TriplePort WISHBONE SPRAM Wrapper

Contents

1- Architecture

This is a wrapper for an inferred single port RAM, that converts it into a Three-port RAM with one WISHBONE slave interface for each port. Very useful as a drop-in module to create configuration registers for any core.

In case there is no need to make atomic operations, a master connected to a port of this core can work as if there weren't any other masters connected to other ports. Plus, the core switches the port priority the moment a master (that was using the memory) drives low its cyc and stb lines.

That is, for example: master B does an operation (R/W), and it is acked by this core. Then, if master B drives low its cyc and stb lines, and in the next cycle master B and master C rise their cyc and stb lines simultaneously, master C will be the one serviced, not master B again. This way a greedy master won't take up the bus.

ATOMIC OPERATIONS (port locking):

Let's suppose that master B wants to make a set of atomic operations consisting of one read, then a pause of 20 cycles, then a write, then another pause of 5 cycles and then another write. This master (B) needs to know that no other master connected to this core (A or C) does any read or write to the memory, while this set of atomic operations is being performed.

In order to make atomic operations, master B (which is connected to, for example, the 2nd WB port) would need to drive high the wb2_cyc and wb2_stb lines, to perform the first read of the set of atomic operations. After receiving the ack from this core, master B will drive low ONLY the line wb2_stb during the 20 pause cycles master B to prepare the next write it needs to do.

Then, master B should drive high again the stb line and perform the write (of course rising also the "we_i" line and putting the right data on "dat_i" and "adr_i" lines), remember that the cyc line was already up, to lock the memory on to this port.

After receiving the ack for this write, the master B of this example needs another 5 cycles of processing in which it won't make any other operations on the RAM, so after the second ack (the one corresponding to the second WB operation, the write) it will drive low ONLY the wb2_stb line, keeping cyc high, that way the memory bus is locked on to this port, and it won't service requests, again, from other ports.

After the five cycles, master B makes its last operation of the example, another write, by driving high the stb line (and with the rigth we_i, dat_i and adr_i lines, obviously) and waiting to be acked by this core, meaning that the memory was written. After receiving this third, and last, ack master B will drive low its wb2_cyc and wb2_stb lines, because it has ended the atomic set of operations, so that master A and C can keep on using the memory.

I short: Keeping the wbN_cyc line high with the wbN_stb line low (after being acked once) will signal to this core that the memory must be kept locked on to master N and no other request from other ports will be serviced until the cyc line is lowered again.

A WISHBONE slave port N may be considered locked onto its master X when:

1) this slave port N acks one time

2) the wbN_cyc line is not driven low by master X.

From this point on, the acked master may make any operations on this slave port, knowing for sure that no other master will be able to access the memory until the cyc line is driven low again. (the "slave acks one time" requirement is because up until that moment, the master can not be sure that there is another master already locking this slave)

In simple terms, any group of atomic operations must be preceded by a read, and the cyc line must not be driven low until the end.
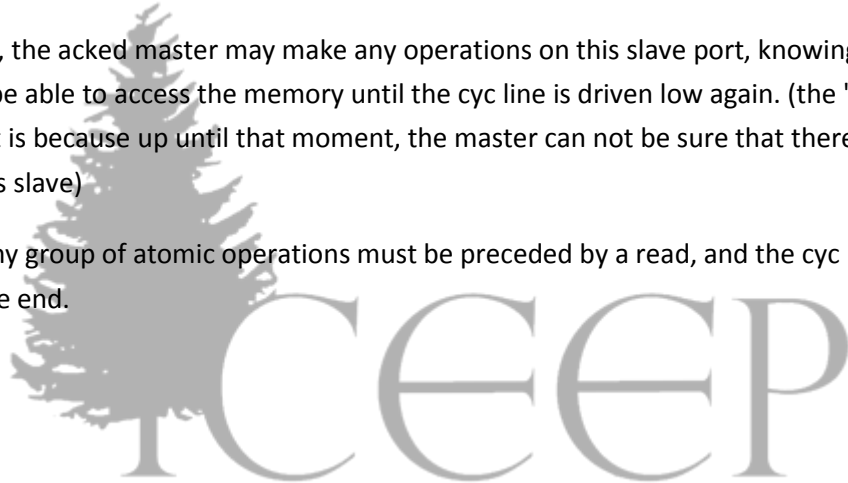
2- References

[1]   http://opencores.org/project,wb_3p_spram_wrapper.