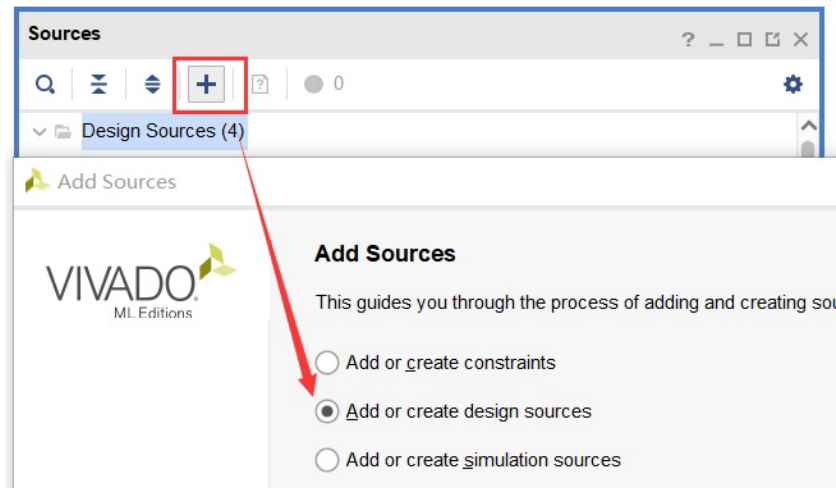


替换文件

1. 使用当前目录下的.v文件替换lab2中的.v文件
2. 使用当前目录下的.xdc文件替换lab2中的.xdc文件
3. 使用当前目录下的.hex文件替换lab2中code\core目录下的.hex文件
4. 使用当前目录下的.hex文件替换lab2中Exp2.sim\sim_1\behav\xsim中的.hex文件
5. 将uart_buffer.v和UART_TX_CTRL.vhd放入lab2中的code\auxillary目录，并在vivado的sources内添加这两个文件：



修改RV32Core.v中的接口

RV32core

由

```
module RV32core(  
    input debug_en, // debug enable  
    input debug_step, // debug step clock  
    input [6:0] debug_addr, // debug address  
    output[31:0] debug_data, // debug data  
    input clk, // main clock  
    input rst, // synchronous reset  
    input interrupter // interrupt source, for future use  
);
```

修改为：

```

module RV32core(
    input debug_en, // debug enable
    input debug_step, // debug step clock
    input [6:0] debug_addr, // debug address
    output[31:0] debug_data, // debug data
    output [7:0] sim_uart_char_out, // uart character
    output sim_uart_char_valid, // control valid
    input clk, // main clock
    input rst, // synchronous reset
    input interrupter // interrupt source, for future use
);

```

ROM

由

```
ROM_D inst_rom(.a(PC_IF[8:2]),.spo(inst_IF));
```

修改为：

```
ROM_D inst_rom(.a(PC_IF[9:2]),.spo(inst_IF));
```

RAM

由

```

RAM_B data_ram(.addra(ALUout_MEM),.clka(debug_clk),.dina(Dataout_MEM),
    .wea(mem_w_MEM),.rea(mem_r_MEM),.douta(RAMout_MEM),
    .mem_u_b_h_w(u_b_h_w_MEM),.l_access_fault(l_access_fault_MEM),
    .s_access_fault(s_access_fault_MEM)
);

```

修改为：

```

RAM_B data_ram(.addra(ALUout_MEM),.clka(debug_clk),.dina(Dataout_MEM),
    .wea(mem_w_MEM),.douta(RAMout_MEM),.mem_u_b_h_w(u_b_h_w_MEM),
    .sim_uart_char_out(sim_uart_char_out),
    .sim_uart_char_valid(sim_uart_char_valid),
    .l_access_fault(l_access_fault_MEM), .s_access_fault(s_access_fault_MEM)
);

```