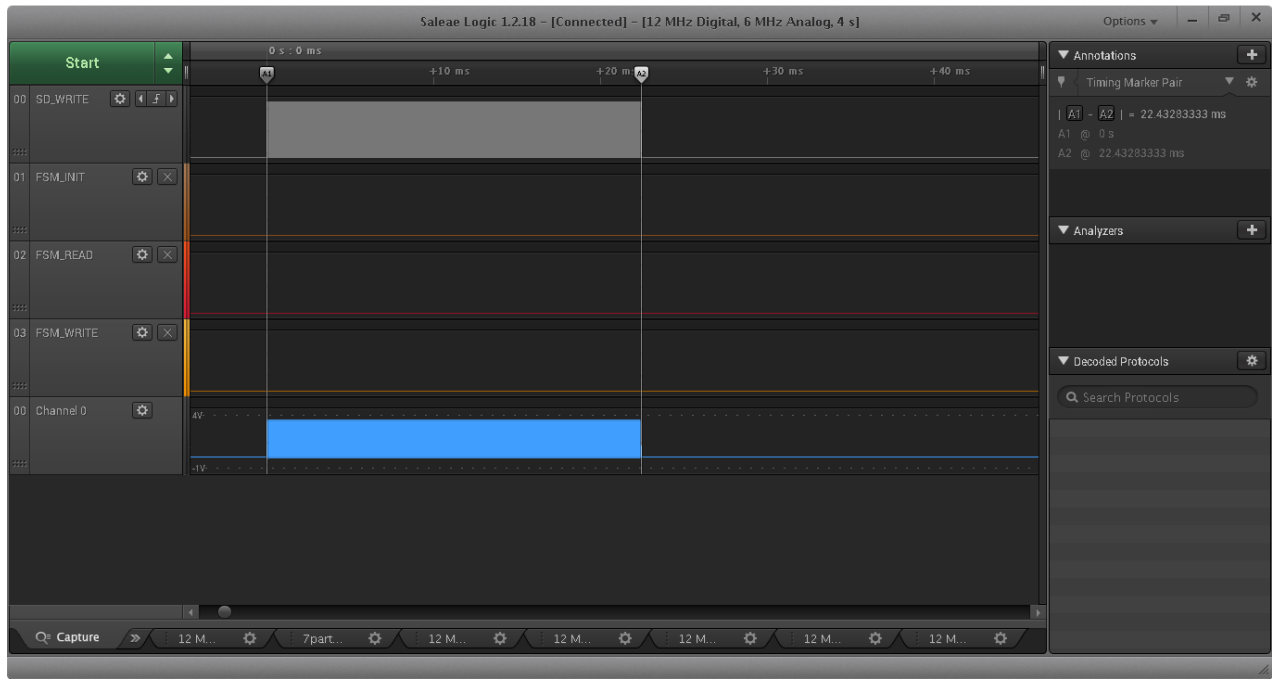
- a. Logic analyzer screen shot showing SPI signals (SPI CLK, SPI DI, SPI DO, SPI CS) and debug signals (SD_Read, SD_Write, SD_Init, test_write).
 - i. SD_Init operation



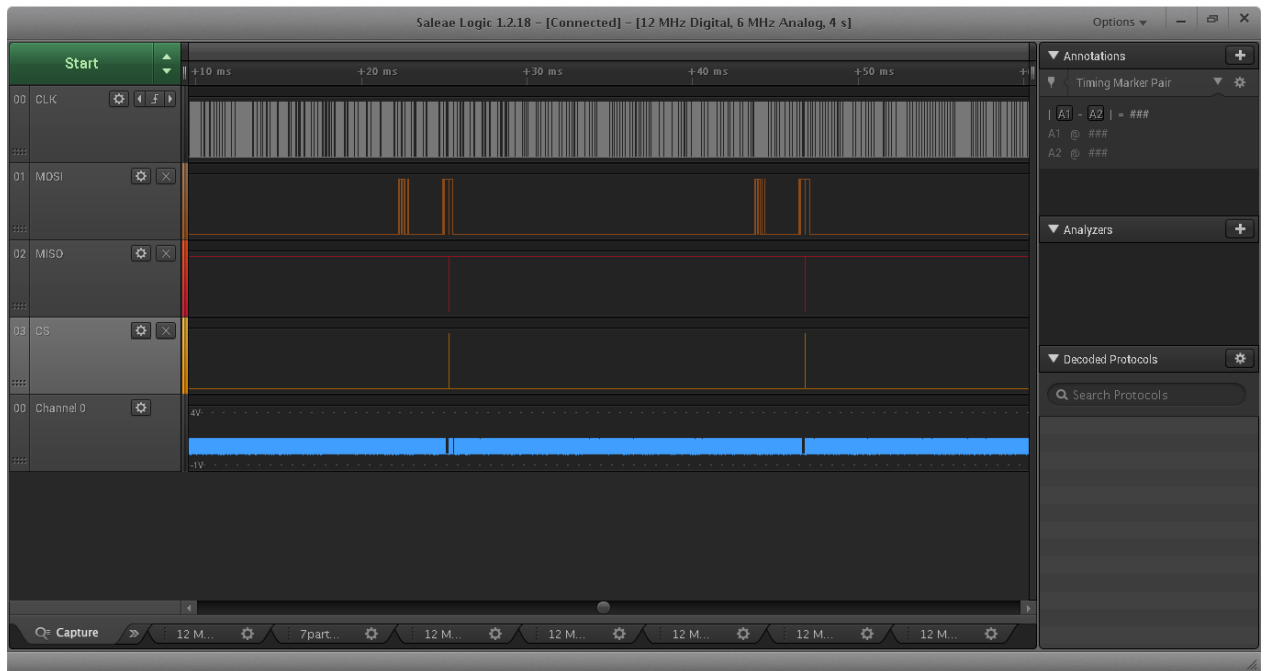
- ii. SD_Read operation



iii. SD_Write operation



SPI CLOCK, MOSI, MISO and CS signals



- b. Find and analyze the state with longest code. List the state name and the maximum execution time observed. Note: this is the state which takes the longest time to execute its code once. It is not the total time spent in the state, or the total time spent executing this state's code.
- i. **SD_Init operation**
State 1 executes for 303.8 μ s
 - ii. **SD_Read operation**
State1 executes for 19.33 μ s
 - iii. **SD_Write operation**
State 1 excutes for 19.17 μ s
- c. Table with maximum state execution times of your FSM-based code.

Function	Name and Duration of State with Longest Code
SD_Init	State 1, Duration 303.8 μ s
SD_Read	State 1, Duration 19.33 μ s
SD_Write	State 1, Duration 19.17 μ s