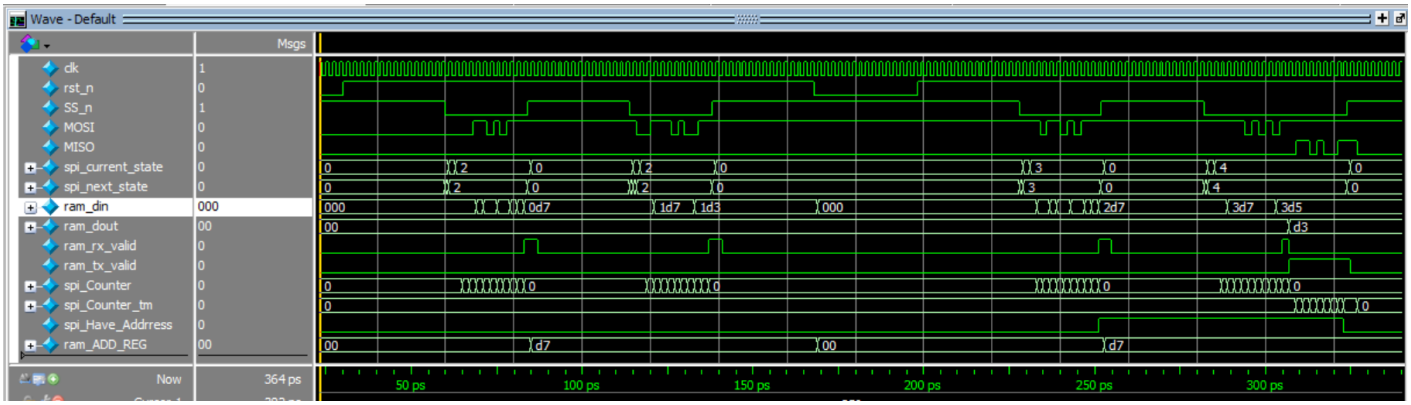
SPI Protocol

Final Project

Karim Osama Khalifa

Overview (Full Waveform)



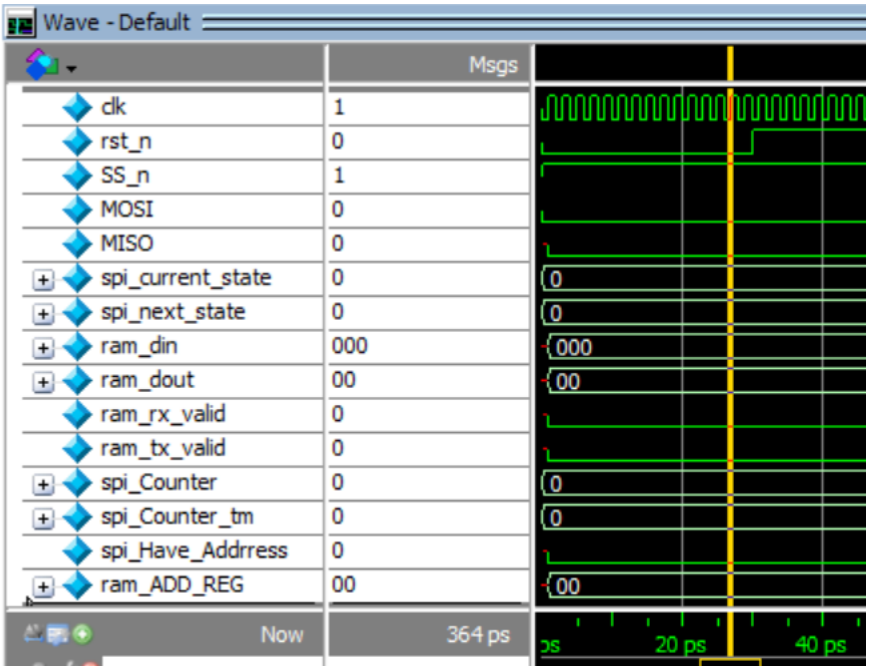
Projects Parts:

1- Clock Generation:

```
// Clock generation
    initial begin
        clk = 0;
        forever #1 clk = ~clk; // 100MHz clock
    end
```

2- Reset Test:

```
initial begin
    // Reset generation
    rst_n = 0;
    SS_n = 1;
    MOSI = 0;
    repeat(15) @(negedge clk);
    rst_n=1;
    repeat(15) @(negedge clk);
```



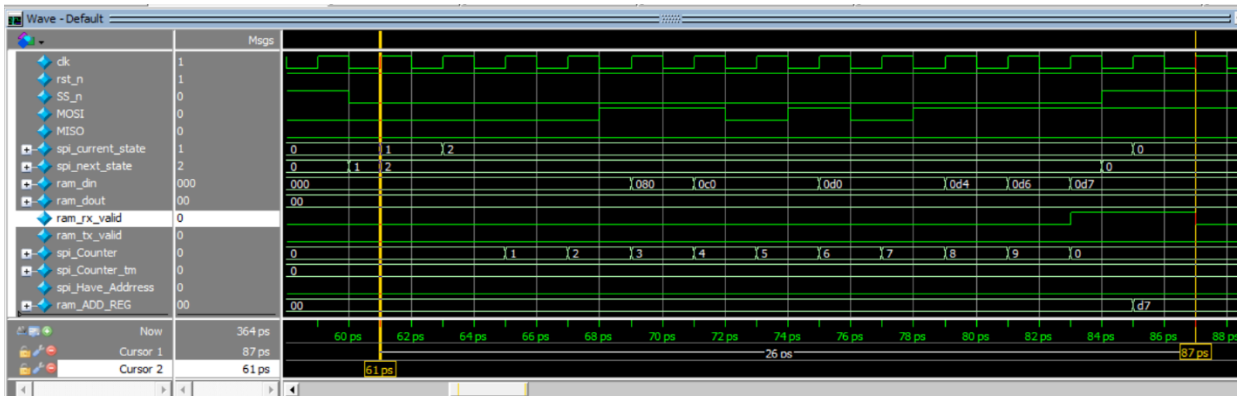
3- Write Address Case:

We give input address as 11010111(d7 in hex)

```
//60ns
        // Start SPI transaction
        SS_n = 0;@(negedge clk);

        // Send command (WRITE ADDRESS)
        MOSI = 0;@(negedge clk);
        MOSI = 0;@(negedge clk);
        MOSI = 0;@(negedge clk);
        //ADDRESS
        MOSI = 1;@(negedge clk);
        MOSI = 1;@(negedge clk);
        MOSI = 0;@(negedge clk);
        MOSI = 1;@(negedge clk);
        MOSI = 0;@(negedge clk);
        MOSI = 1;@(negedge clk);
        MOSI = 1;@(negedge clk);
        MOSI = 1;@(negedge clk);

        // End SPI transaction
        SS_n = 1;
        repeat(15) @(negedge clk);
```



Note: Start SPI @61ns – end SPI & return ideal case @87ns

Number of clocks = (87-61)/2 = 13 clock cycle ✓

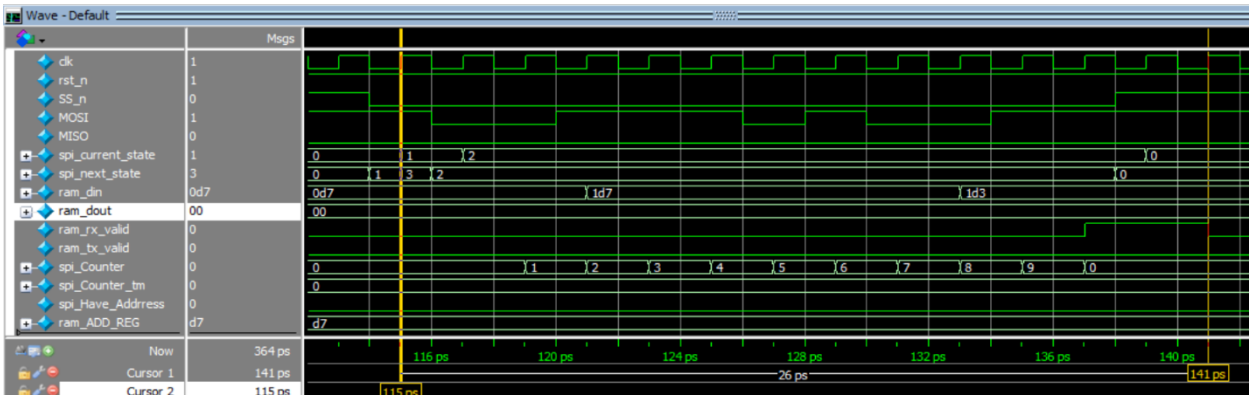
4- Write Data Case:

We give input address as 11010011(d3 in hex)

```
//114ns
        // Start SPI transaction
        SS_n = 0;@(negedge clk);

        // Send command (WRITE DATA)
        MOSI = 0;@(negedge clk);
        MOSI = 0;@(negedge clk);
        MOSI = 1;@(negedge clk);

        //DATA
        MOSI = 1;@(negedge clk);
        MOSI = 1;@(negedge clk);
        MOSI = 0;@(negedge clk);
        MOSI = 1;@(negedge clk);
        MOSI = 0;@(negedge clk);
        MOSI = 0;@(negedge clk);
        MOSI = 1;@(negedge clk);
        MOSI = 1;@(negedge clk);
        // End SPI transaction
        SS_n = 1;
        repeat(15) @(negedge clk);
```



Note: Start SPI @115ns – end SPI & return ideal case @141ns

Number of clocks = (141-115)/2 = 13 clock cycle ✓

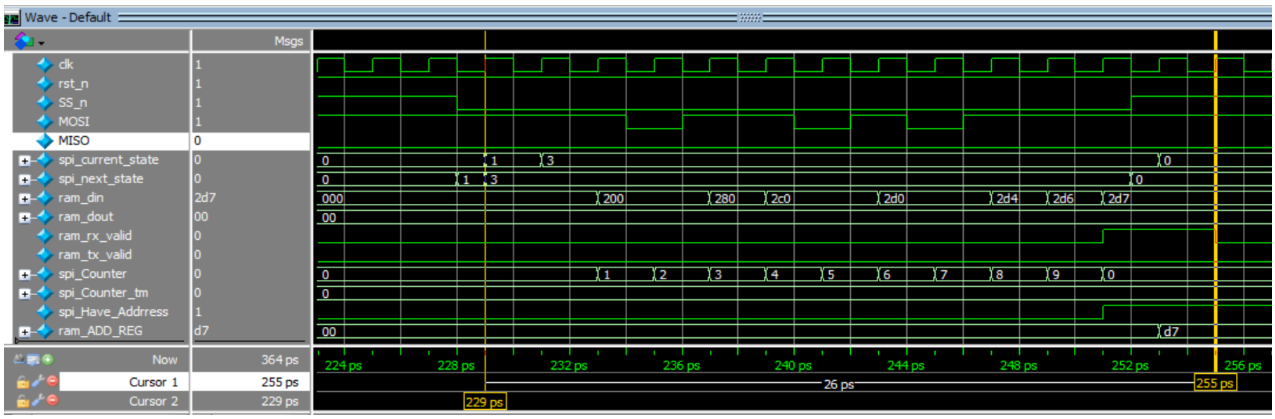
5- Read Address Case:

We give input address as 11010111(d7 in hex) to read previous data we write on miso and verify its write and read data successfully...

```
//228ns
    // Start SPI transaction
    SS_n = 0;@(negedge clk);

    // Send command (READ ADDRESS)
    MOSI = 1;@(negedge clk);
    MOSI = 1;@(negedge clk);
    MOSI = 0;@(negedge clk);
    //ADDRESS
    MOSI = 1;@(negedge clk);
    MOSI = 1;@(negedge clk);
    MOSI = 0;@(negedge clk);
    MOSI = 1;@(negedge clk);
    MOSI = 0;@(negedge clk);
    MOSI = 1;@(negedge clk);
    MOSI = 1;@(negedge clk);
    MOSI = 1;@(negedge clk);

    // End SPI transaction
    SS_n = 1;
    repeat(15) @(negedge clk);
```



Note: Start SPI @229ns – end SPI & return ideal case @255ns

Number of clocks = (255-229)/2 = 13 clock cycle ✓

6- Read Data Case:

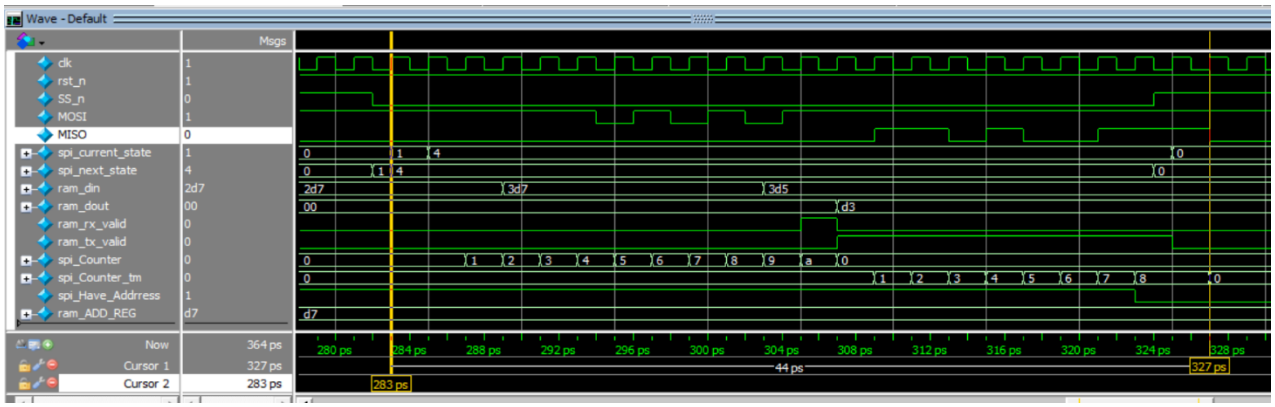
We here interested on command input.(ignore rest of data)

```
//282ns
    // Start SPI transaction
    SS_n = 0;@(negedge clk);

    // Send command (READ DATA)
    MOSI = 1;@(negedge clk);
    MOSI = 1;@(negedge clk);
    MOSI = 1;@(negedge clk);

    //ignored following data just for protocol
    MOSI = 1;@(negedge clk);
    MOSI = 1;@(negedge clk);
    MOSI = 0;@(negedge clk);
    MOSI = 1;@(negedge clk);
    MOSI = 0;@(negedge clk);
    MOSI = 1;@(negedge clk);
    MOSI = 0;@(negedge clk);
    MOSI = 1;@(negedge clk);

    repeat(9) @(negedge clk);
    SS_n=1;
    repeat(15) @(negedge clk);
```



Note: Start SPI @283ns – end SPI & return ideal case @327ns

Number of clocks = (327-283)/2 = 22 clock cycle ✓

-MISO return 11010011(d3 in hex) which we write it in write data case... ✓