

- Move the value from register 4 to the hardware unit for operand 1
- Move the value from register 13 to the hardware unit for operand 2
- Arrange for the ALU to perform addition
- Move the value from the hardware unit for result 2 (the low-order bits of the result) to register 4

Figure 8.8 An example sequence of steps that the functional units must execute to add values from general-purpose registers 4 and 13, and place the result in general-purpose register 4.

Each of the steps can be expressed as a single micro instruction in our example system. The instruction has bits set to specify which functional unit(s) operate when the instruction is executed. For example, Figure 8.9 shows a microcode program that corresponds to the four steps.

In the figure, each row corresponds to one instruction, which is divided into fields that each correspond to a functional unit. A field contains a command to be sent to the functional unit when the instruction is executed. Thus, commands determine which functional units operate at each step.

Instr.	ALU			OP ₁	OP ₂	RES ₁	RES ₂	REG. INTERFACE					
1	0	0	0	1	0	0	0	0	1	0	1	0	0
2	0	0	0	0	1	0	0	0	1	1	1	0	1
3	0	0	1	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	1	1	0	0	1	0	0

instructions with thirteen bits per instruction. Each instruction corresponds to a step listed in Figure 8.8."