Write controller files:



2.



3.



4.



Architecture:

Simulations:

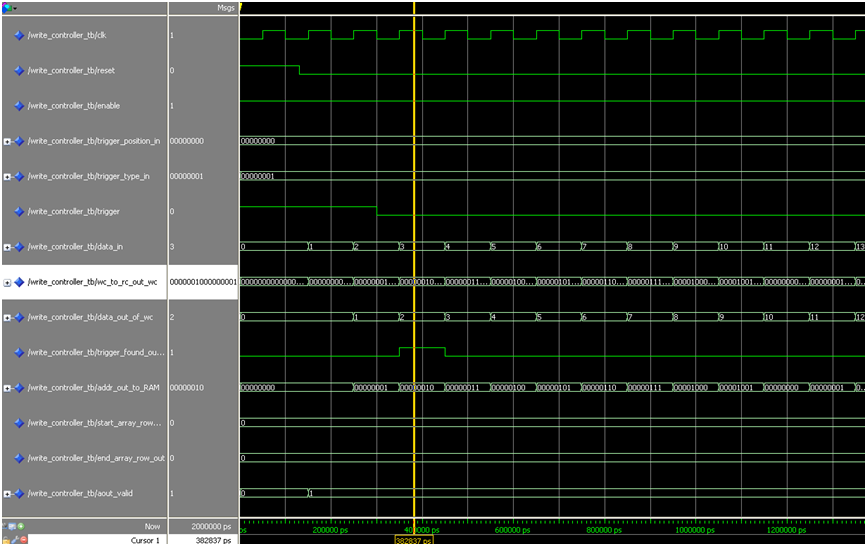
(Every simulation is from 0-1000 ns, clk duty cycle is 50 ns, Reset is on 0-130 and off 130-1000 ns)

Check list:

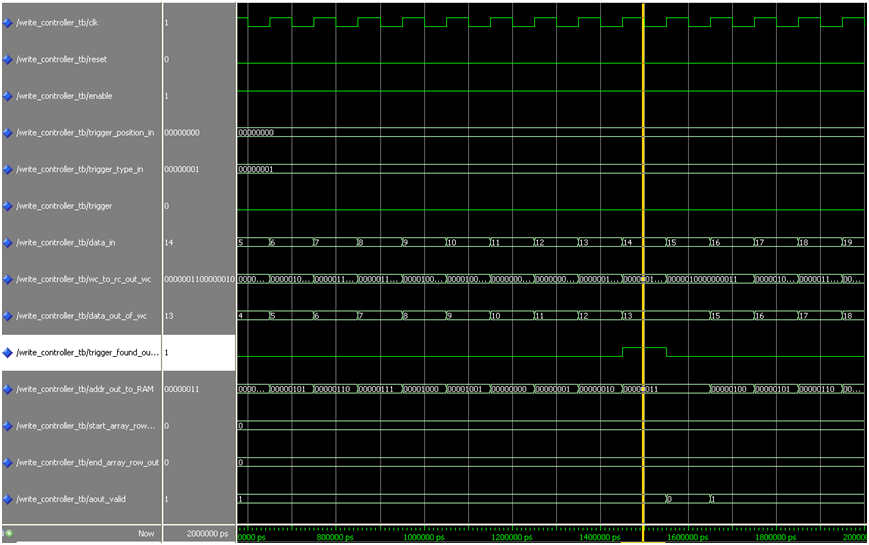
1. Trigger type 1, trigger pos 0 – trigger high 1-300, trigger low 300-2000

We expect to see trigger found='1' at 300 ns. wc\_to\_rc="0000000000001001".

The system will ignore triggers after the first one,until # record\_depth\_g of clk cycles will pass.

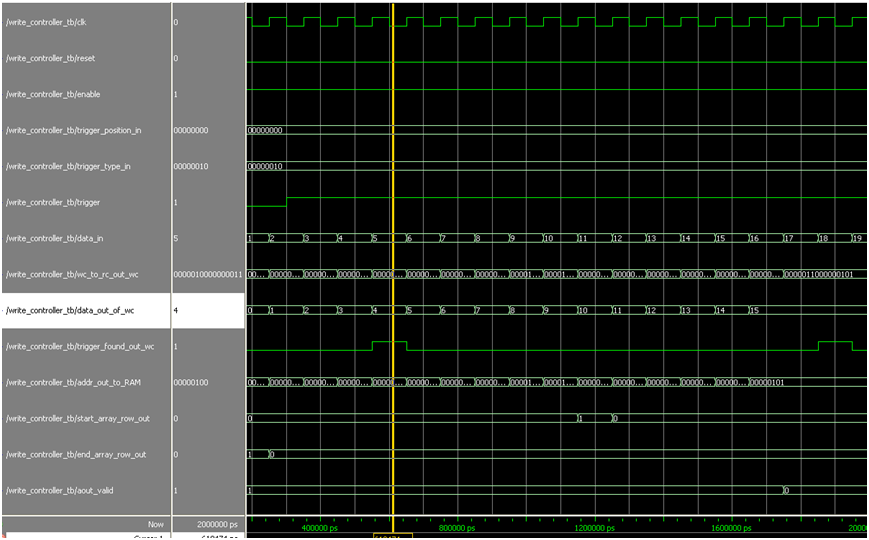


We can see that according the defened configuration above, the trigger is identify with the first clock rise, in addition we can see that the address that is been send to the RC as start address is the current address (00000010), and the end address is one before that (00000001). [wc\_to\_rc\_out is combination of those two addresses].



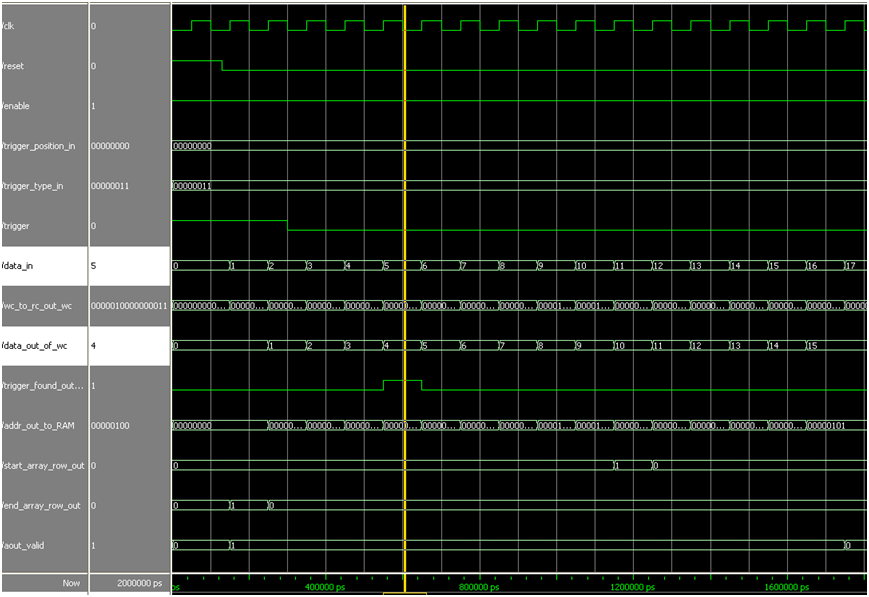
The trigger is been identify second time in the 13 iteration, after 10 iterations have passed since the last trigger has been found. [correction to the middle presentation]

1. Trigger type 2, trigger pos 0 – trigger low 1-300, trigger high 300-2000.



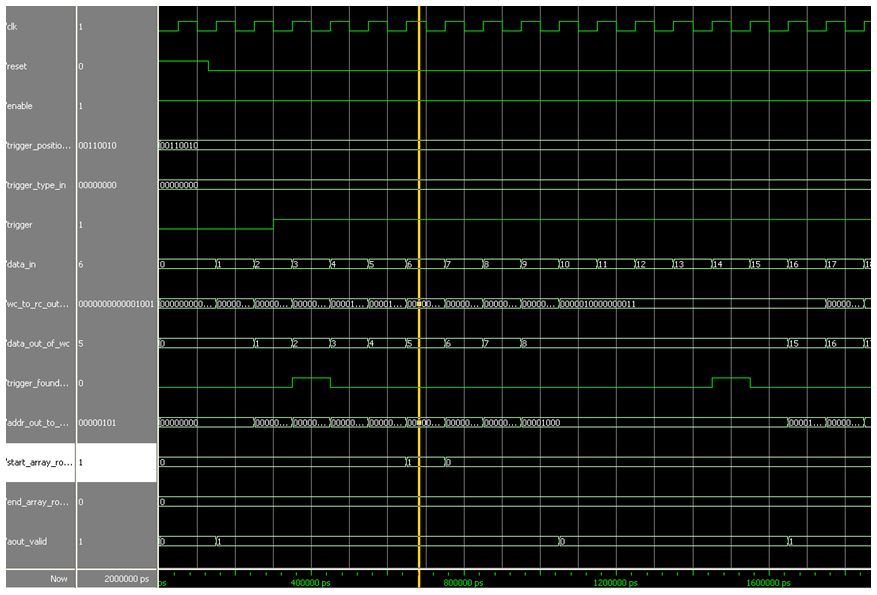
We can see the trigger rise with the correct address as we expect. There is also a match between the start and end addresses of the recorded data and the trigger address that we found.

1. Trigger type 3, trigger pos 0 – trigger high 1-300, trigger low 300-1000



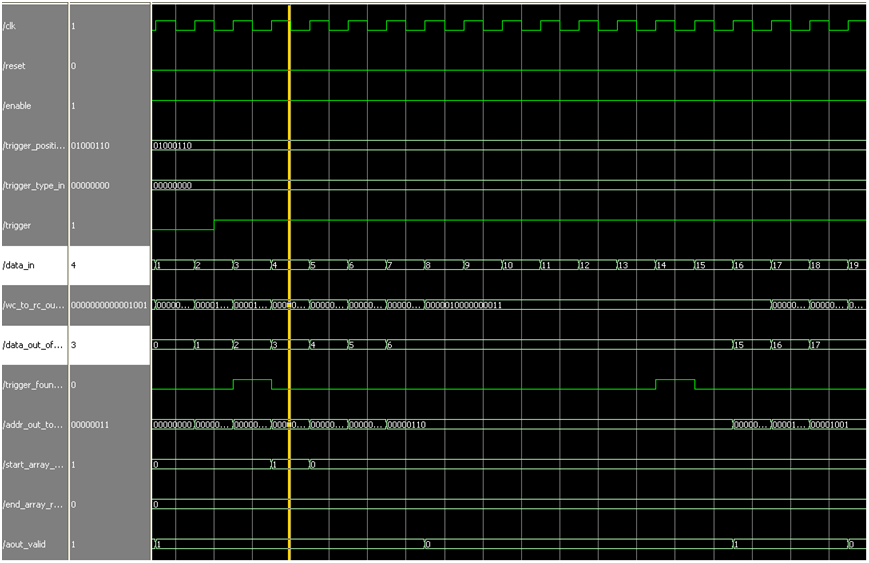
Again, we can see the trigger rise with the correct address according the definition.

1. Trigger type 0, trigger pos 50 – trigger low 1-300, trigger high 300-1000



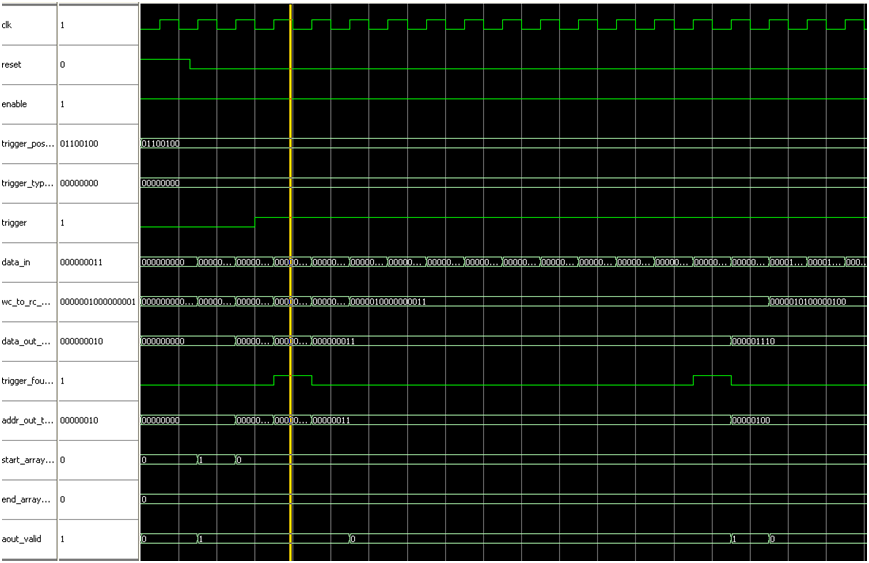
Now trigger\_position = 50% and we can see the exact address according an offset of 5 places as we expected. We can also see that 5 clock cycles after trigger rise, the aout\_valid signal is turn into 0 in order to give the RC part enough time to output the data from the RAM.

1. Trigger type 0, trigger pos 70 – trigger low 1-300, trigger high 300-1000



We can see the change in the address according trigger position, address 1 is changed to 7, and in address 3 we are in the next RAM (address 10). We can also see that after trigger found, we wait 3 clock cycles and turn aout\_valid to 0, in order to give the RC time to output the data.

1. Trigger type 0, trigger pos 100 – trigger low 1-300, trigger high 300-1000



In the last test we can see that the address has not changed (change in 100% will bring us back to the same address). Not like in 0% case, aout\_valid is turned into 0 for 10 cycles in order to give the RC time to read from the RAM.