

整体设计实现冒泡法排序，根据冒泡法的设计原理，n 个数据比较大小，将第一个元素和第二个元素进行比较，若为逆序则将两个元素交换，然后比较第二个元素和第三个元素。依次类推，直至第 n-1 个元素和第 n 个元素进行比较为止。上述过程称为第一趟冒泡排序，其结果使最大值元素被放置在最后一个位置（第 n 个位置）。然后进行第二趟冒泡排序，对前 n-1 个元素进行同样操作，其结果是使第二大元素被放置在倒数第二个位置上（第 n-1 个位置）。第 n-1 趟可以实现 n 个数据按顺序排列的结果。

## 1. 指令设计

在之前设计内容基础上，参考需要指令的汇编指令，根据设计要求，新增运行指令文件 instruction\_memory3.vhd，设计冒泡排序汇编指令转换为 16 进制数据内容如下

```
MOV R0, #0x20    转换为 16 进制数据"E3A00020"
MOV R2, #0x01    转换为 16 进制数据"E3A02001"
MOV R2, #0x00    转换为 16 进制数据"E3A02000"
MOV R1, #0x00    转换为 16 进制数据"E3A01000"
LDR R3, 0(R0)    转换为 16 进制数据"E6103000"
LDR R4, 0(R0, #1) 转换为 16 进制数据"E6104001"
CMP R3, R4       转换为 16 进制数据"E1530004"
BLT loop        转换为 16 进制数据"BA000002"
STR R4, 0(R0)    转换为 16 进制数据"E6004000"
STR R3, 0(R0, #1) 转换为 16 进制数据"E6003001"
ADD R0, R0, #1   转换为 16 进制数据"E2800001"
ADD R1, R1, #1   转换为 16 进制数据"E2811001"
CMP R1, 0x07     转换为 16 进制数据"E3510007"
```

```
BLT loop        转换为 16 进制数据"BAFFFFFF6"
ADD R2, R2, #1   转换为 16 进制数据"E2822001"
CMP R2, 0x07     转换为 16 进制数据"E3520007"
MOV R0, #0x20    转换为 16 进制数据"E3A00020"
BEN while        转换为 16 进制数据"FAFFFFFF0"
```

设计代码对应如下图所示

```
21      result (0) := x"E3A00020";
22      result (1) := x"E3A02001";
23      result (2) := x"E3A02000";
24      result (3) := x"E3A01000";
25      result (4) := x"E6103000";
26      result (5) := x"E6104001";
27      result (6) := x"E1530004";
28      result (7) := x"BA000002";
29      result (8) := x"E6004000";
30      result (9) := x"E6003001";
31      result (10) := x"E2800001";
32      result (11) := x"E2811001";
33      result (12) := x"E3510007";
34      result (13) := x"BAFFFFFF6";
35      result (14) := x"E2822001";
36      result (15) := x"E3520007";
37      result (16) := x"E3A00020";
38      result (17) := x"FAFFFFFF0";
```

## 2. 仿真设计

根据之前 part5 部分，修改 sim=3 可以运行 instruction\_memory3.vhd 文件，sim 仿真运行修改位置在仿真代码 MCU\_TOP.vht 中，如下图所示第 46 行位置，根据运行的仿真指令修改数值既可运行对应的文件。

```
45      il : MCU_TOP
46      generic map (sim=>3)
47      PORT MAP (
48      -- list connections between master ports and signals
49      CLK => CLK,
50      Reset => Reset
51      );
52      init : PROCESS
```

根据设计要求需要对内存数据初始化，作为需要排序的序列 TAB = {3, 107, 27, 12, 322, 155, 63}，修改代码如下图所示，需要注意的是内存数据 38 是数据 0，因此排序需要多增加一个数据。

```
40      result(32) := std_logic_vector(to_unsigned(3, 32));
41      result(33) := std_logic_vector(to_unsigned(107, 32));
42      result(34) := std_logic_vector(to_unsigned(27, 32));
43      result(35) := std_logic_vector(to_unsigned(12, 32));
44      result(36) := std_logic_vector(to_unsigned(322, 32));
45      result(37) := std_logic_vector(to_unsigned(155, 32));
```