**HW#05: AMBA Bus**

**Issued:** April 11 (Tuesday), 2023 **Due:** April 17 (Monday), 2023

**What to turn in**: **Copy the text from your MODIFIED codes and paste it into a document**. If a question asks you to plot or display something on the screen, also include the plot and screen output your code generates. Submit either a \*.doc or \*.pdf file.

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**Problem 1 (10p): ALU IP**

Implement an AHB slave interface of ALU IP. Please see the description in the lecture note for details.



Fig. 1-1: An AHB wrapper of RISC-V ALU.

What you have to do:

* Design an AHB slave interface of ALU IP based on the baseline code.
* Submit your RTL files (riscv\_alu\_if.v)
* Capture the waveform.

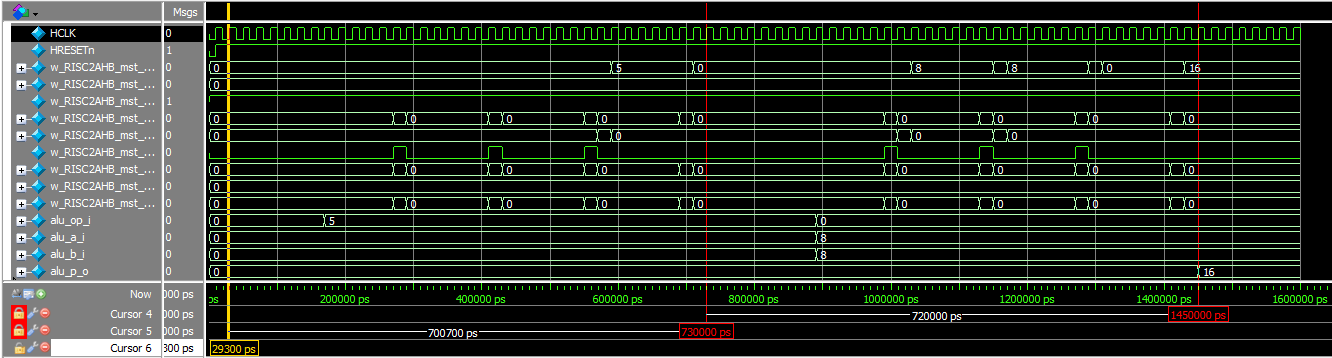
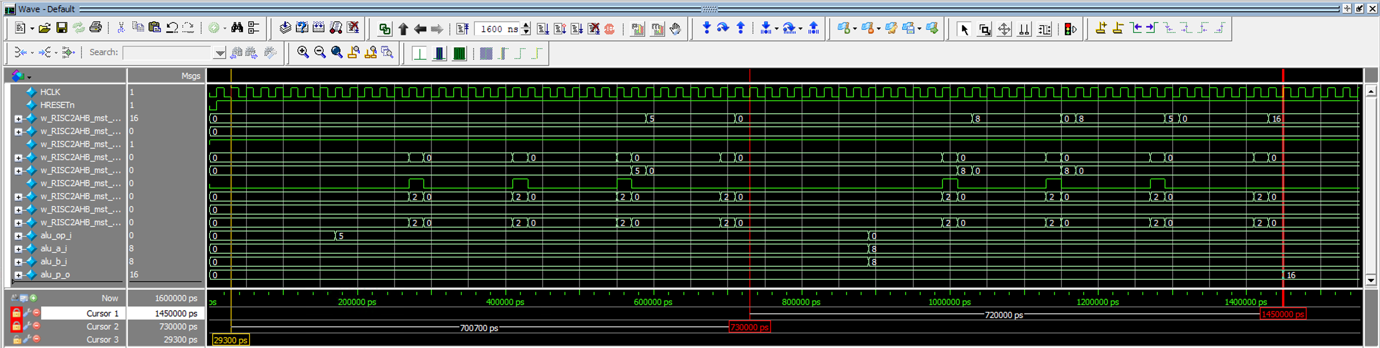
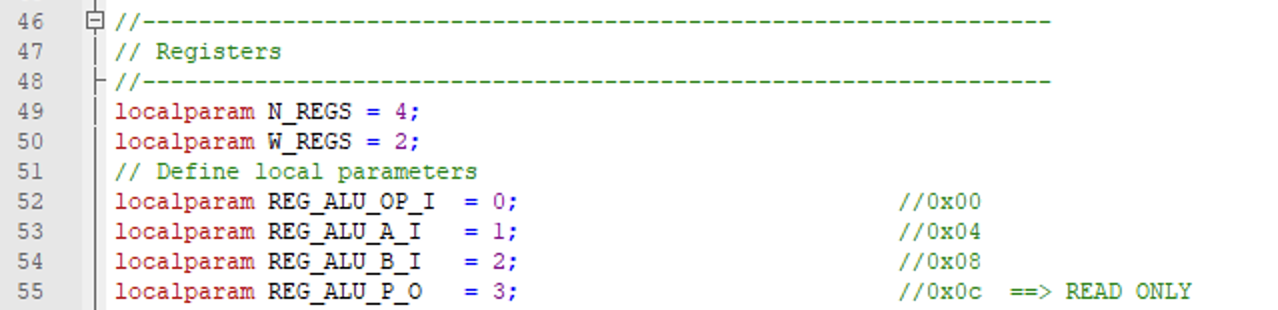


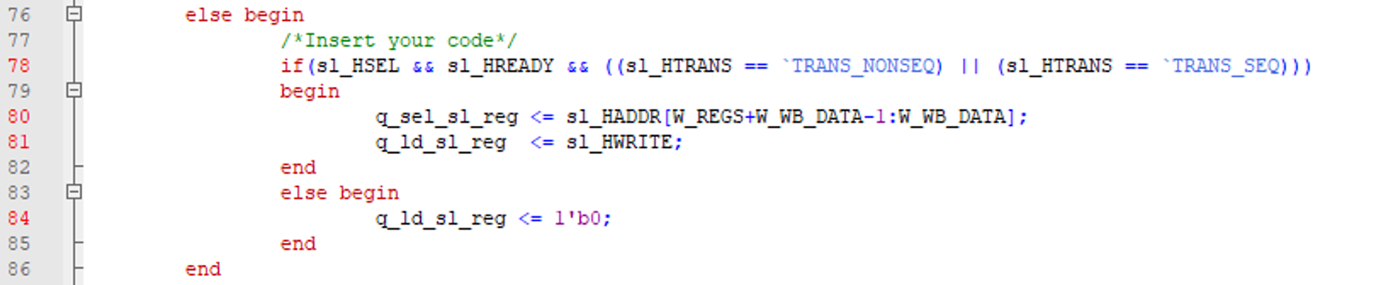
Fig. 1-2: Waveform

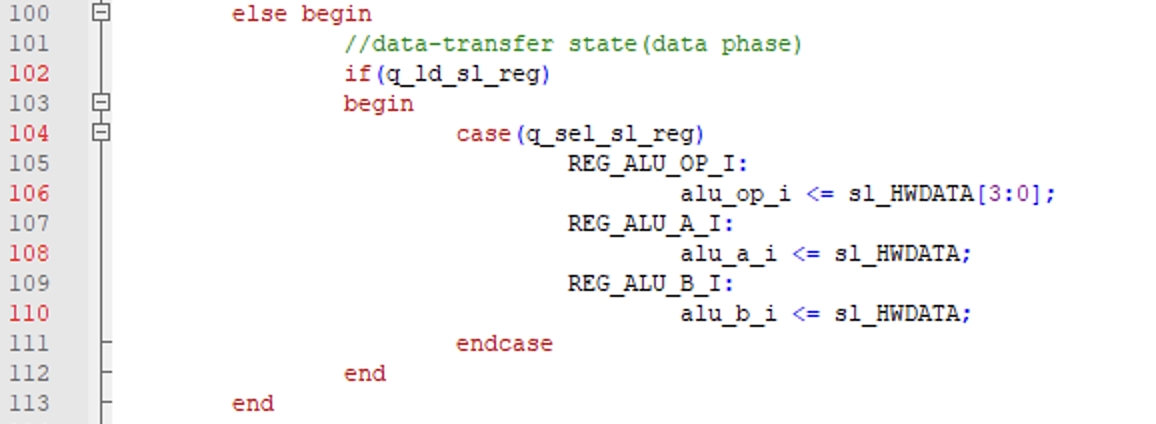
**Solution 1 : WaveForm**

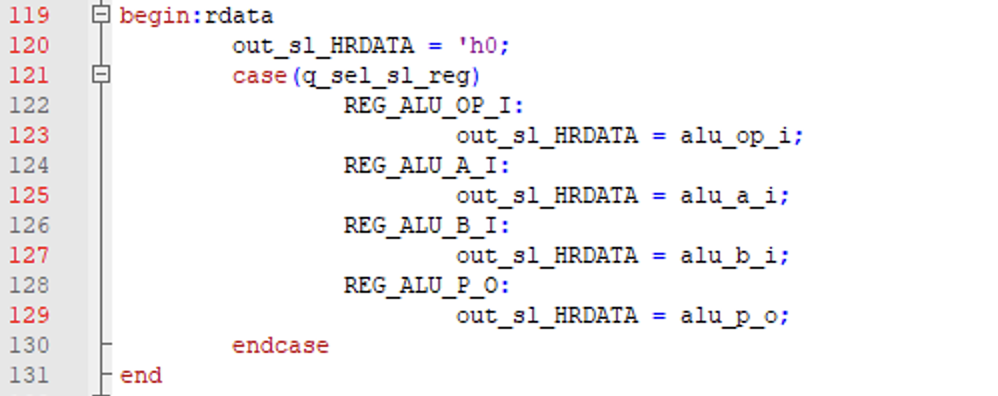
****

**Solution 1 : Modified RTL Code**

****

****

****

****

**Problem 2 (15p): Multi-AHB-Slave system**

Implement a multi-AHB-slave system. Please see the description in the lecture note for details.



Fig. 2-1: An AHB wrapper of RISC-V Multiplier.



Fig. 2-2: Top module.

What you have to do:

* Design an AHB a multi-AHB-slave system that includes a master and two slaves, i.e. ALU and multiplier.
* Reuse riscv\_alu\_if.v from Problem 1
* Complete the missing codes in riscv\_multiplier\_if.v and top\_system.v.
* Submit your RTL files.
* Capture the waveform.

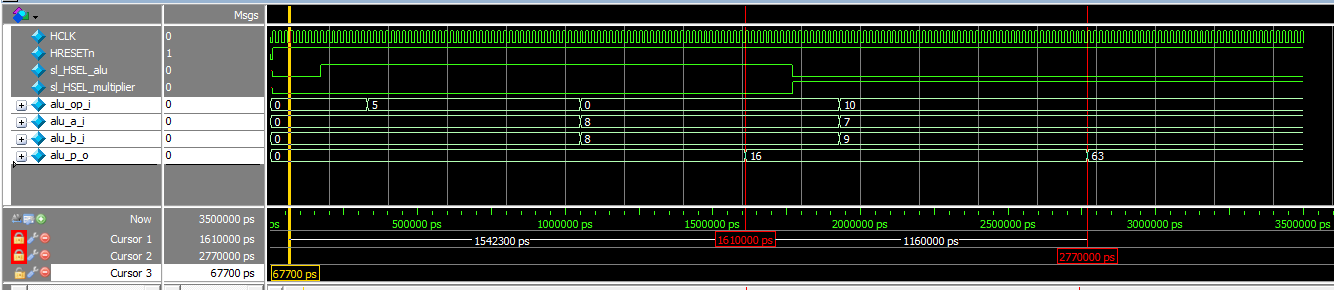
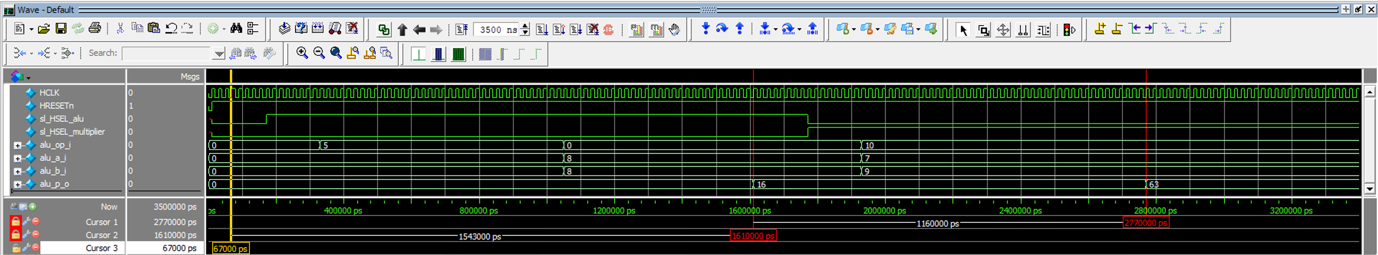
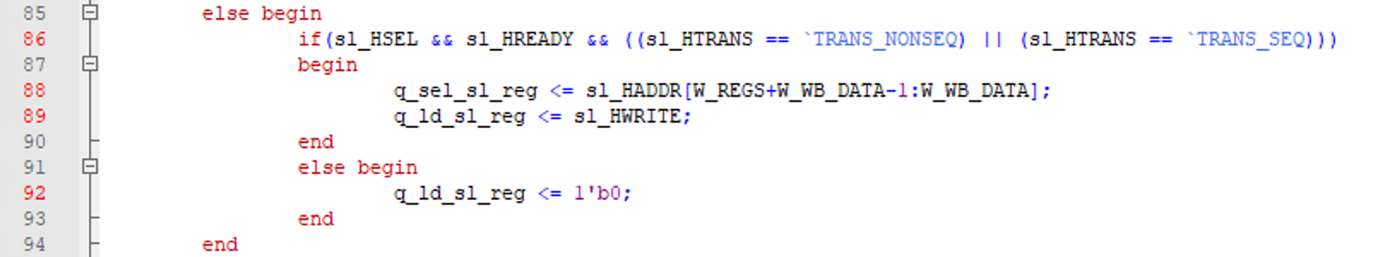


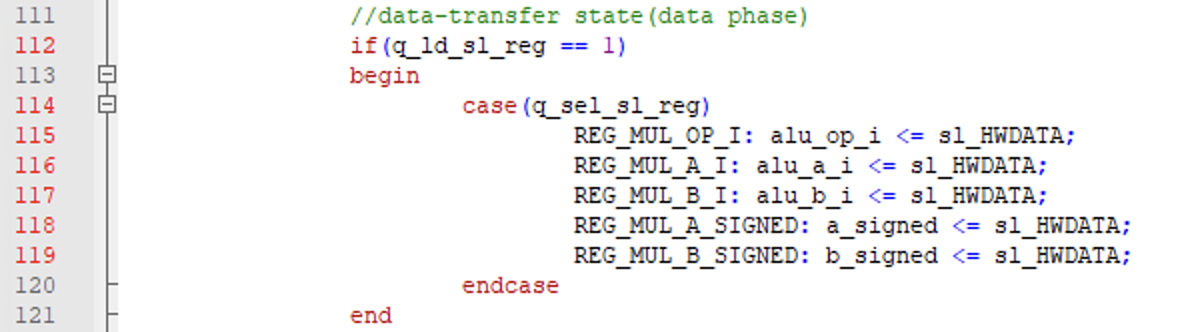
Fig. 2-3: Waveform.

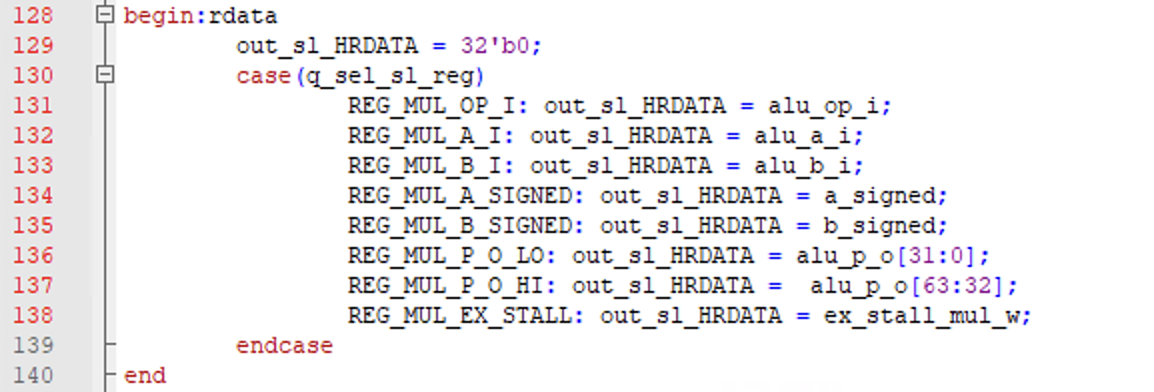
**Solution 2 : Waveform**

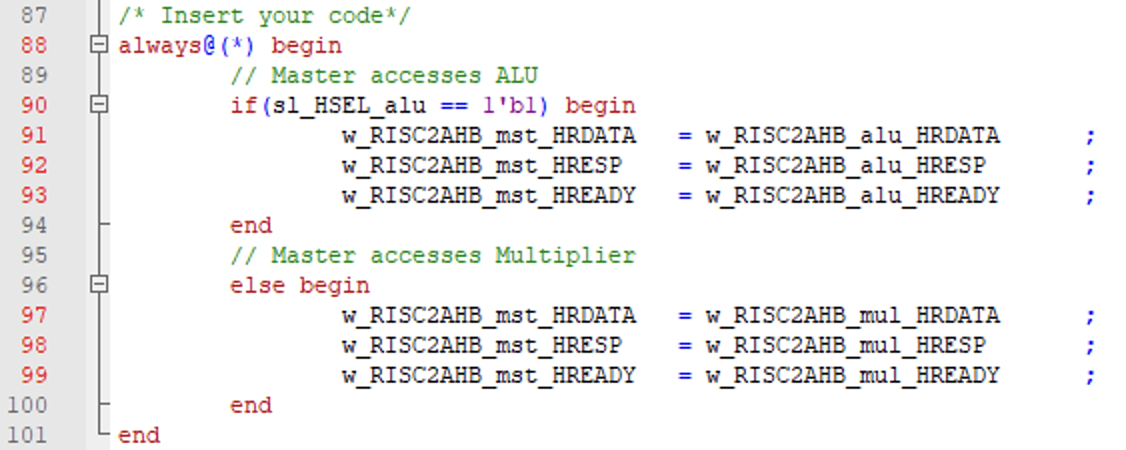
****

**Solution 2 : Modified RTL Code**

****

****

****

****

**Problem 3 (5p): AHB Decoder**

Implement an AHB decoder. Please see the description in the lecture note for details.

What you have to do:

* Design a simple AHB decoder.
* Reuse riscv\_alu\_if.v, riscv\_multiplier\_if.v from Problem 1 and 2.
* Complete the missing codes in top\_system.v.
* Submit your RTL files.
* Capture the waveform.

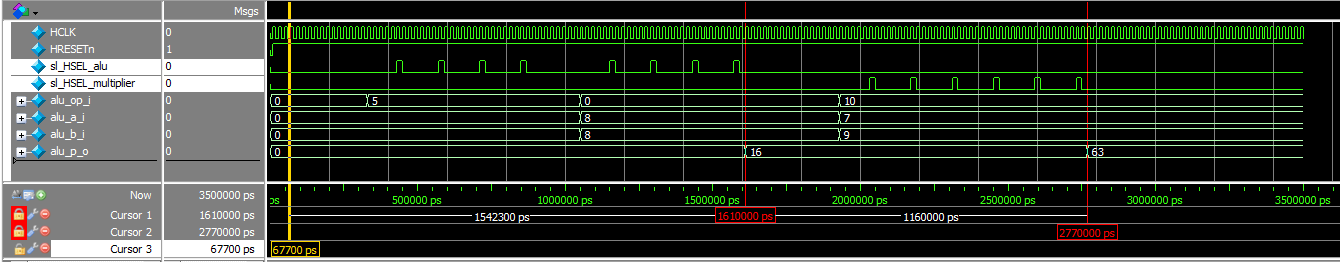
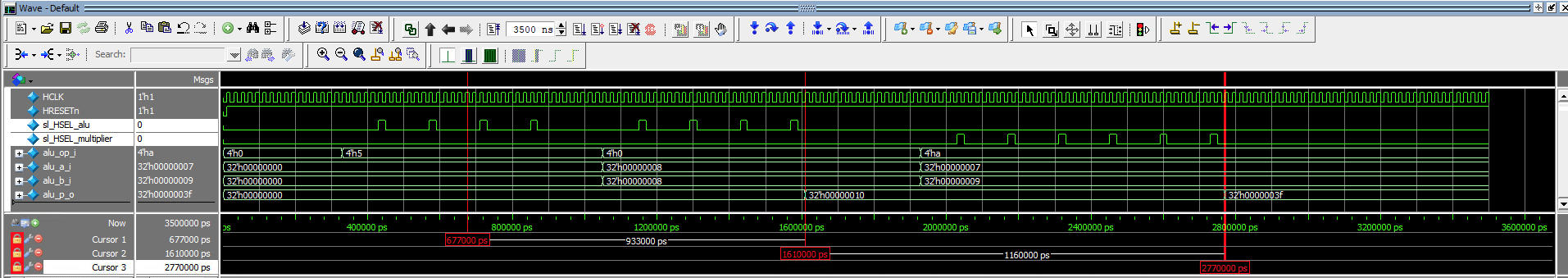
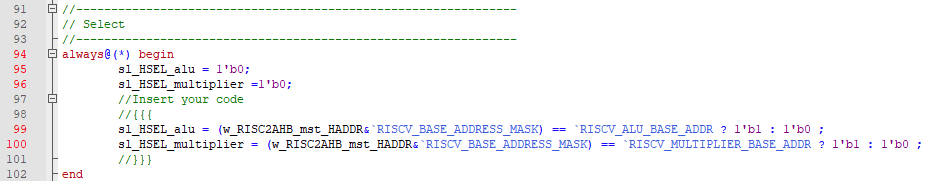


Fig. 3-1: Waveform

**Solution 3 : Waveform**

****

**Solution 3 : Modified RTL Code**

****

**Problem 4 (10p): AHB Bus Interconnection**

Implement a top system using an AHB Bus. Please see the description in the lecture note for details.



Fig. 4-1: A top system using an AHB bus.

What you have to do:

* Design a top stem using an AHB bus by completing the top module (top\_system.v).
* Submit your RTL files (top\_system.v)
* Capture the waveform.

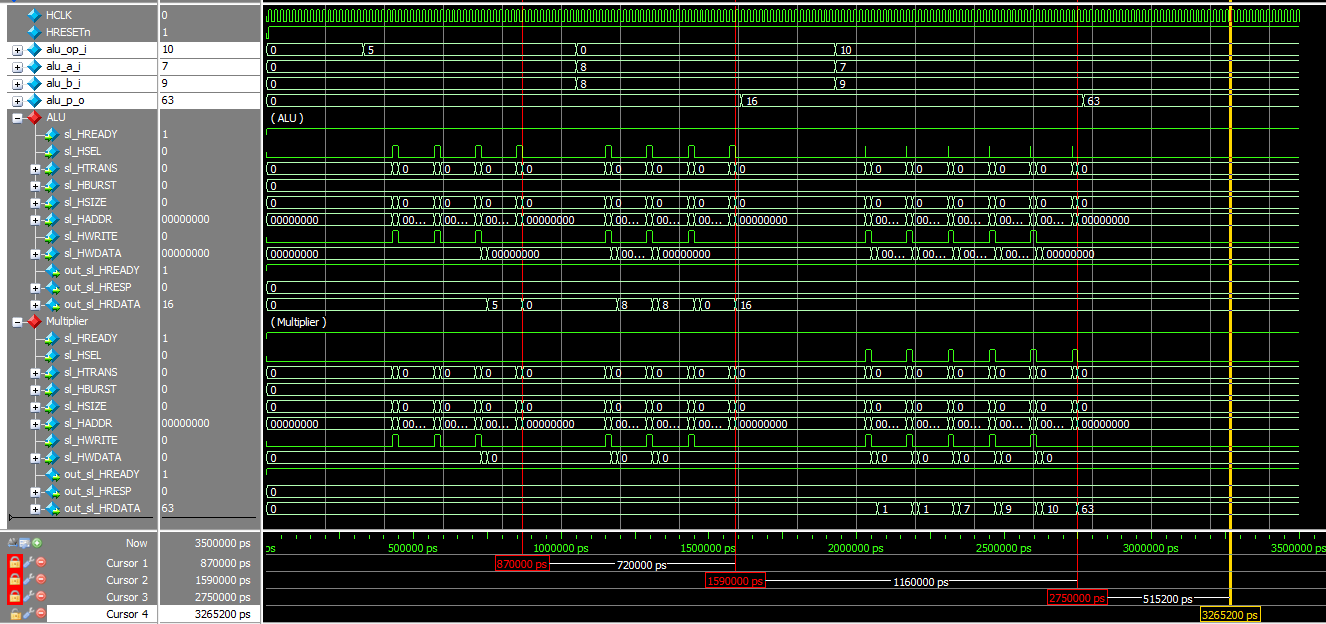
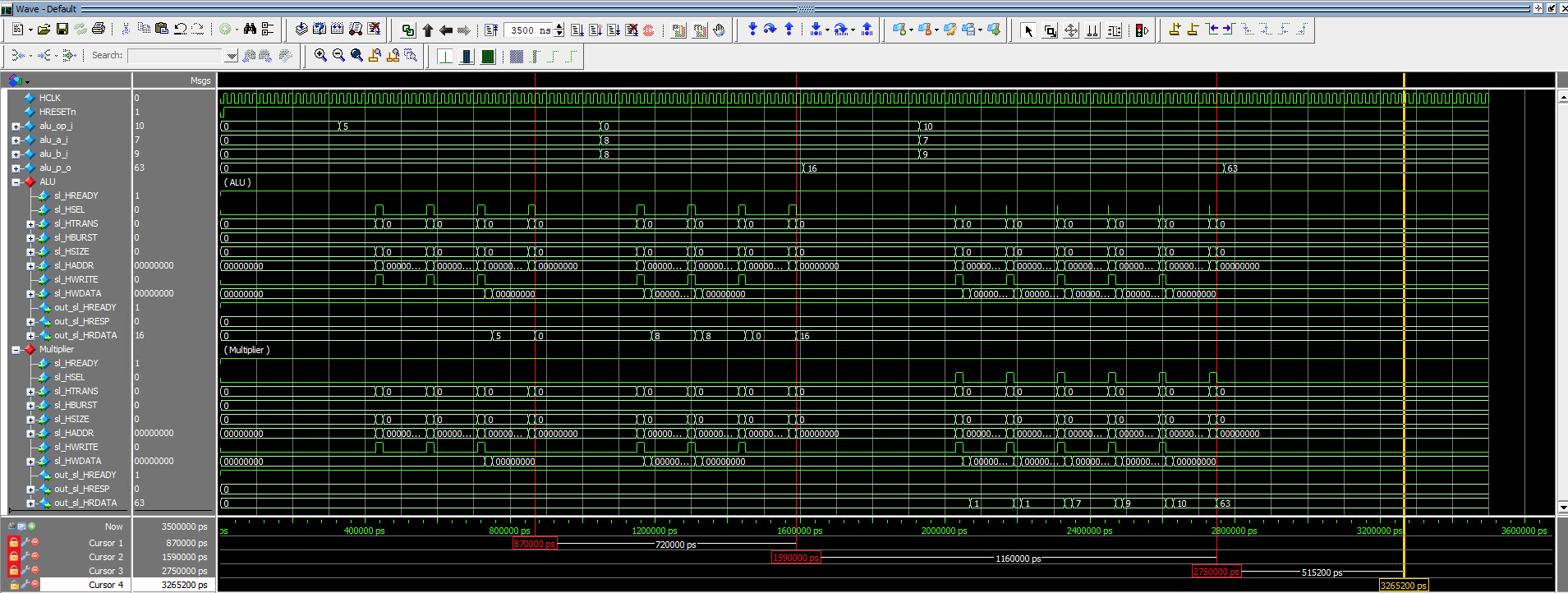
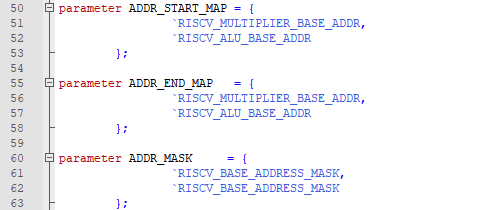


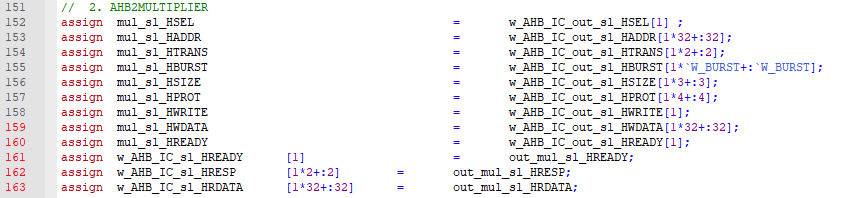
Fig. 4-2: Waveform

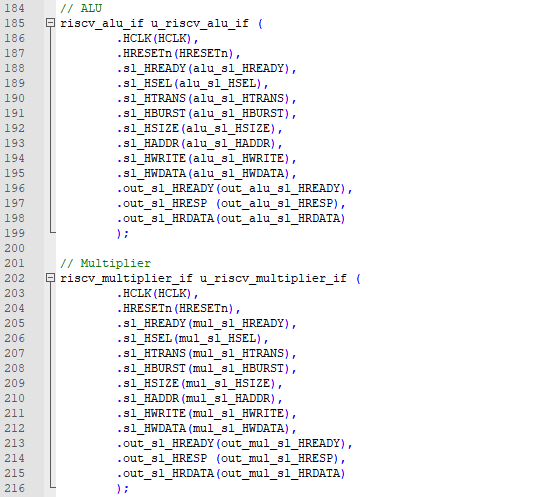
**Solution 4 : Waveform**

****

**Solution 4 : Modified RTL Code**

****

****

****

**Problem 5 (3p) (Optional): Bonus**

1. At an address decoding stage in a custom AHB slave, two registers q\_sel\_sl\_reg and q\_lg\_sl\_reg, are updated:

* q\_sel\_sl\_reg: store the address of a register that an AHB master accesses.
* q\_lg\_sl\_reg: define READ/WRITE mode.

Explain the codes used to update these registers in Problems 1 and 2.

1. The file “map.v” globally defines the addresses of all registers of slaves that are visible to masters. Each address consists of a base address and an offset. Explain how those addresses are used in an SoC system (e.g., Problem 4) when a master issues a Read/Write request to a slave.

**Solution 5.1. :**

* q\_sel\_sl\_reg : Extract the W\_REGS bits length bits in the sl\_HADDR W\_WB\_DATA off from the LSB. This extraction will result the q\_sel\_sl\_reg to have the address of the register of slave an AHB master access. W\_WB\_DATA means the interval of the register address each and W\_REGS means the width of number of registers the slave has. Eventually, extraction will derive the index of address the AHB master access in the AHB slave register.
* q\_lg\_sl\_reg : If the sl\_HWRITE signal is HIGH, the master wants to write data to slave. And if the sl\_HWRITE signal is LOW, the master wants to read data from slave. So, just directly substituting the sl\_HWRITE to the q\_lg\_sl\_reg, we can know the READ/WRITE mode for the desired register in slave.

**Solution 5.2. :**

Step 1 – An AHB master issues the access of the register in the slave and the address the AHB master issued moves to the AHB decoder.

Step 2 – An AHB decoder gets the address from the master and verify the address to determine which slave to connect or use.

Step 3 – By the given address map, decoder computes the base address of the given address with the given address mask and verify the base address for whether base address is in the interval of start address and end address.

Step 4 – After the verification, the AHB interconnection connects correct slave to the master corresponding of result of decoding the address.