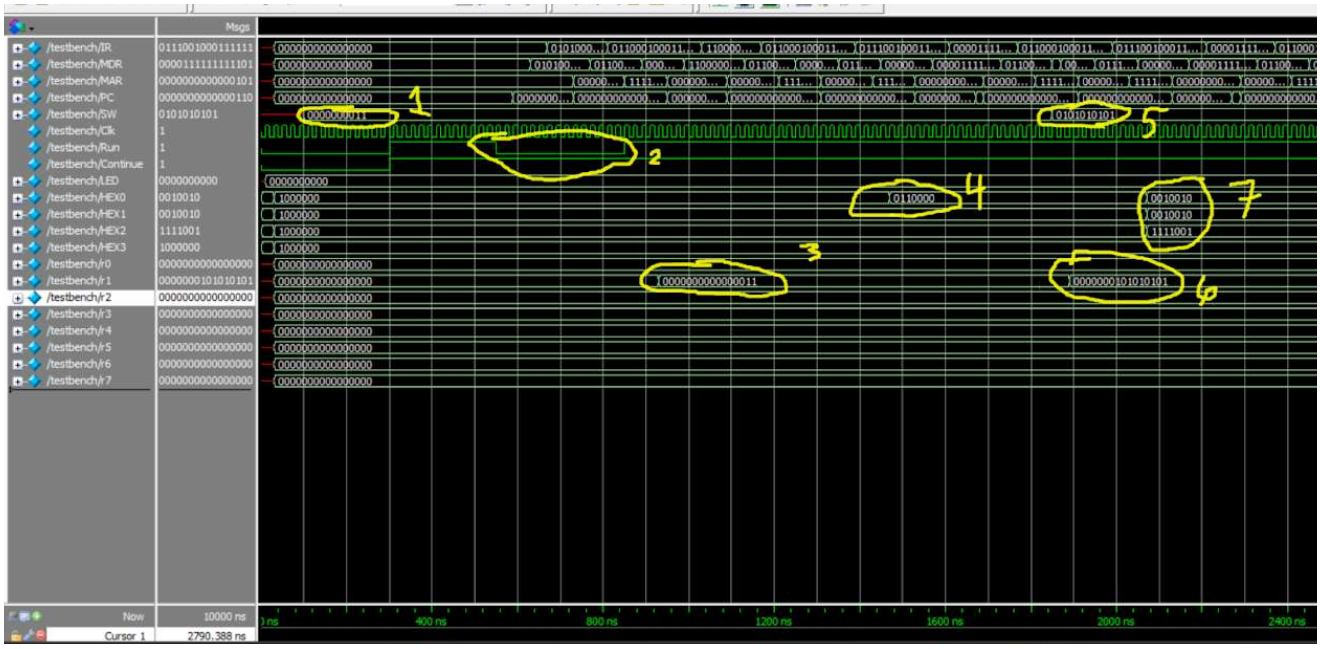
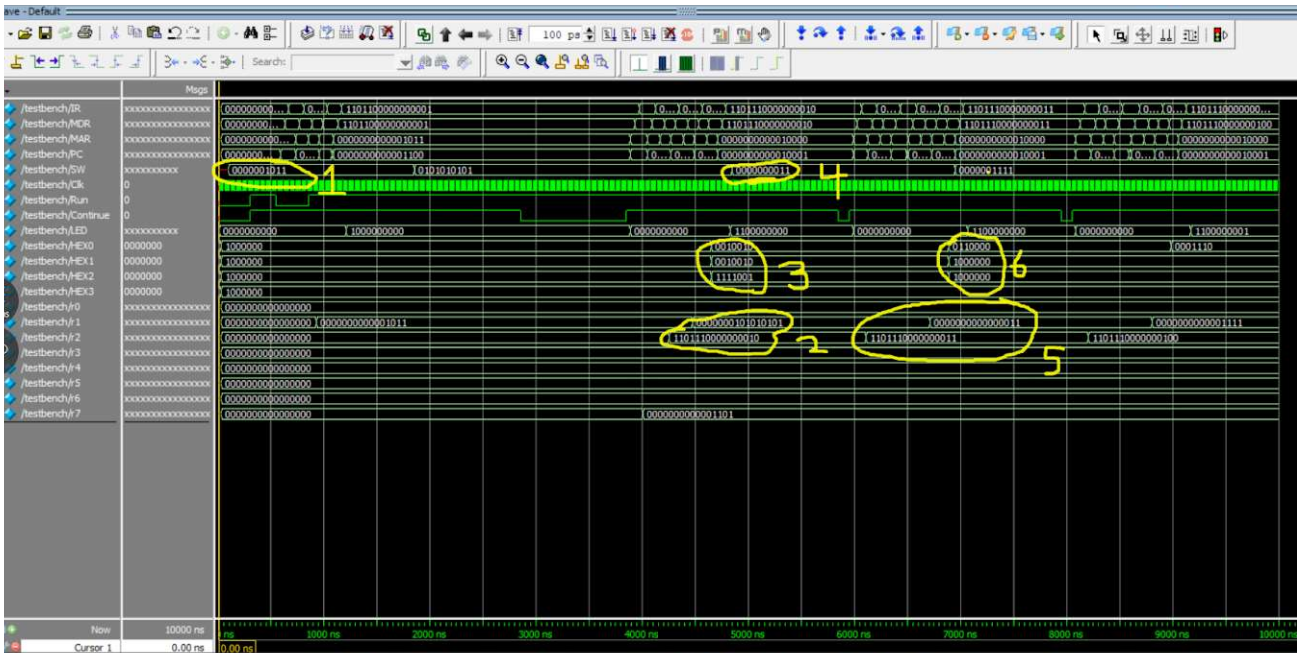


1. Switches set to 0x3 to load I/O Test 1
2. Run the program
3. The value of SW loads into R1
4. The value of SW is shown on HEX display
5. The value of SW is changed
6. The value of SW loads into R1
7. The value of SW is shown on HEX display

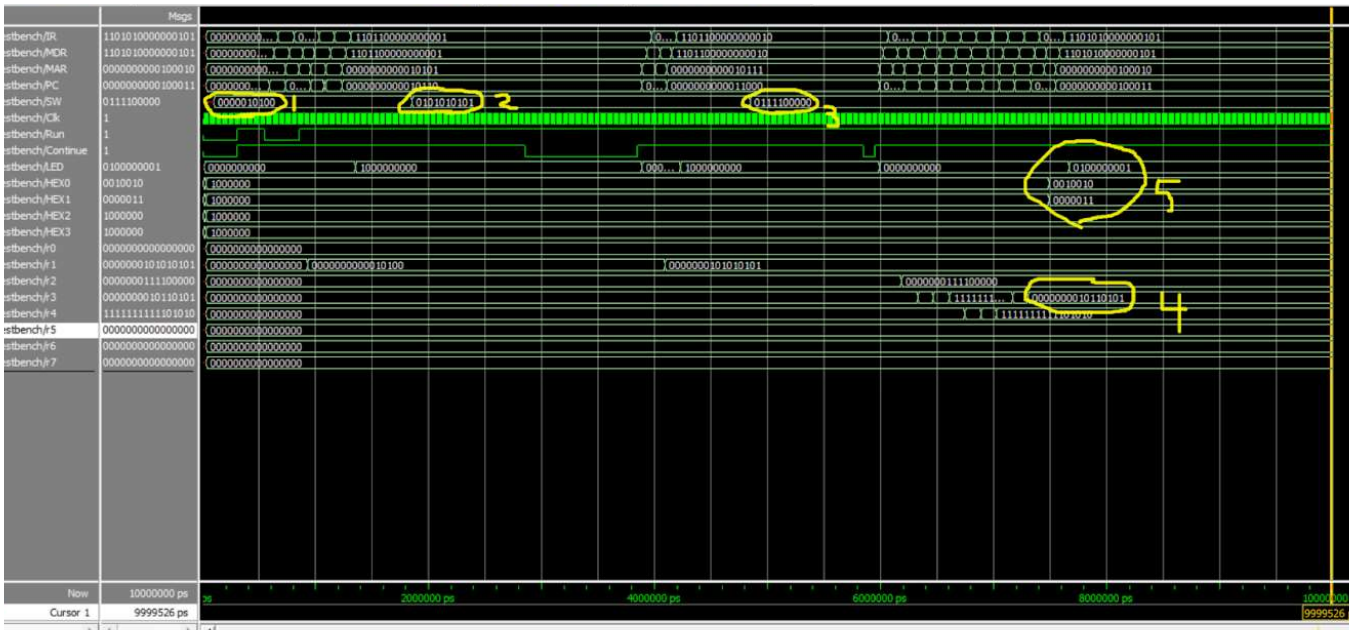
- I/O Test 2

1. Load 0x6 onto switches because it is the program address for I/O Test 2

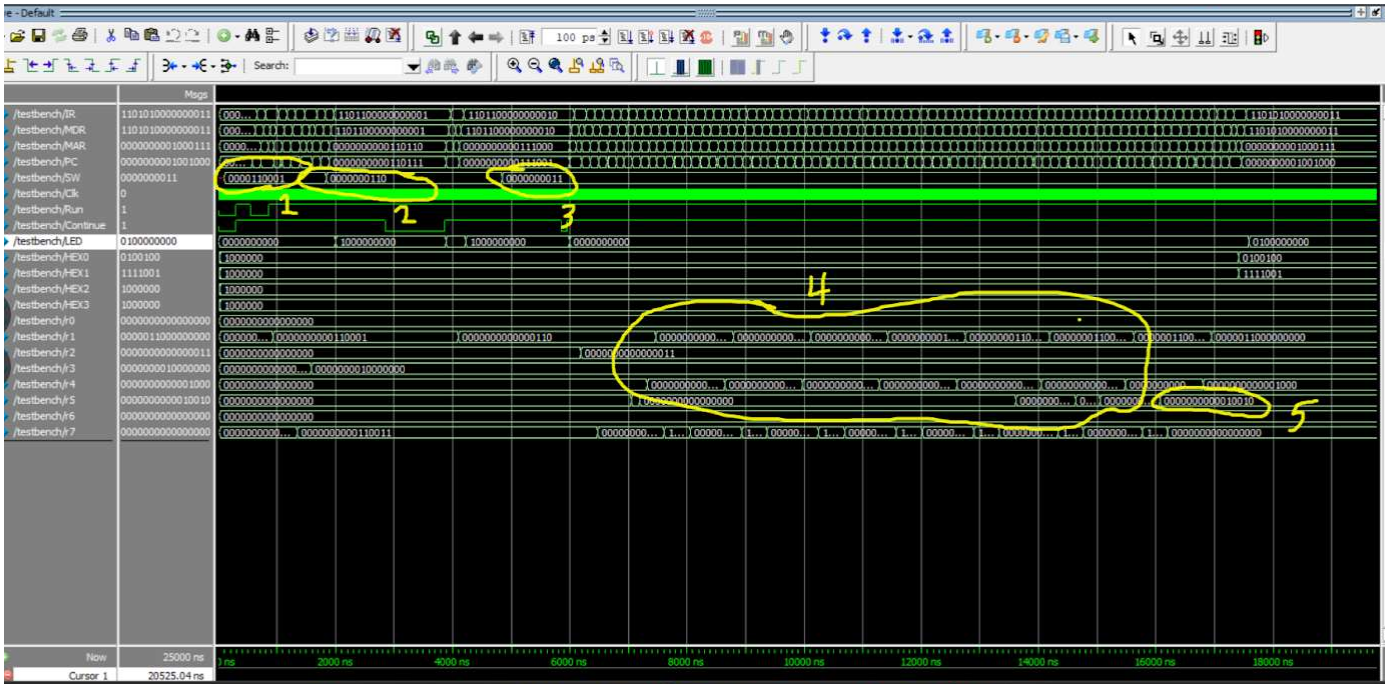
2. Put new value on switches
3. Press continue
4. Value is displayed on HEX display
5. Put switch value in R1
6. Switch value is changed
7. Nothing changes until continue is pressed
8. Hex display is updated
    - Self-Modifying Code



1. Load program start onto switches and run
2. The PC is put in R7 as a checkpoint. Switch value in R1, R2 stores pause instruction
3. HEX display updated
4. Switch value changed
5. R2 is incremented, R1 stores new switch value
6. HEX display updated
    - XOR

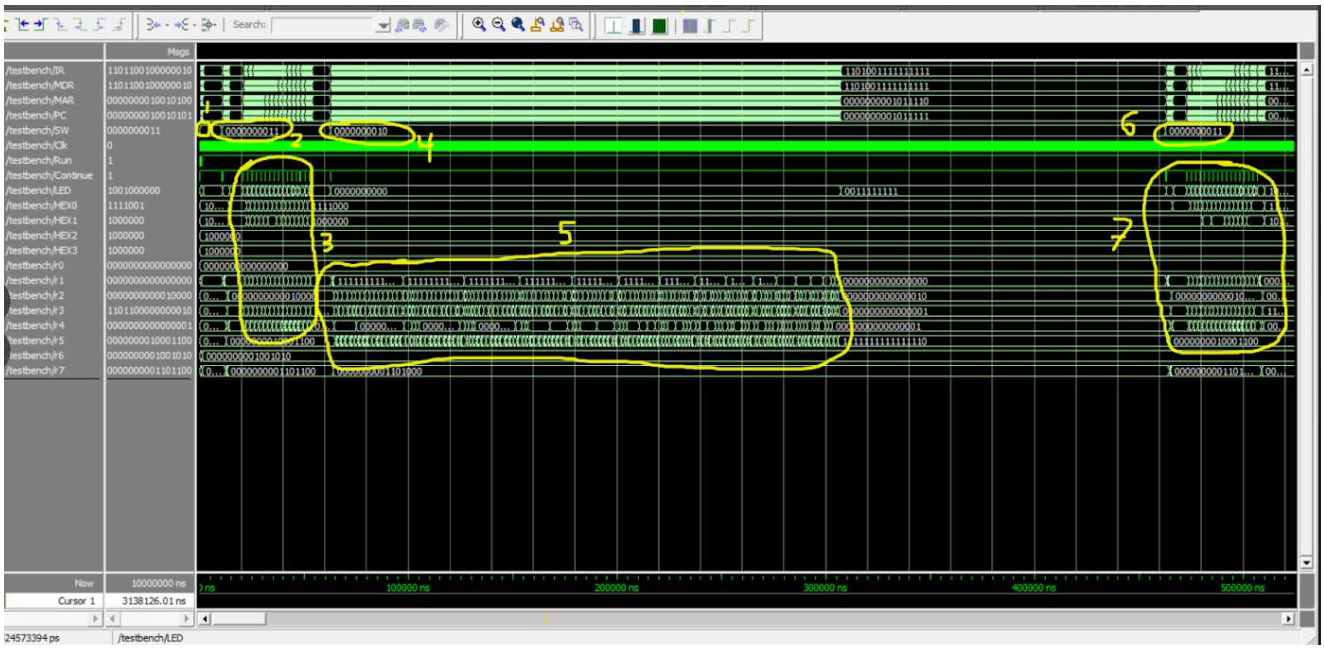


1. Load program start address on switches
2. Load A onto SW, it is stored in R1 after continue is pressed
3. Load B onto SW, it is stored in R2 after continue is pressed
4. A XOR B is stored in R3
5. R3 (A XOR B) is shown on HEX display
    - Multiplier



1. Address of multiplier program loaded onto switches
2. Multiplicand is loaded on switches (decimal 6)
3. Multiplier is loaded on switches (decimal 3)

4. Shift-add algorithm is used
5. Answer (decimal 18) is now in R5 and displayed on HEX display
    - Sort



1. Program start is loaded on switches (0x5A)
2. Display loop is selected from menu by placing 0x3 on switches
3. Continue is pressed 16 times to iterate over the stored values
4. Sort loop is selected from menu by placing 0x2 on switches
5. Sort loop is executed
6. Display loop is selected from menu by placing 0x3 on switches
7. Continue is pressed 16 times to iterate over the stored values which are now sorted least to greatest, shown below