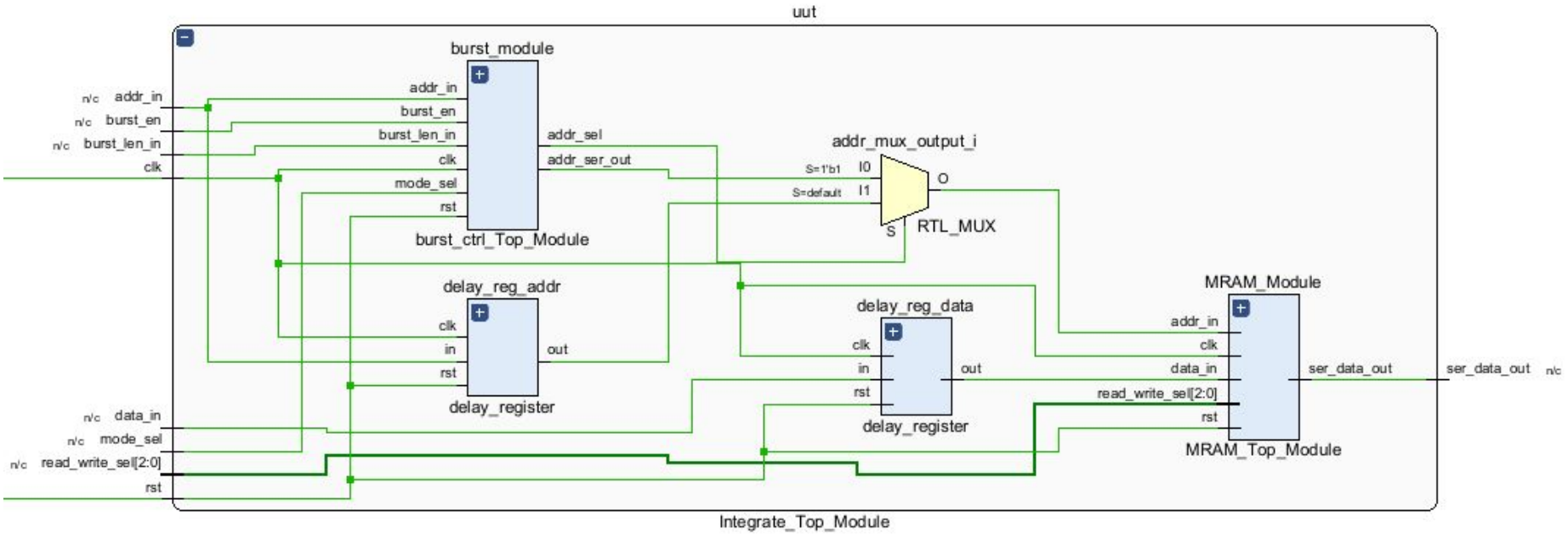
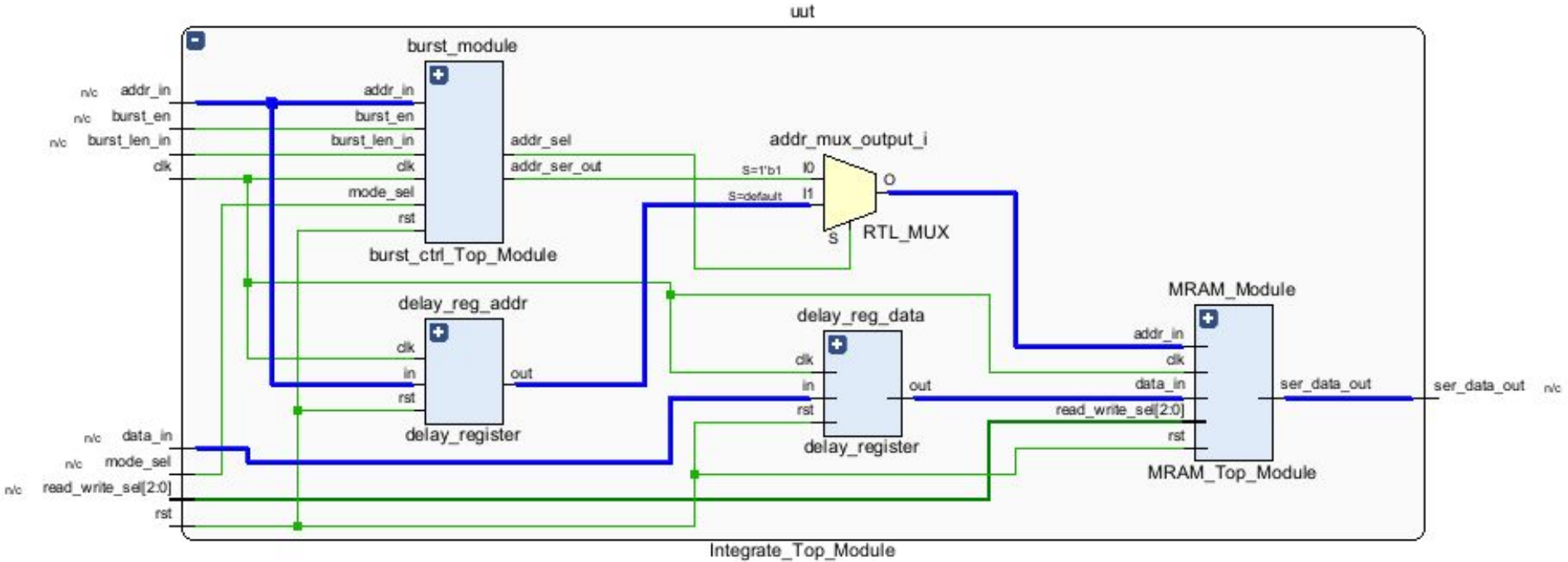
Timing analysis

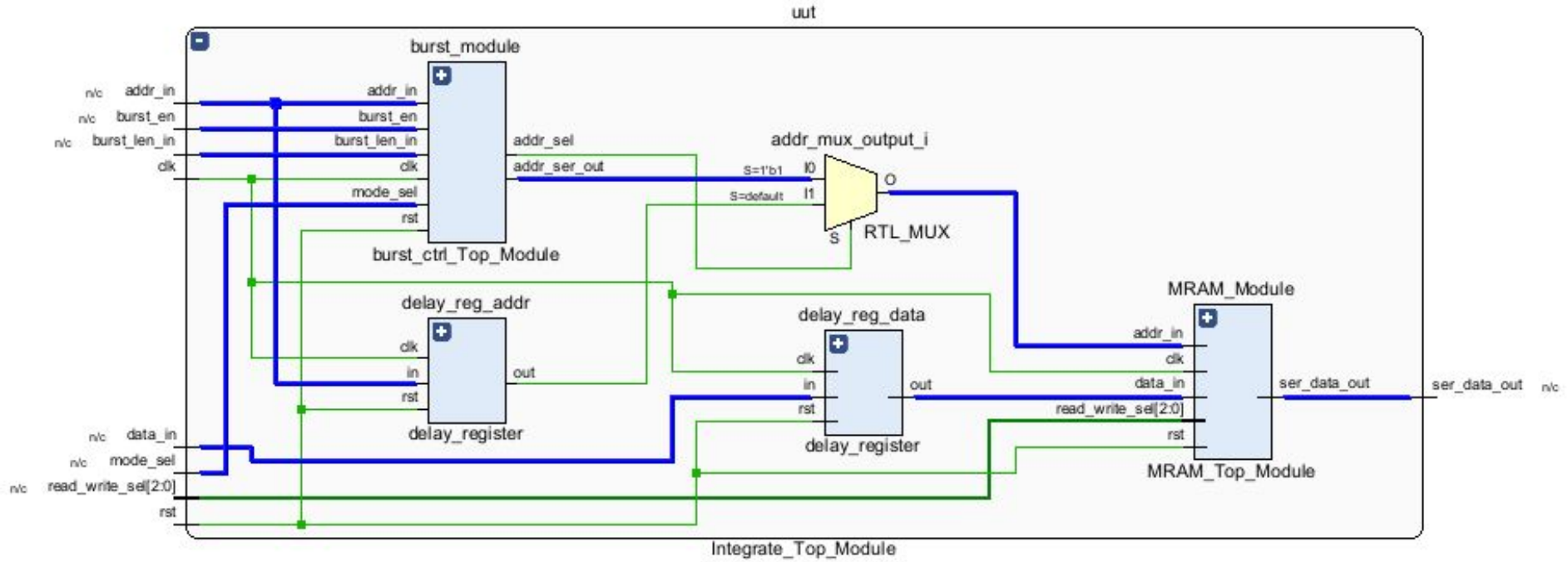
Integrated Overview of Controller



During Single Transfer / First loop of Burst Transfer



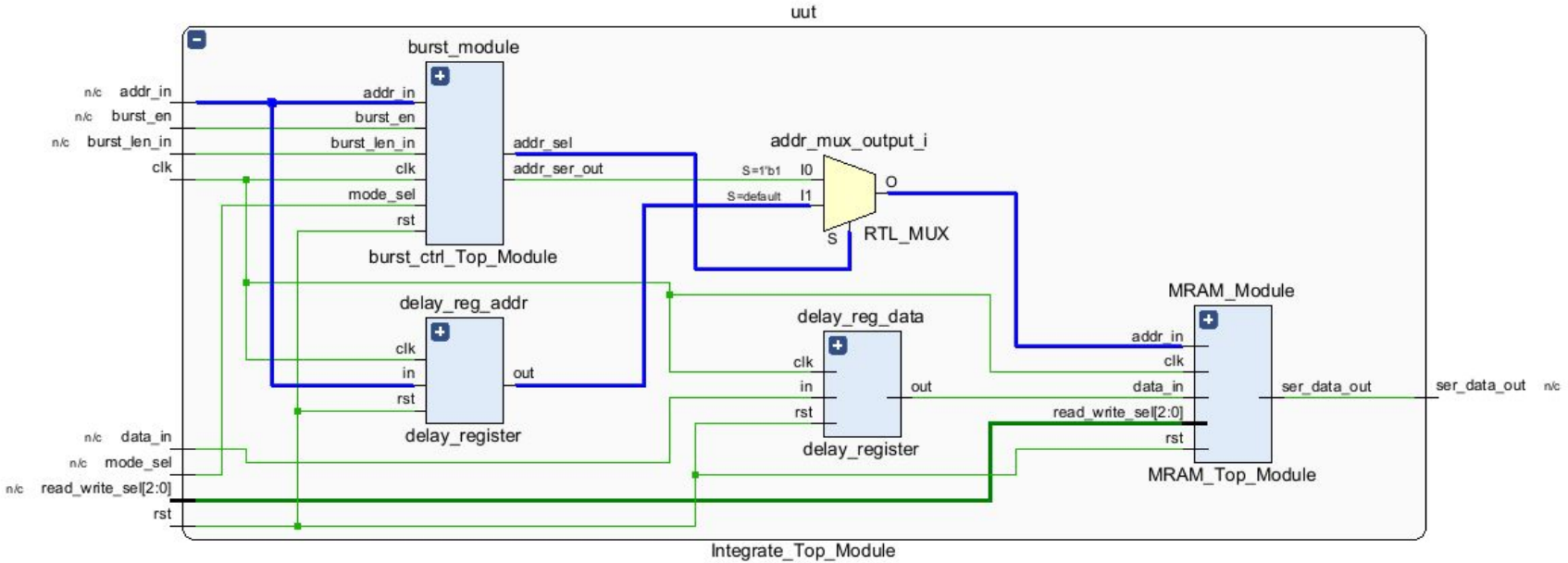
During Burst Transfer



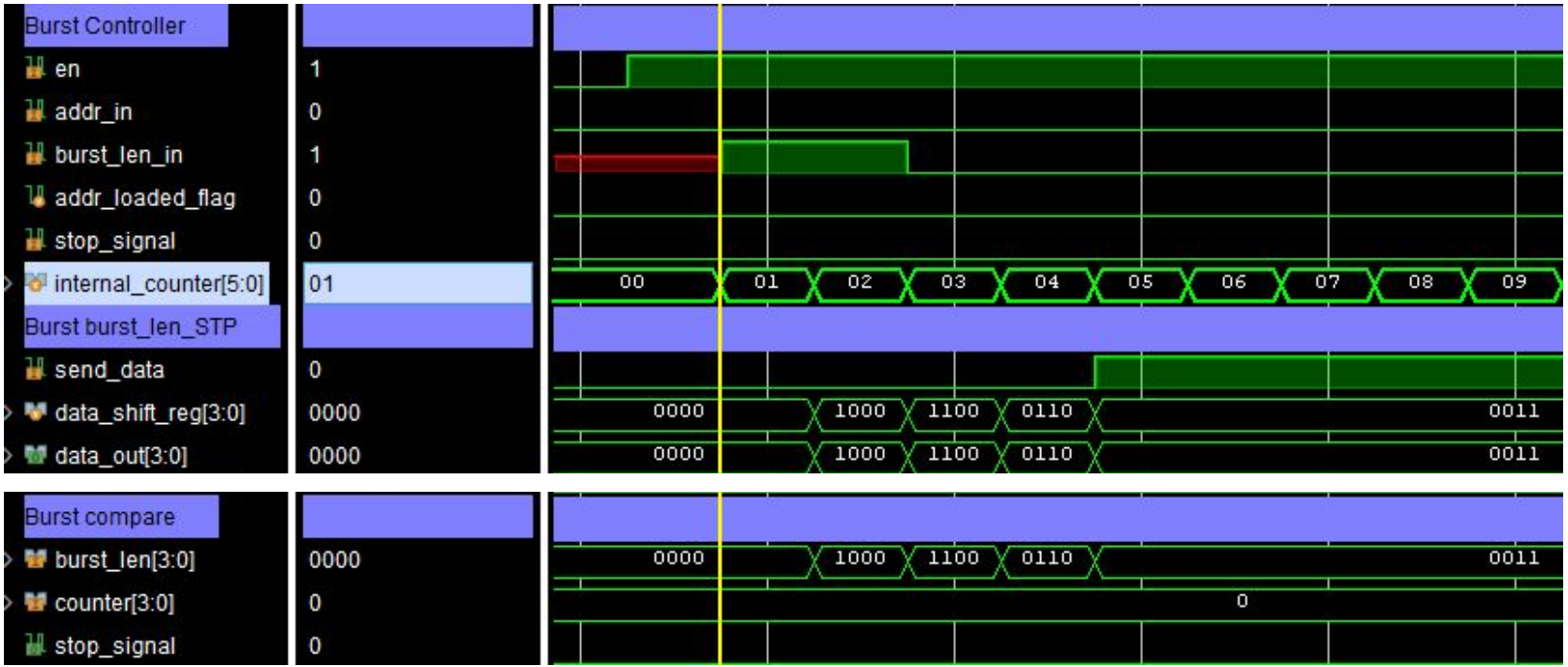
Burst Module

If Single Transfer is selected, don’t do anything.

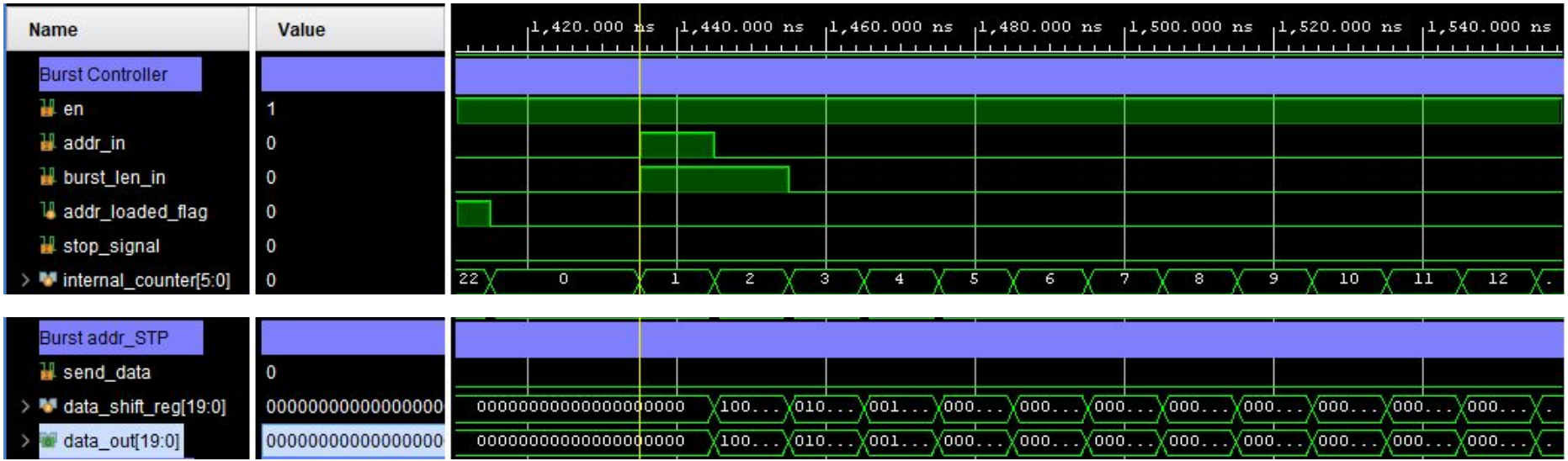
Done through setting the mux to 0.



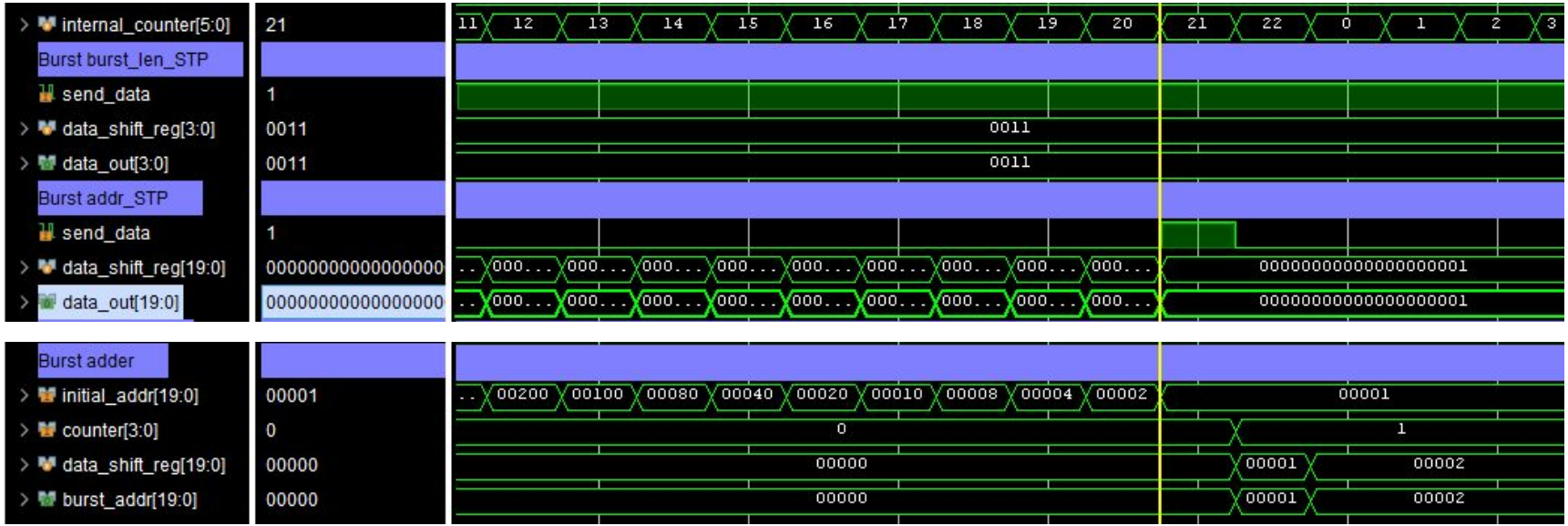
Burst Module - Burst len



Burst Module - Initial addr



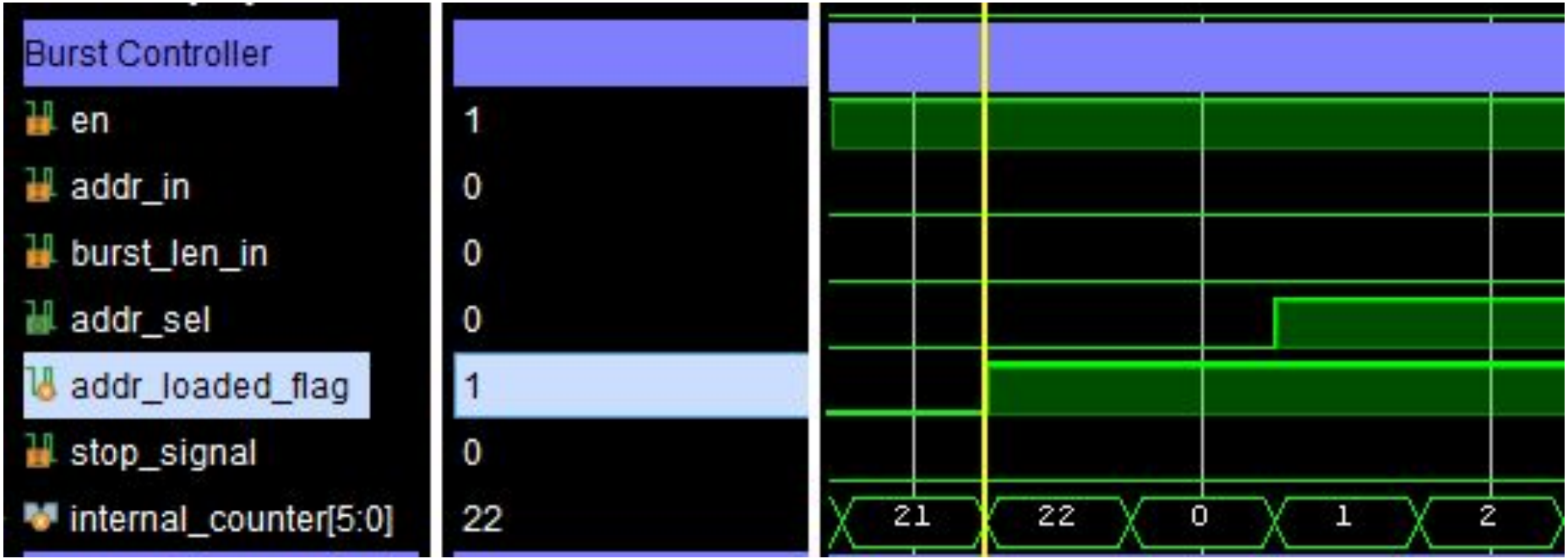
Burst Module - Initial addr



Burst Module - Initial addr

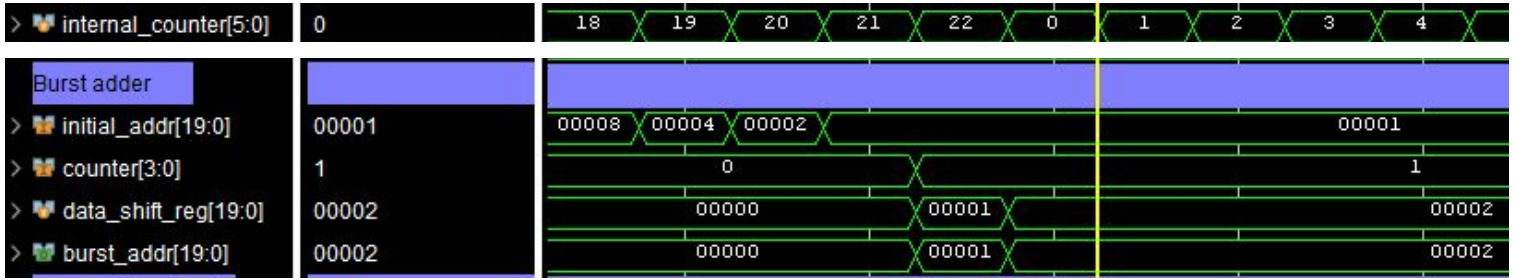
After initial addr is loaded, set the loaded flag and addr_sel(mux control)

Addr read for the STP/PTS module starts when counter = 2



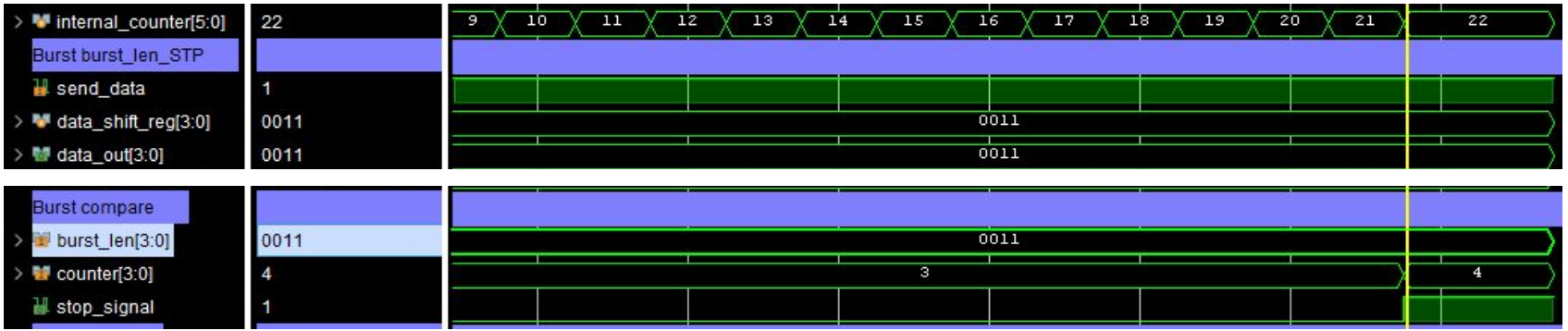
Burst Module - Adder

Takes initial addr and counter module’s value and adds them tgt

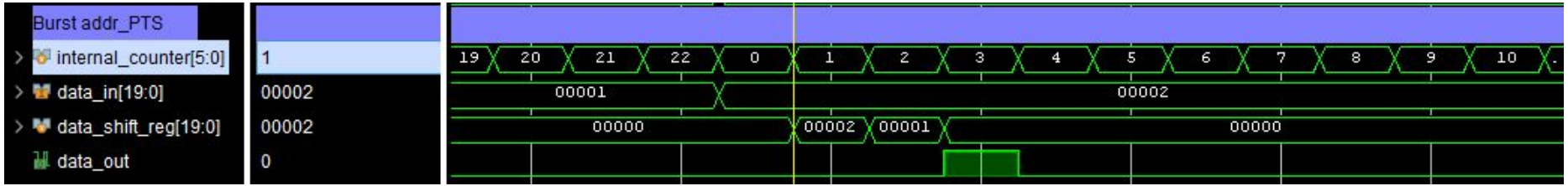


Burst Module - compare module

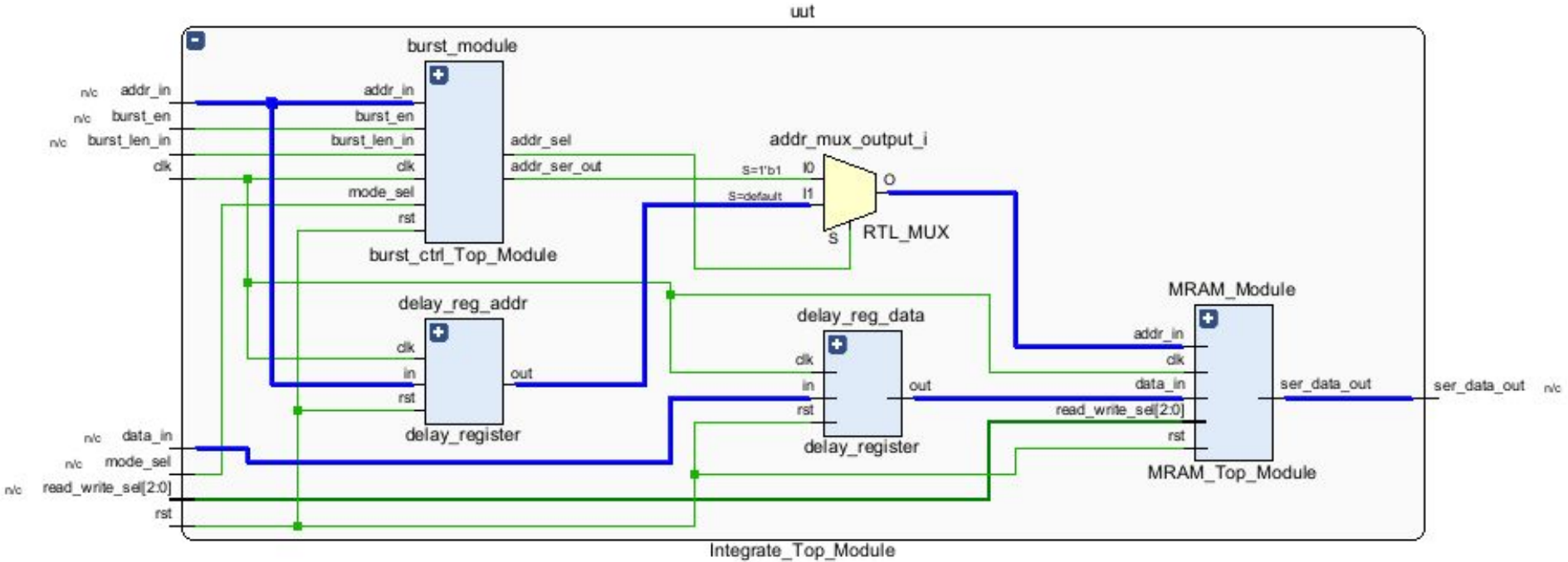
Cycle 22: When compare module detects that the counter value exceeds the burst len, assert stop signal to the burst control to stop sending address serially



Burst Module - address PTS module

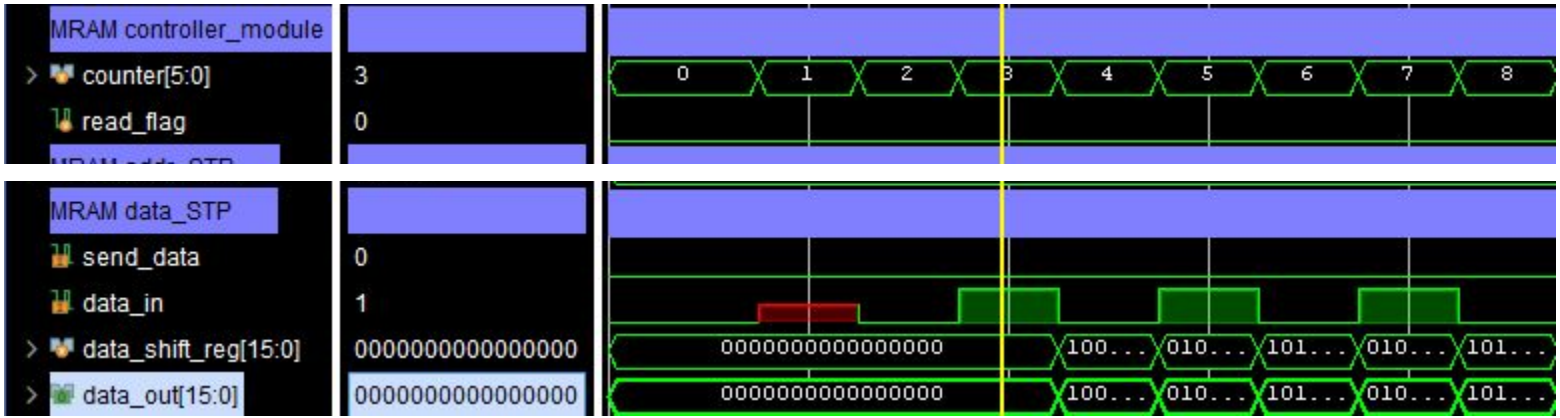


MRAM STP/PTS module -



MRAM STP/PTS module - data/addr STP module (write op)

At counter = 2, data gets read and takes 1 cycle to be shifted into the internal shift register

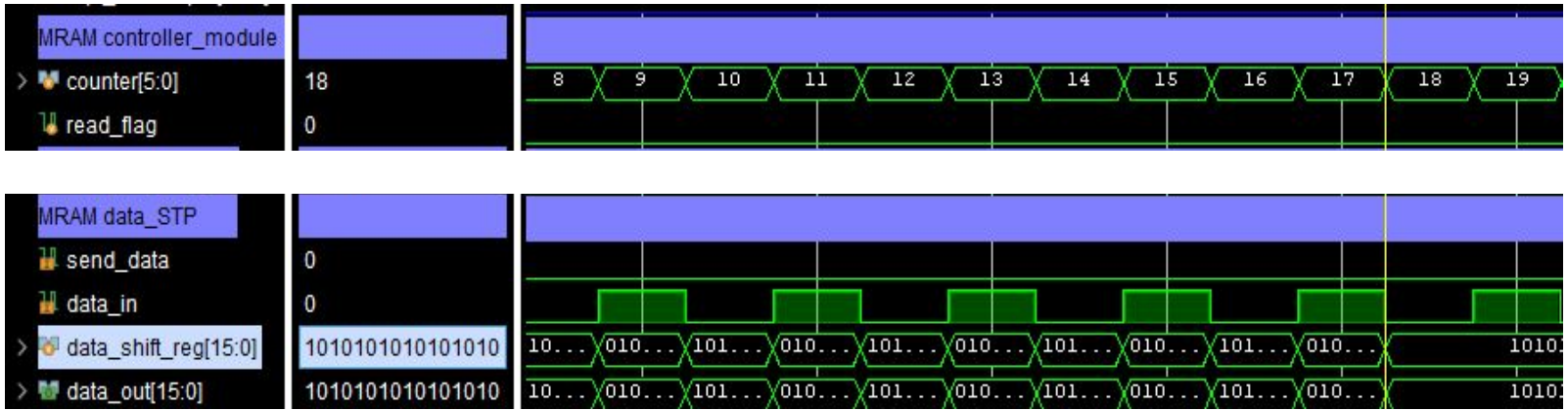


MRAM STP/PTS module - data/addr STP module (write op)

At counter = 18, data is fully in the internal shift register.

At counter = 22, addr is fully in the internal shift register.

Can be sent to the MRAM alongside the control signals



MRAM STP/PTS module - addr STP module (write op)

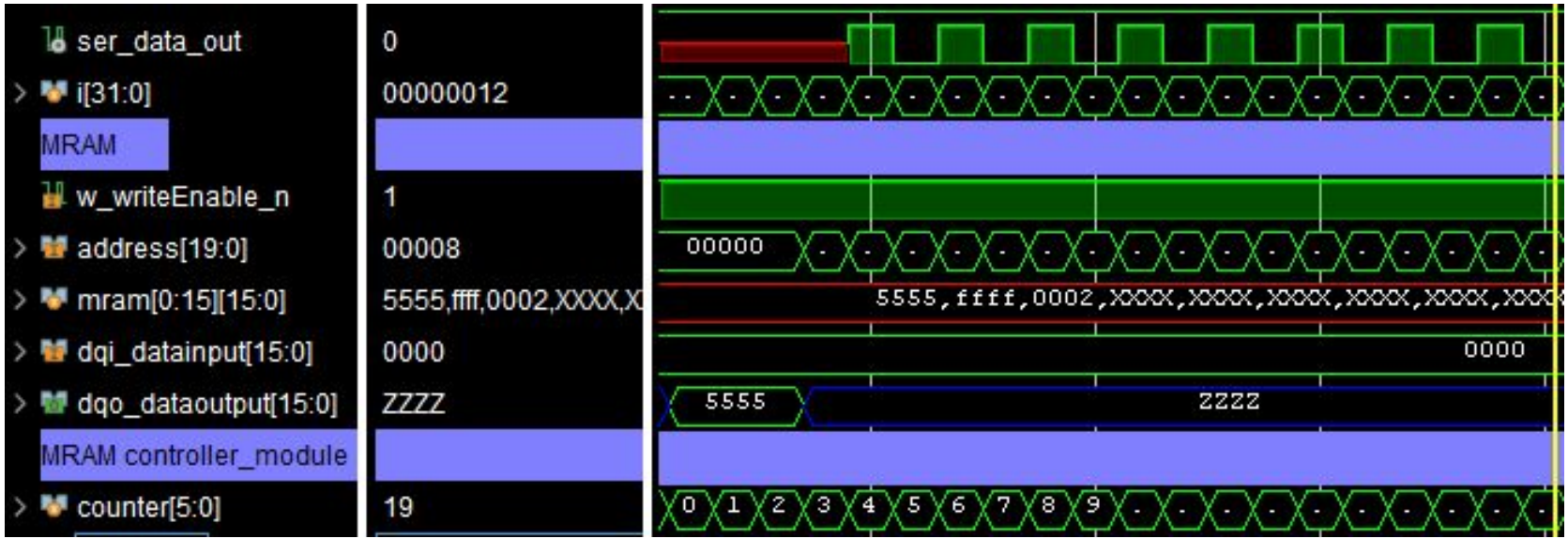
At counter = 22, signals are asserted to write to the MRAM

At counter = 0 of the next cycle, it is in the MRAM

MRAM STP/PTS module - Read operation

Data is read out in the next loop after address is provided

Addr = 0; data in address 0 is 5555h, 0101_0101_0101_0101b. Data is output from LSB to MSB



Testbench - burst read example

```
198     /*-----------------------------------------------
199       // BURST READ
200       @(posedge clk)
201       rst <= 0;
202
203       burst_en <= 1;
204       mode_sel <= 1;
205       read_write_sel <= 3'b110;
206
207       @(posedge clk)
208       burst_len_in <= 1;
209       addr_in <= 0;
210
211       @(posedge clk)
212       burst_len_in <= 1;
213       addr_in <= 0;
214
215       for (i = 0; i < 18; i = i+1) begin
216           @(posedge clk)
217           addr_in <= 0;
218           burst_len_in <= 0;
219       end
220
221       @(posedge clk)
222       @(posedge clk)
223       @(posedge clk)
224
225
226       // Stall to see output
227       @(posedge clk)
228       for (i = 0; i < 69; i = i+1) begin
229           @(posedge clk)
230           addr_in <= 0;
231       end
```

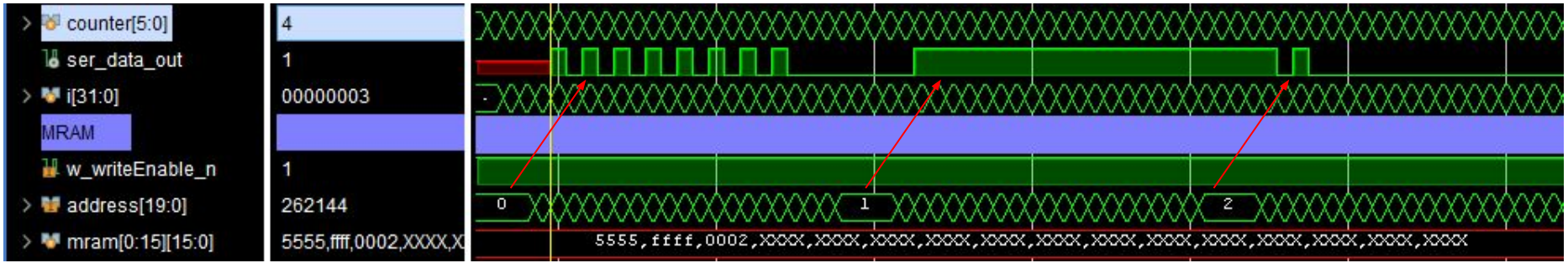
Control signals

Burst len = 3

Addr = 0

Stall to see output

Testbench - burst read example



Integrated module summary

1 loop consists of 23 cycles (0-22)

Writing:

- Takes 23 cycles

Reading:

- Takes 23 cycles to provide address
- Data is output on the next loop, cycle 4 to 19