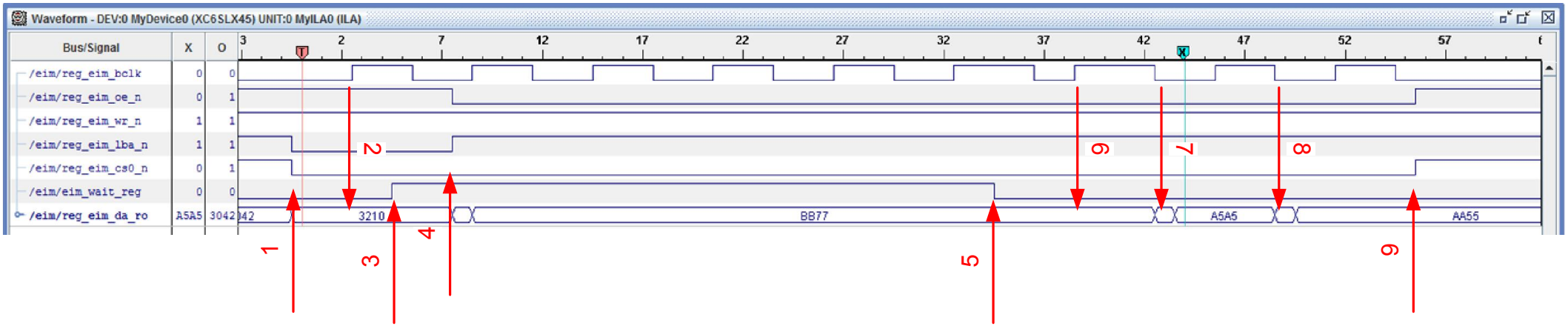

Here is what READ transaction looks like. Wait flag is shown using positive logic, it is inverted in actual hardware.

1. Chip Select is asserted, write flag is not asserted, LBA is asserted.

2. First databeat takes place, arbiter latches address as LBA is asserted and understands that read transaction is initiated, because write flag was not asserted.

3. Arbiter asserts wait flag, sends address to user-side logic and starts waiting for an acknowledge signal along with data value.

4. LBA flag is cleared, meaning that next databeat will contain data, not address. Output Enable is asserted, bus direction is reversed, arbiter starts driving it.

5. Arbiter receives acknowledge from user-side logic along with some value read from user-side logic.

6. Master samples wait flag and finds out that it is clear, arbiter will drive its data onto bus on next falling clock edge.

7. Second databeat takes place, arbiter places lower bits of data on the bus, master will sample them on next rising clock edge.

8. Third databeat takes place, arbiter places higher bits of data on the bus, master will sample them on next rising clock edge.

9. Chip Select is cleared, transaction finished.