

```
1    #include <iostream>
2    using namespace std;
3
4    int arr[10] = {0, 1, 2, 3, 4, 5, 6, 7, 8, 9};
5
6    ✓ void swap(int a, int b) {
7        int x = arr[a];
8        int y = arr[b];
9        arr[a] = y;
10       arr[b] = x;
11    }
12
13    int main(void) {
14        swap(5, 7);
15        swap(0, 1);
16    }
```

Figure 1: Leaf Procedure in C++

| | riscv1.asm | bubble_sort.asm | procedure_call.asm* |
|----|------------|-----------------|---------------------|
| 1 | main: | addi t0 x0 11 | |
| 2 | | addi t1 x0 22 | |
| 3 | | addi t2 x0 33 | |
| 4 | | addi a0 x0 5 | |
| 5 | | addi a1 x0 7 | |
| 6 | | jal swap | |
| 7 | | addi a0 x0 0 | |
| 8 | | addi a1 x0 1 | |
| 9 | | jal swap | |
| 10 | | j end | |
| 11 | swap: | addi sp sp -12 | |
| 12 | | sw t0 8(sp) | |
| 13 | | sw t1 4(sp) | |
| 14 | | sw t2 0(sp) | |
| 15 | | la t0 array | |
| 16 | | slli a0 a0 2 | |
| 17 | | slli a1 a1 2 | |
| 18 | | add a0 a0 t0 | |
| 19 | | add a1 a1 t0 | |
| 20 | | lw t1 0(a0) | |
| 21 | | lw t2 0(a1) | |
| 22 | | sw t1 0(a1) | |
| 23 | | sw t2 0(a0) | |
| 24 | | lw t2 0(sp) | |
| 25 | | lw t1 4(sp) | |
| 26 | | lw t0 8(sp) | |
| 27 | | addi sp sp 12 | |
| 28 | | jalr x0 ra 0 | |
| 29 | end: | addi s0 s0 122 | |
| 30 | | | |

Figure 2: Leaf Procedure in RISC-V

1 0 2 3 4 7 6 5 8 9

Figure 3: Expected Result

| | | | | |
|----|----|----|----|----|
| 0 | 01 | 00 | 00 | 00 |
| 4 | 00 | 00 | 00 | 00 |
| 8 | 02 | 00 | 00 | 00 |
| 12 | 03 | 00 | 00 | 00 |
| 16 | 04 | 00 | 00 | 00 |
| 20 | 07 | 00 | 00 | 00 |
| 24 | 06 | 00 | 00 | 00 |
| 28 | 05 | 00 | 00 | 00 |
| 32 | 08 | 00 | 00 | 00 |
| 36 | 09 | 00 | 00 | 00 |

Figure 4: Actual Result

| Wave - Default | | |
|------------------------|---------|---|
| | Msgs | |
| + /tb_cpu/u... 1 | 1 | (0 |
| + /tb_cpu/u... 0 | 0 | (1 |
| + /tb_cpu/u... 2 | 2 | (2 |
| + /tb_cpu/u... 3 | 3 | (3 |
| + /tb_cpu/u... 4 | 4 | (4 |
| + /tb_cpu/u... 7 | 7 | (5 |
| + /tb_cpu/u... 6 | 6 | (6 |
| + /tb_cpu/u... 5 | 5 | (7 |
| + /tb_cpu/u... 8 | 8 | (8 |
| + /tb_cpu/u... 9 | 9 | (9 |
| + /tb_cpu/u... 0000... | 0000... | (00000000 0...) (00000000..) (00000000 00000000 00000000...) (|

Figure 5: Data Memory (Memory Segment)

| Wave - Default | | |
|--|---------|---|
| | Msgs | |
| + /tb_cpu/uut/exe_stage/data_memory/memory[1000] | 0 | (0 |
| + /tb_cpu/uut/exe_stage/data_memory/memory[1004] | 0 | (0 |
| + /tb_cpu/uut/exe_stage/data_memory/memory[1008] | 0 | (0 |
| + /tb_cpu/uut/exe_stage/data_memory/memory[1012] | 33 | (0 |
| + /tb_cpu/uut/exe_stage/data_memory/memory[1016] | 22 | (0 |
| + /tb_cpu/uut/exe_stage/data_memory/memory[1020] | 11 | (0 |
| + /tb_cpu/uut/exe_stage/data_memory/memory[1024] | 0 | (0 |
| + /tb_cpu/uut/exe_stage/data_memory/memory | 0000... | (00000000 0...) (00000000..) (00000000 00000000 00000000...) (|

Figure 6: Data Memory (Stack Segment)

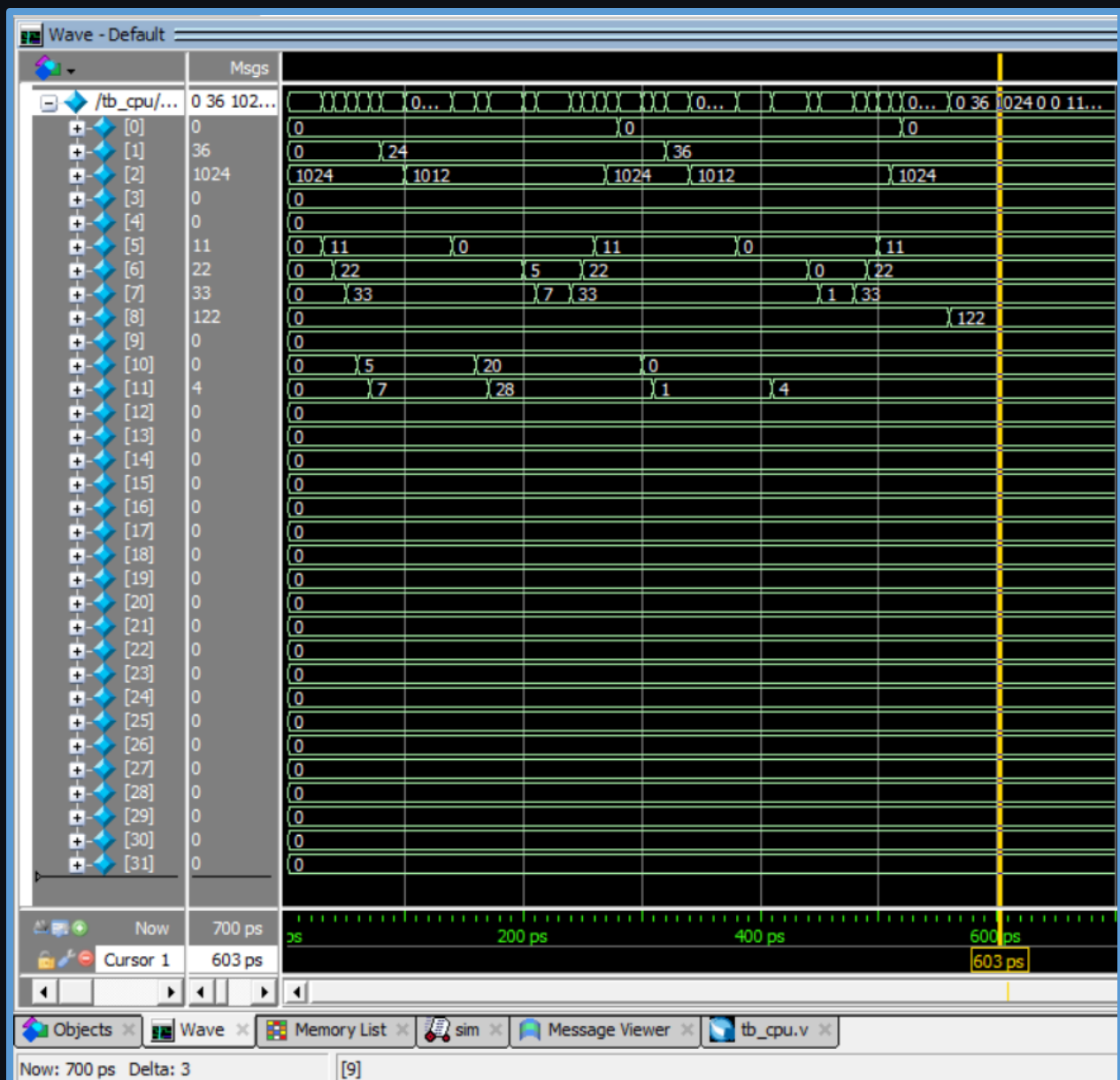


Figure 7: Register File

| | | | | |
|-----|----|----|----|----|
| 0 | 13 | 2c | 00 | 28 |
| 4 | 13 | 58 | 00 | 30 |
| 8 | 13 | 84 | 00 | 38 |
| 12 | 13 | 14 | 00 | 50 |
| 16 | 13 | 1c | 00 | 58 |
| 20 | 6f | 04 | 00 | 00 |
| 24 | 13 | 00 | 00 | 50 |
| 28 | 13 | 04 | 00 | 58 |
| 32 | 6f | 01 | 00 | 00 |
| 36 | 67 | 13 | 00 | 00 |
| 40 | 13 | d0 | bf | 10 |
| 44 | 23 | 21 | 8a | 00 |
| 48 | 23 | 11 | 8c | 00 |
| 52 | 23 | 01 | 8e | 00 |
| 56 | 33 | 00 | 00 | 28 |
| 60 | 13 | 00 | 40 | 29 |
| 64 | 93 | 08 | 80 | 52 |
| 68 | 93 | 08 | c0 | 5a |
| 72 | 33 | 00 | 8a | 52 |
| 76 | 33 | 00 | ca | 5a |
| 80 | 03 | 01 | 80 | 32 |
| 84 | 03 | 01 | c0 | 3a |
| 88 | 23 | 01 | cc | 02 |
| 92 | 23 | 01 | 8e | 02 |
| 96 | 03 | 01 | 80 | 38 |
| 100 | 03 | 11 | 80 | 30 |
| 104 | 03 | 21 | 80 | 28 |
| 108 | 13 | 30 | 80 | 10 |
| 112 | 6b | 00 | 40 | 00 |
| 116 | 13 | e8 | 01 | 42 |

Figure 8: Instruction Memory