Description:

The read controller gets the start address of the valid data that was calculated in the write controller and extract the correct data from the RAM and send it out.

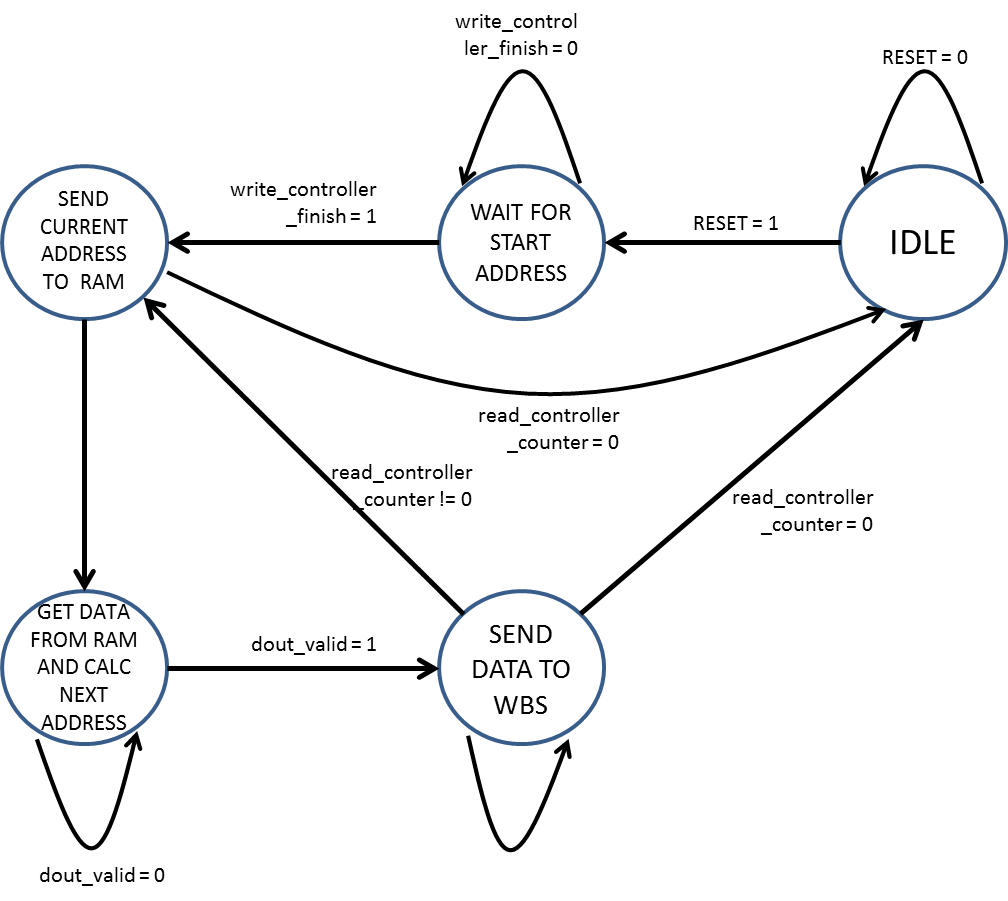
Generic table

|  |  |  |
| --- | --- | --- |
| Name | Width | Description |
| reset\_polarity\_g | 1 | '1' reset active high, '0' active low |
| record\_depth\_g | 4 | number of bits that are recorded from each signal is 2^record\_depth\_g |
| data\_width\_g | 8 | defines the width of the data lines of the system |
| num\_of\_signals\_g | 8 | number of signals that will be recorded simultaneously |

Signals table

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Direction | Width | Description |
| clk | In | 1 | System clock |
| Reset | In | 1 | System reset |
| start\_addr\_in | In | record\_depth\_g | The start address of the data that we need to send out to the user |
| write\_controller\_finish | In | 1 | '1' ->WC has finish working and saving all the relevant data |
| dout\_valid | In | 1 | Data coming from RAM validity |
| data\_from\_ram | In | num\_of\_signals\_g | Data coming from RAM |
| data\_in | In | num\_of\_signals\_g | Data in from signal generator |
| read\_controller\_finish | Out | 1 | 1 -> RC finish sending all the data out. 0 -> other |
| addr\_out | Out | record\_depth\_g | Address sent to the RAM in order to extract it out back to the user |
| aout\_valid | Out | 1 | Validity of address that sent to the RAM |
| data\_out\_to\_WBM | Out | num\_of\_signals\_g | Data sent out back to the user |
| data\_out\_to\_WBM\_valid | Out | Add\_width\_g | Validity of the outputting data |

Read controller state machine



Output table

