Chapter 6

Architecture

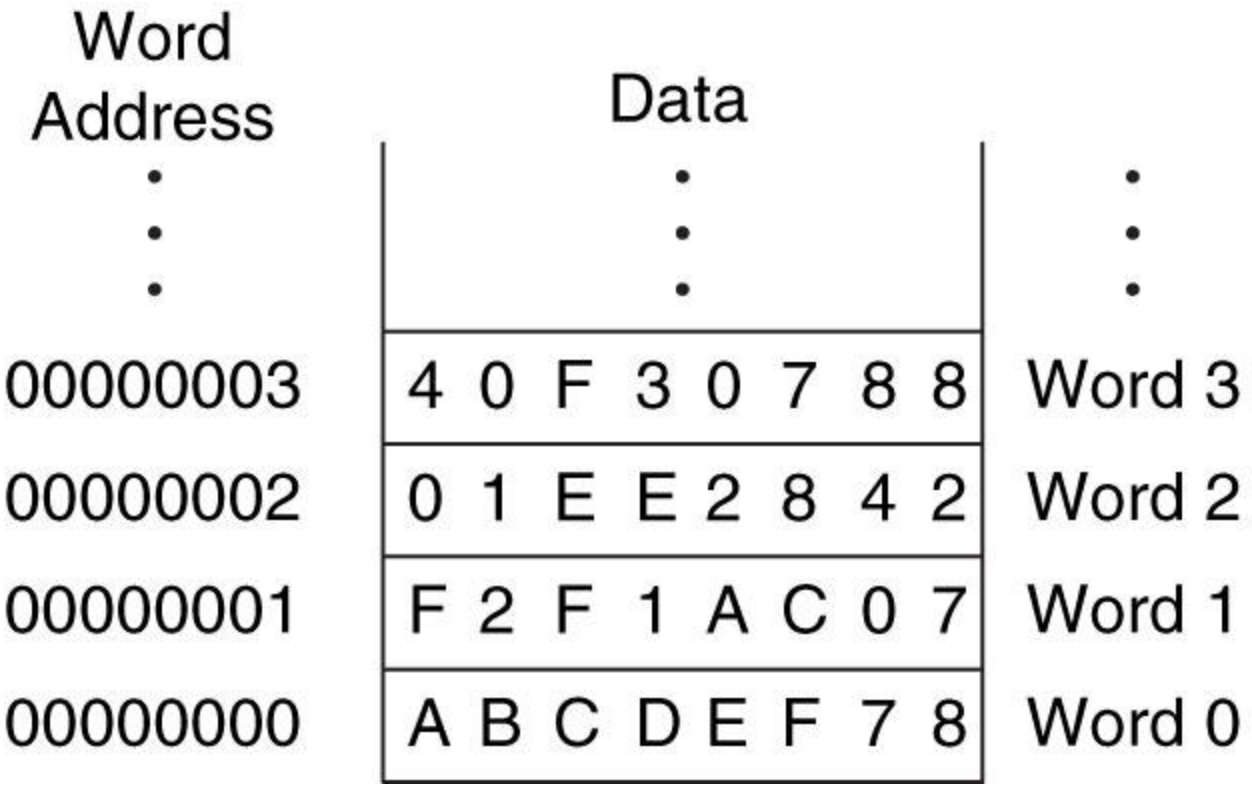


Figure 6.1 Word-addressable memory

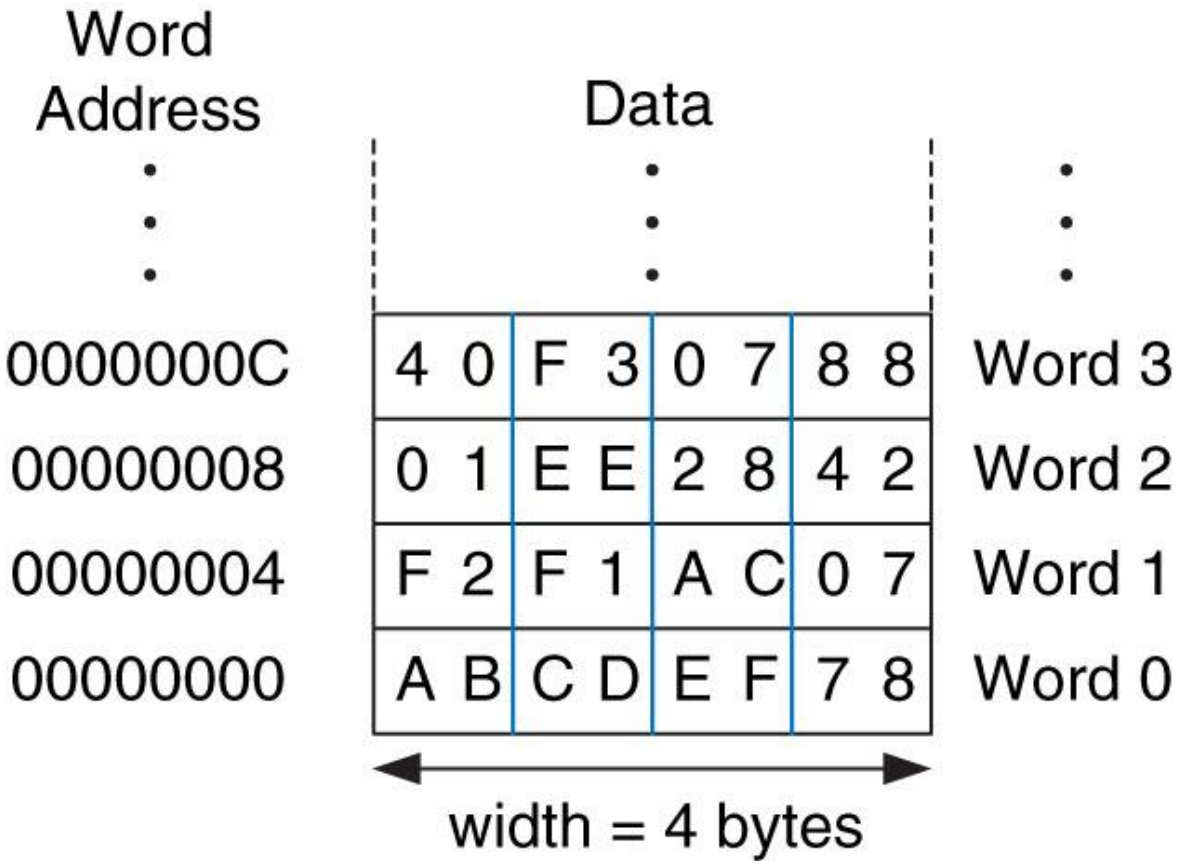


Figure 6.2 Byte-addressable memory

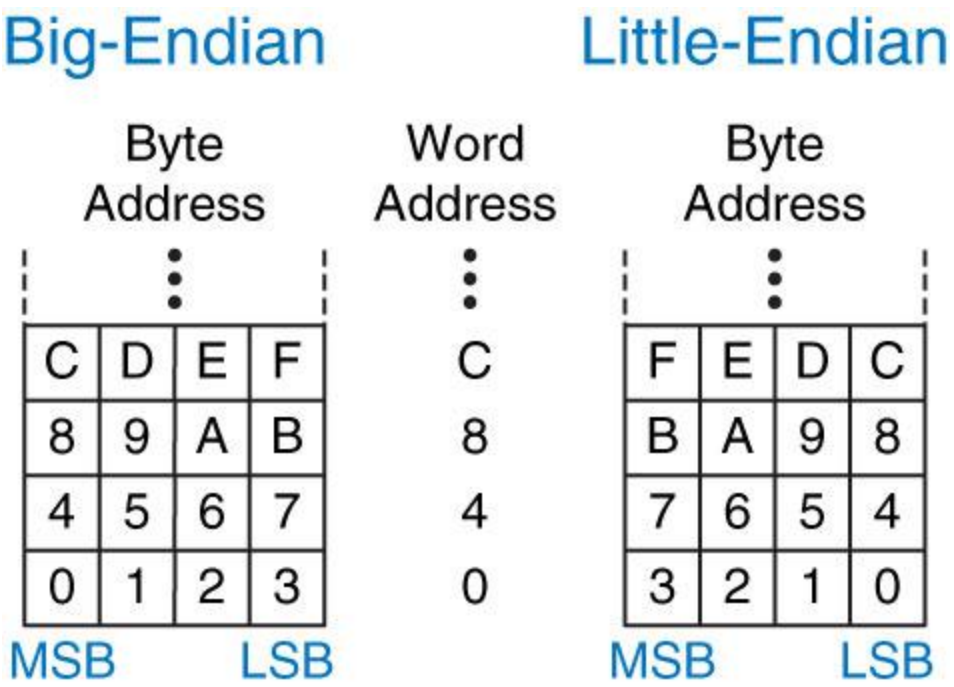


Figure 6.3 Big- and little-endian memory addressing

Figure 6.4 Big-endian and little-endian data storage

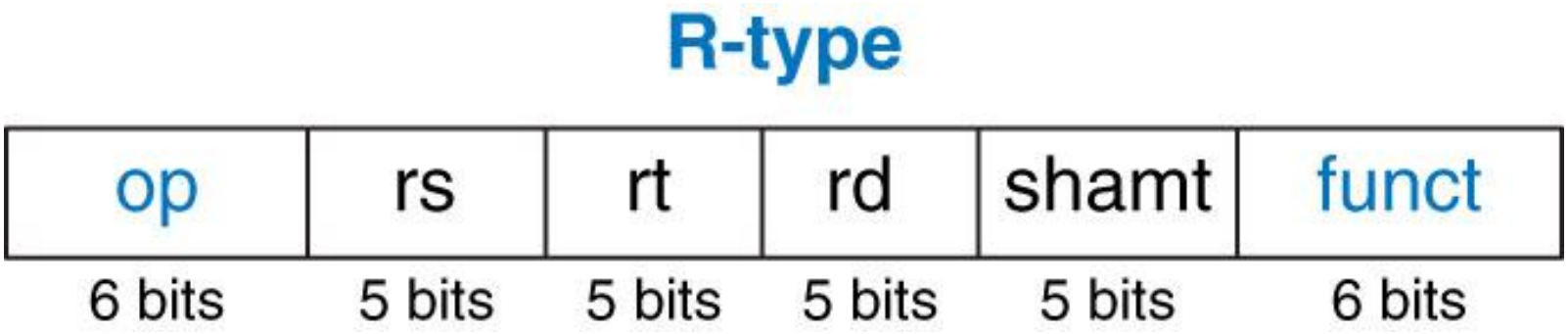


Figure 6.5 R-type machine instruction format

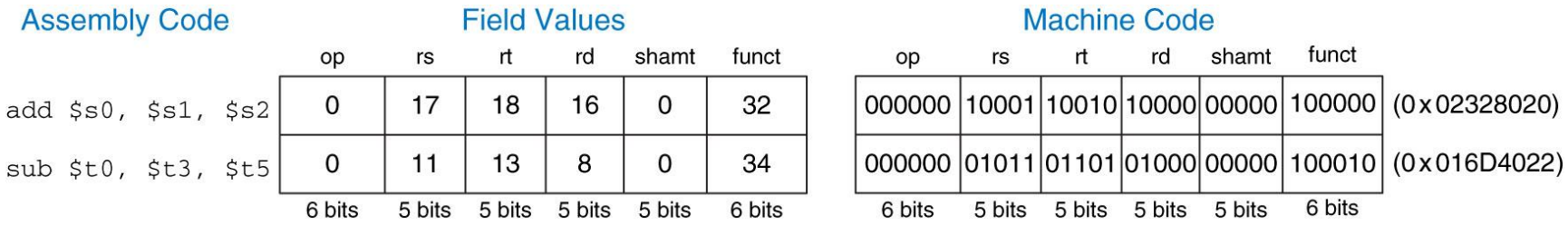


Figure 6.6 Machine code for R-type instructions

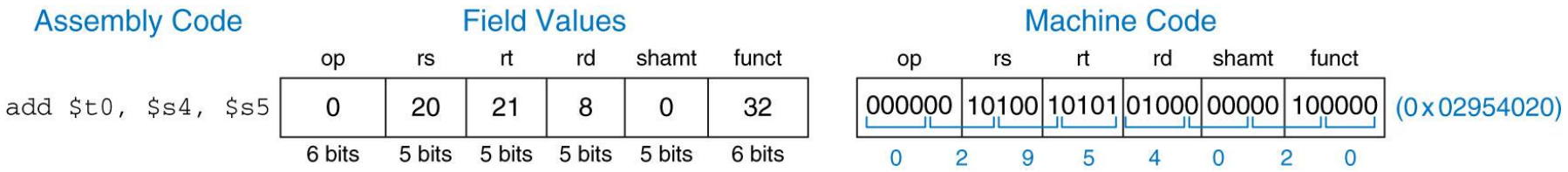


Figure 6.7 Machine code for the R-type instruction of Example 6.3

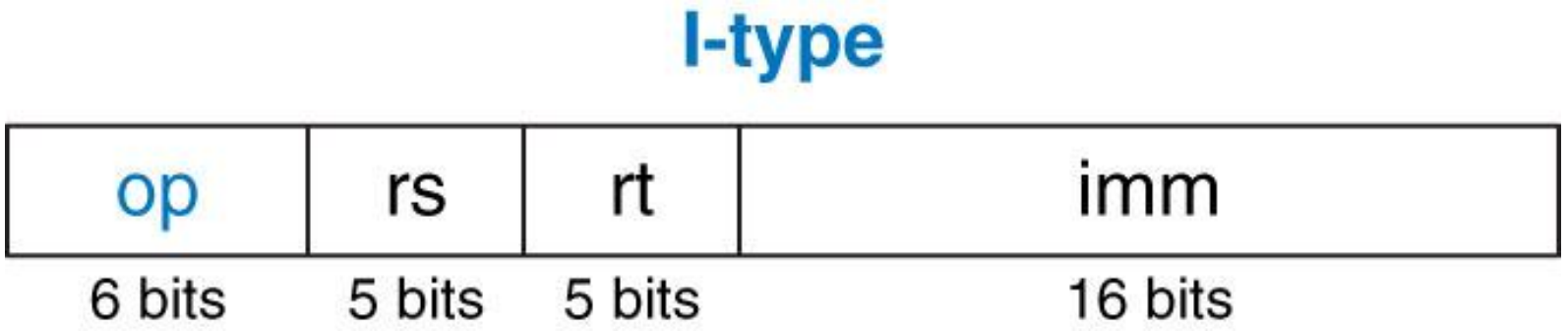


Figure 6.8 I-type instruction format

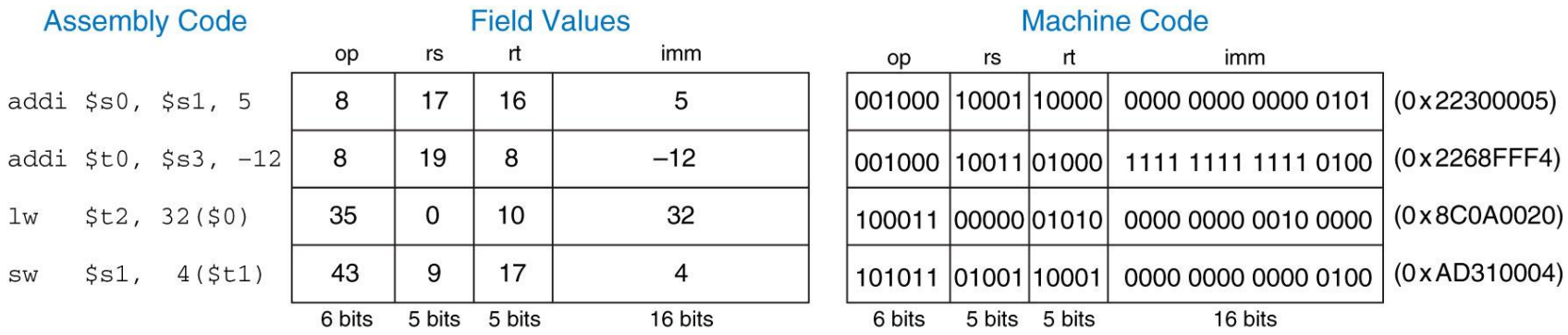


Figure 6.9 Machine code for I-type instructions

Figure 6.10 Machine code for an I-type instruction

Figure 6.11 J-type instruction format

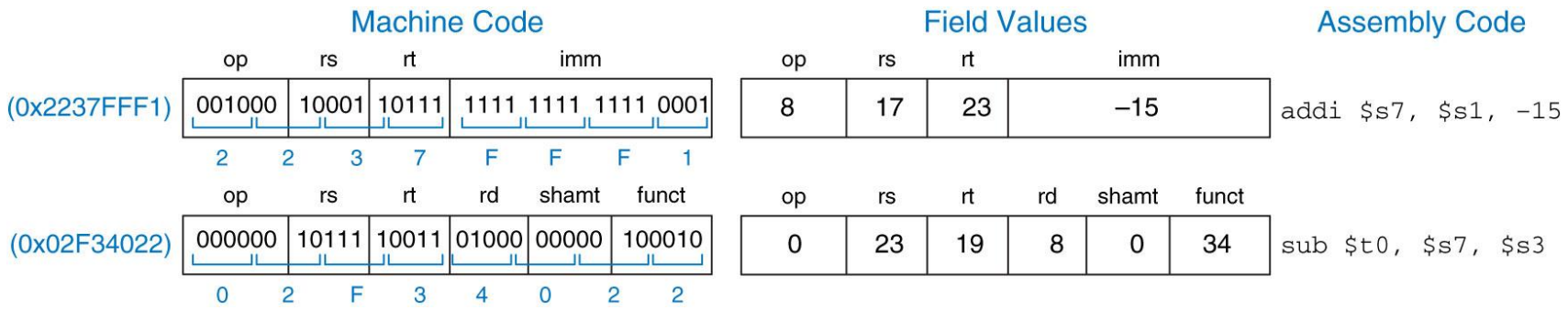


Figure 6.12 Machine code to assembly code translation

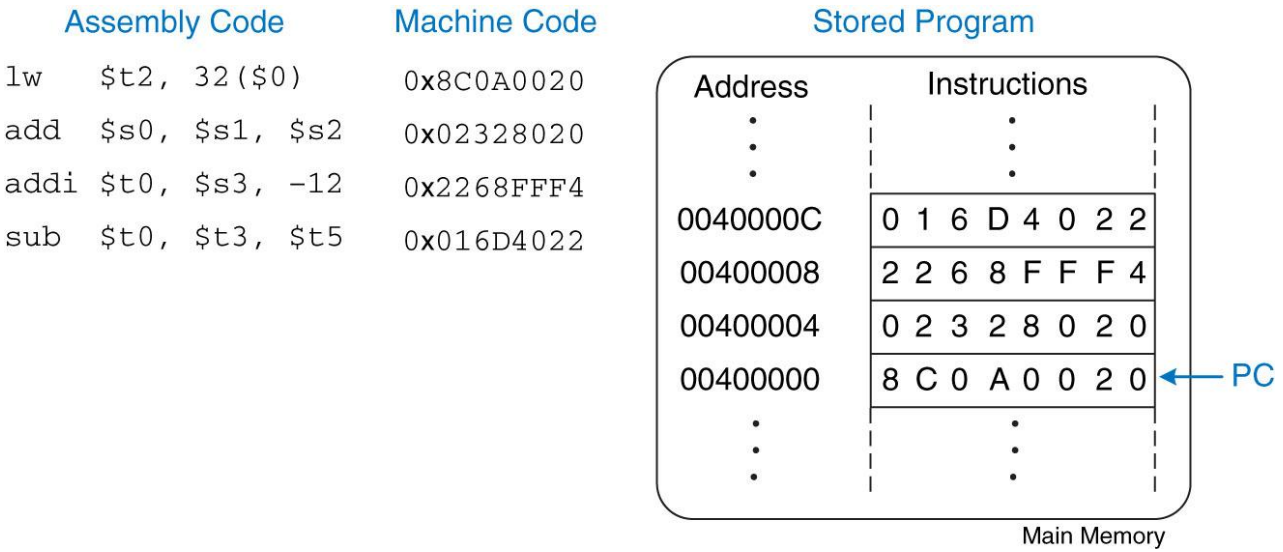


Figure 6.13 Stored program

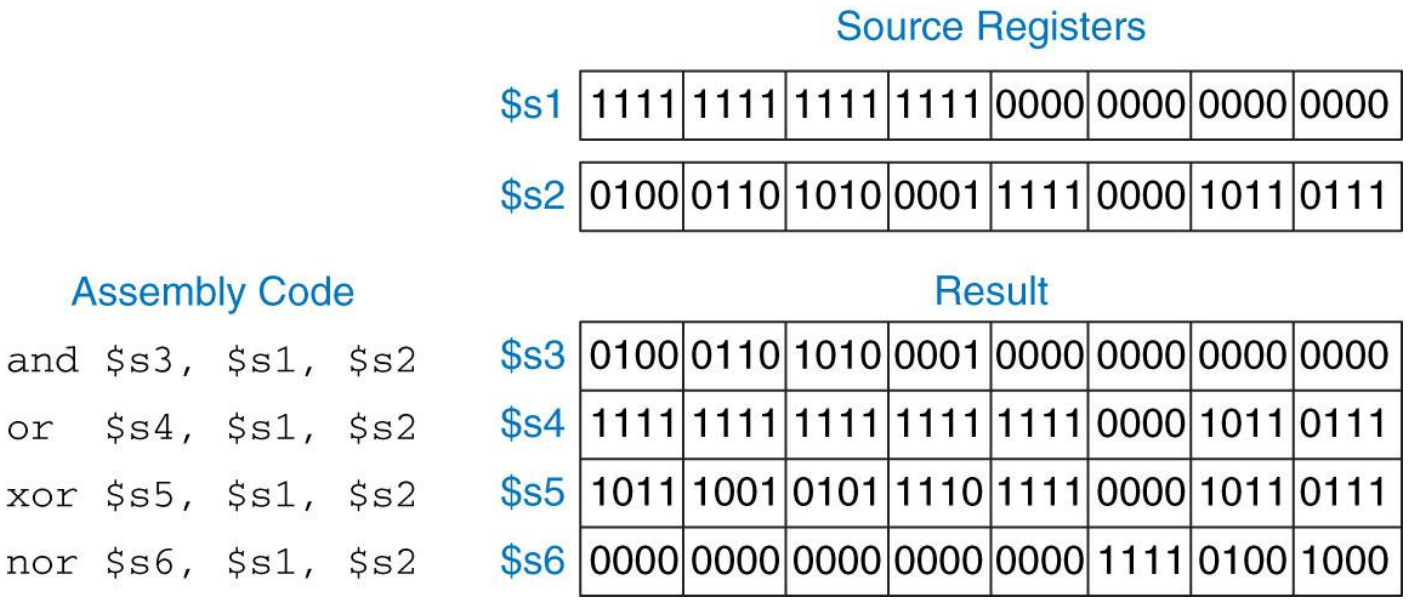


Figure 6.14 Logical operations

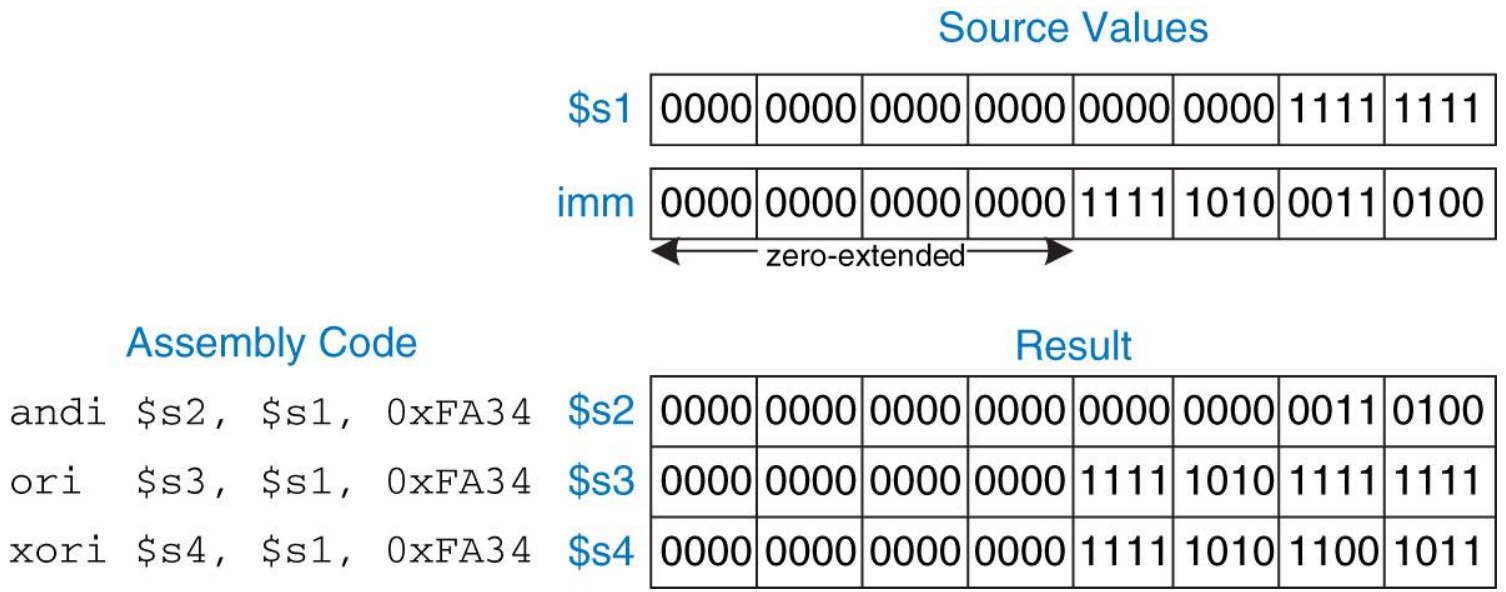


Figure 6.16 Shift instruction machine code

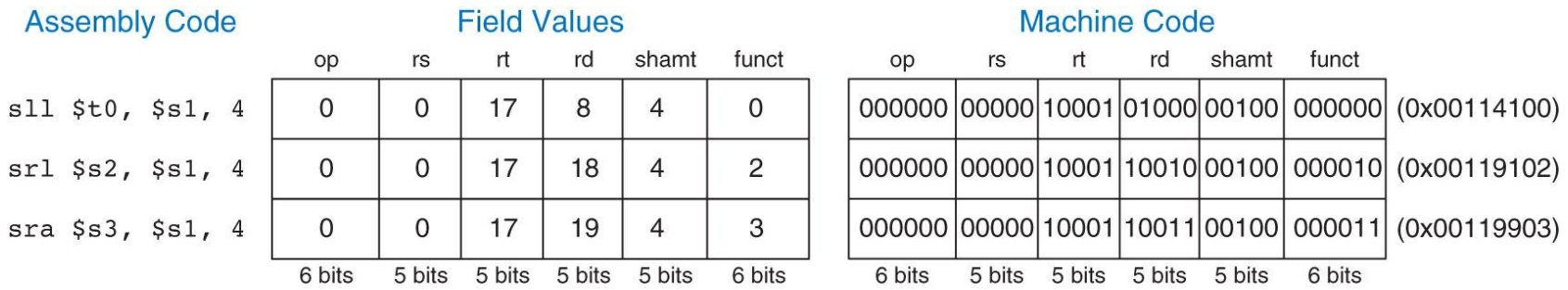


Figure 6.16 Shift instruction machine code

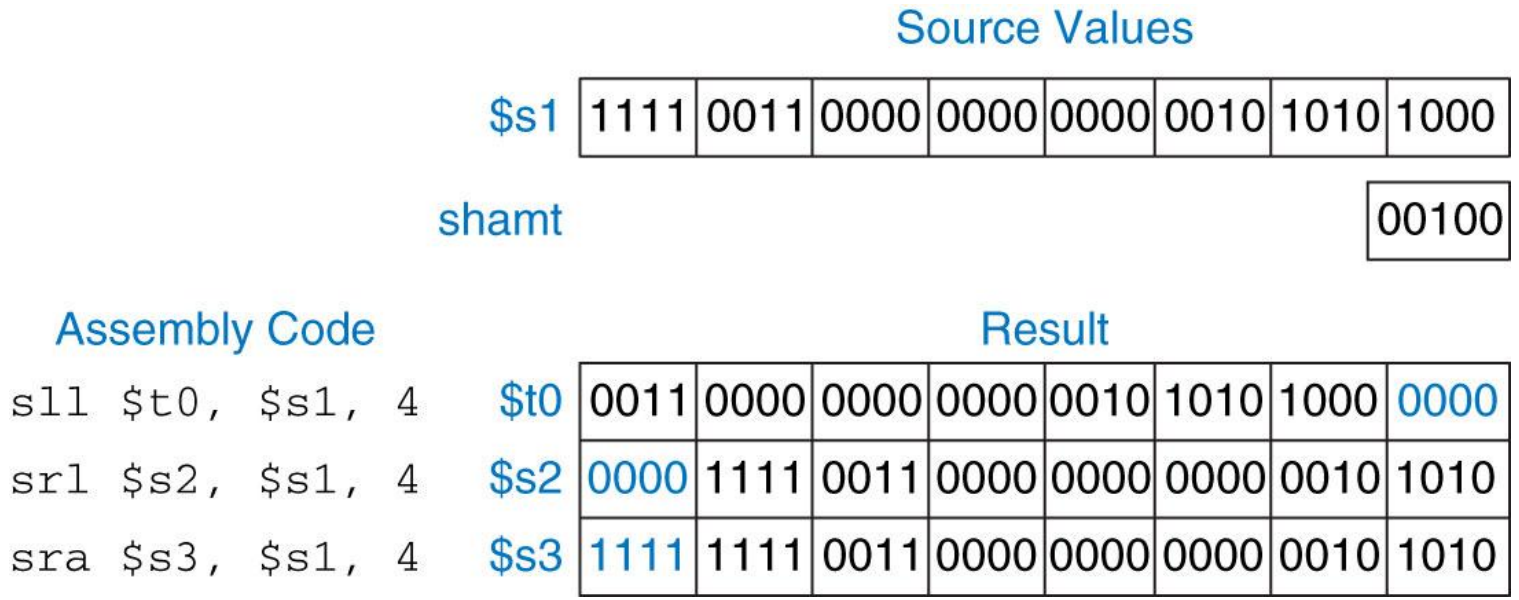


Figure 6.17 Shift operations

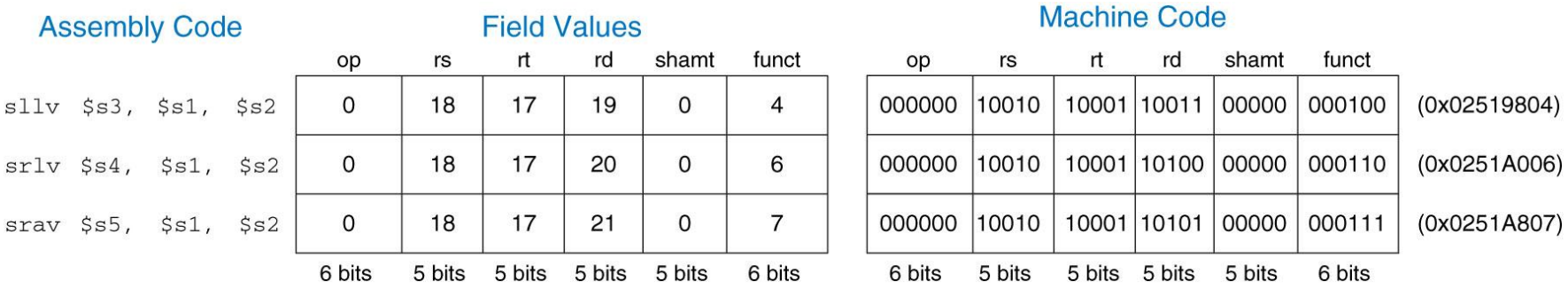


Figure 6.18 Variable-shift instruction machine code

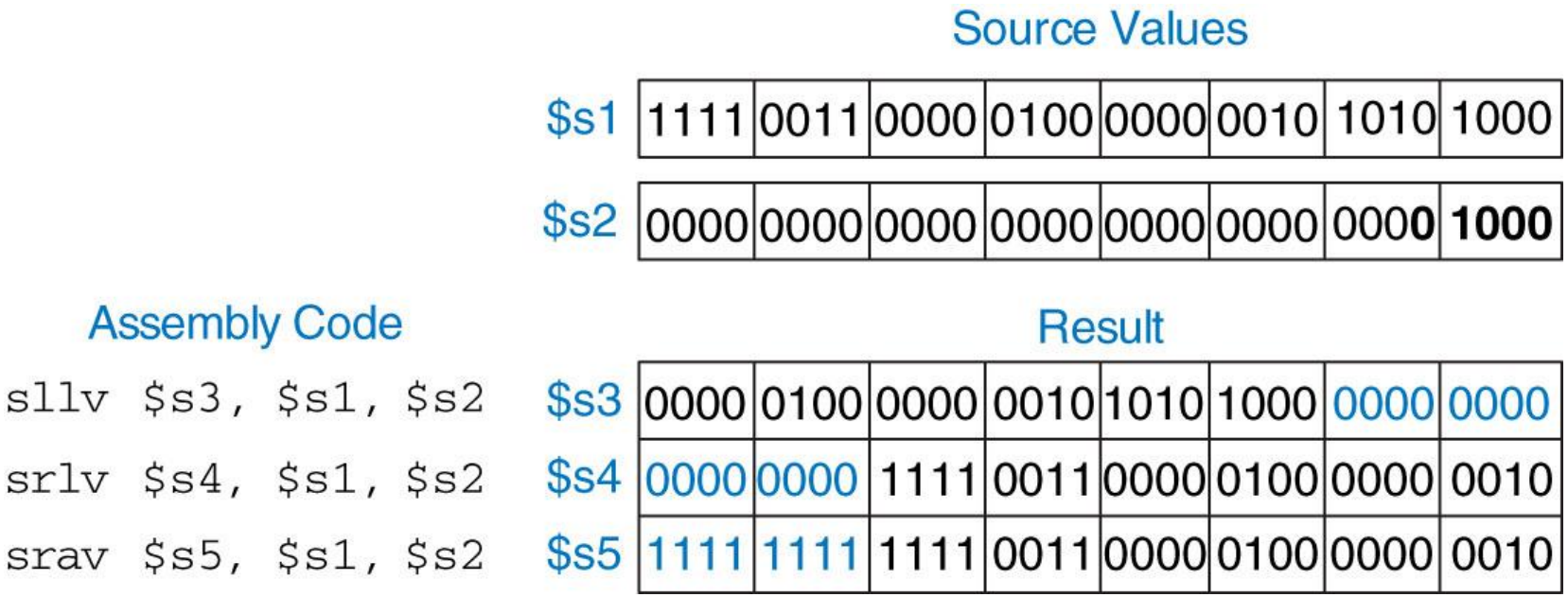


Figure 6.19 Variable-shift operations

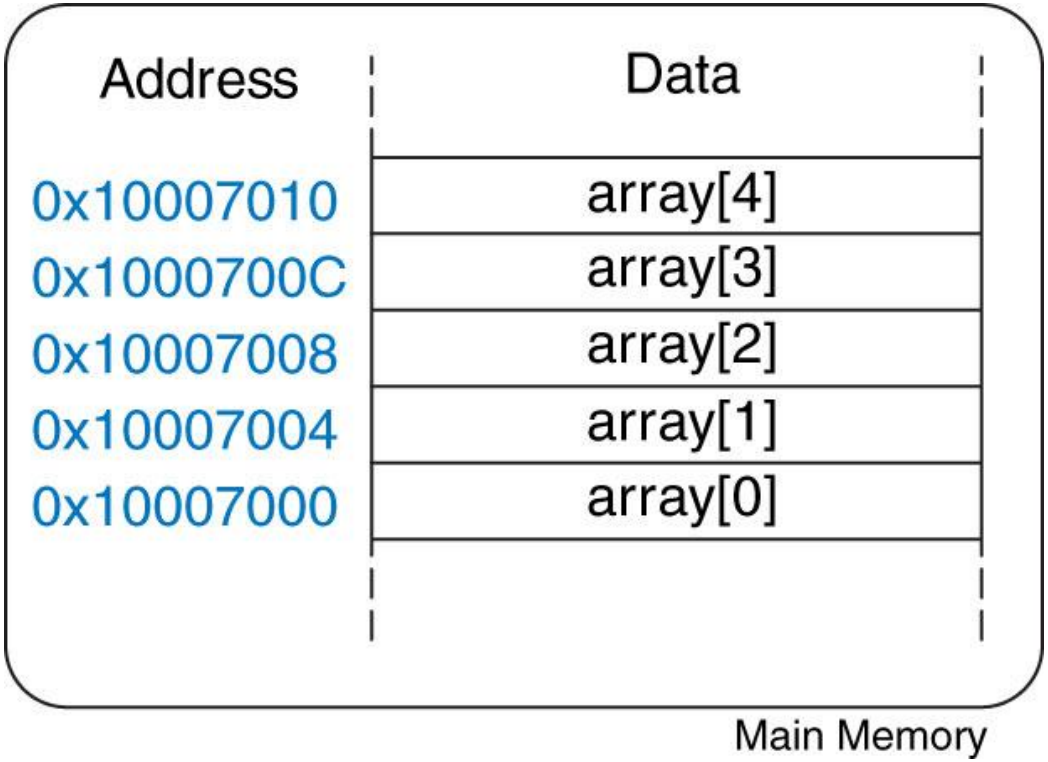


Figure 6.20 Five-entry array with base address of 0x10007000

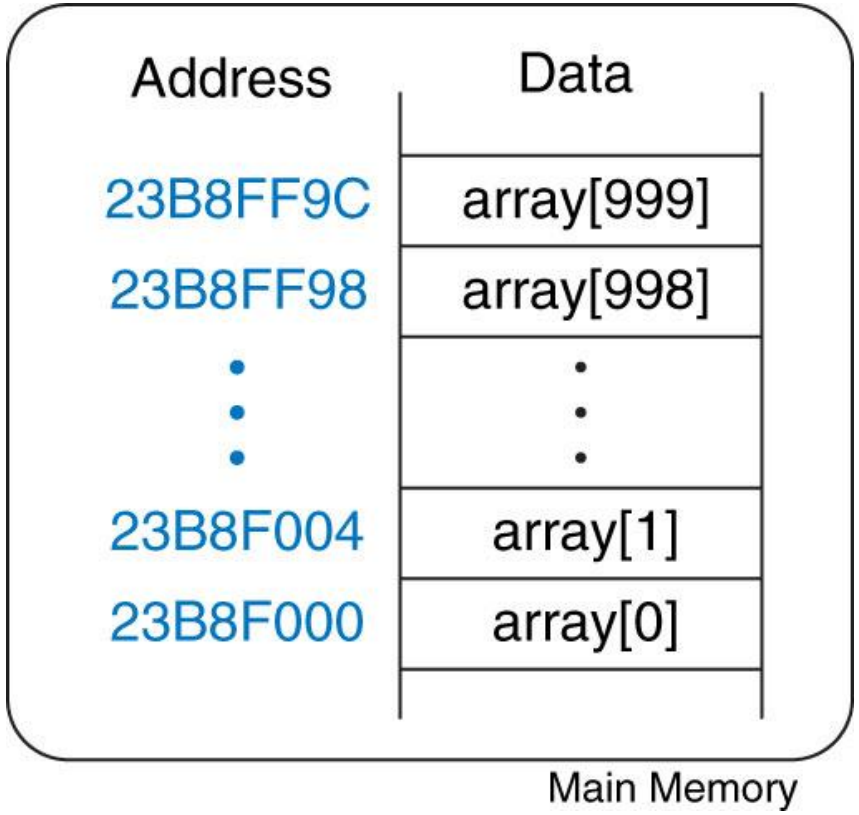


Figure 6.21 Memory holding array[1000] starting at base address 0x23B8F000

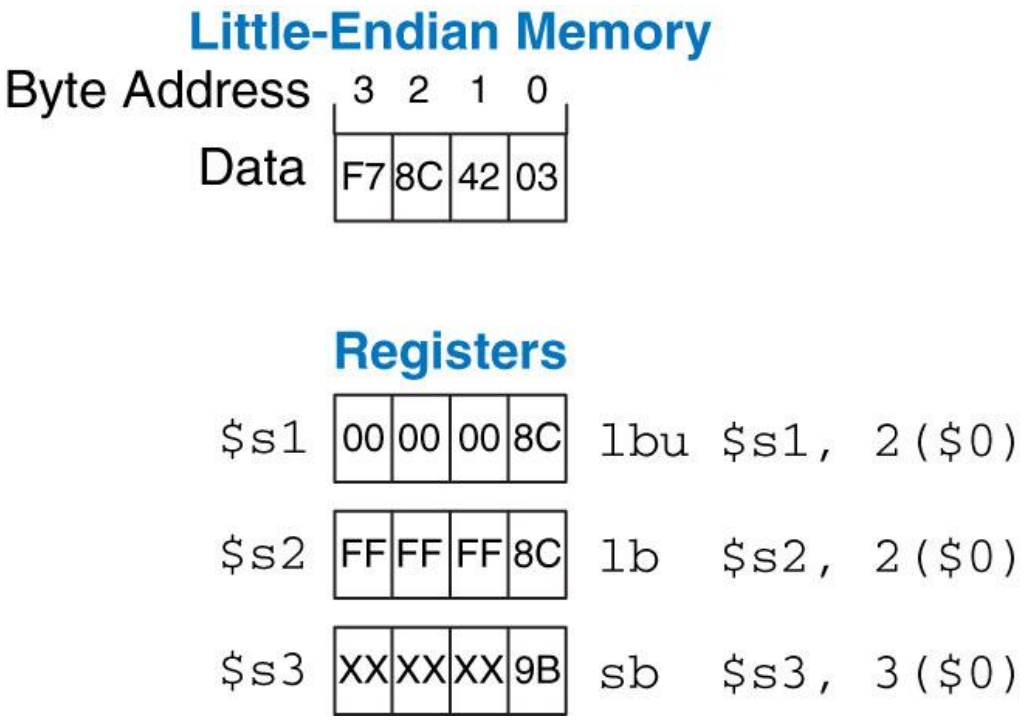


Figure 6.22 Instructions for loading and storing bytes

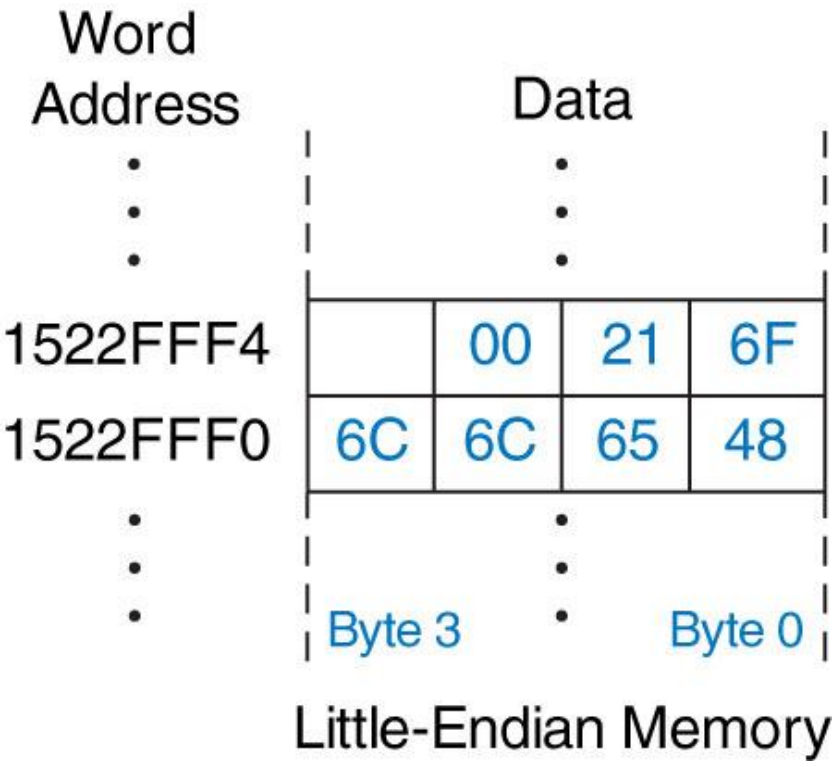


Figure 6.23 The string “Hello!” stored in memory

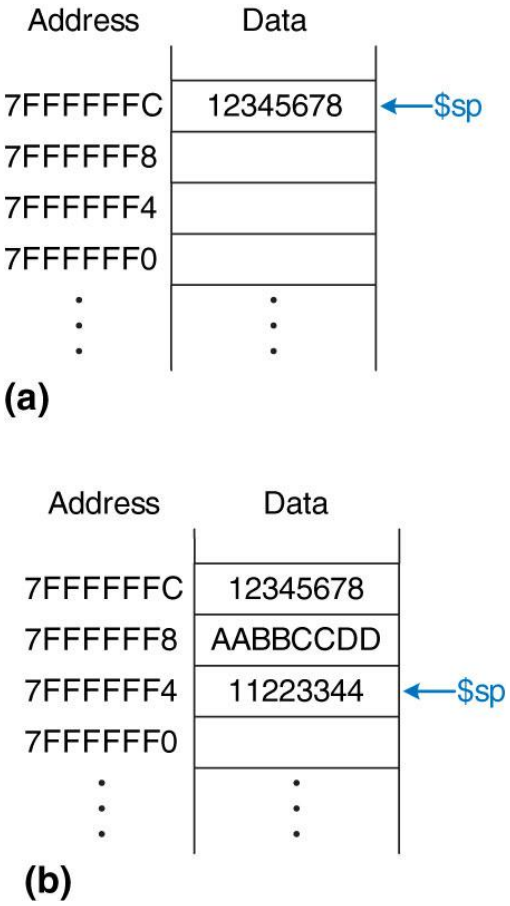


Figure 6.24 The stack

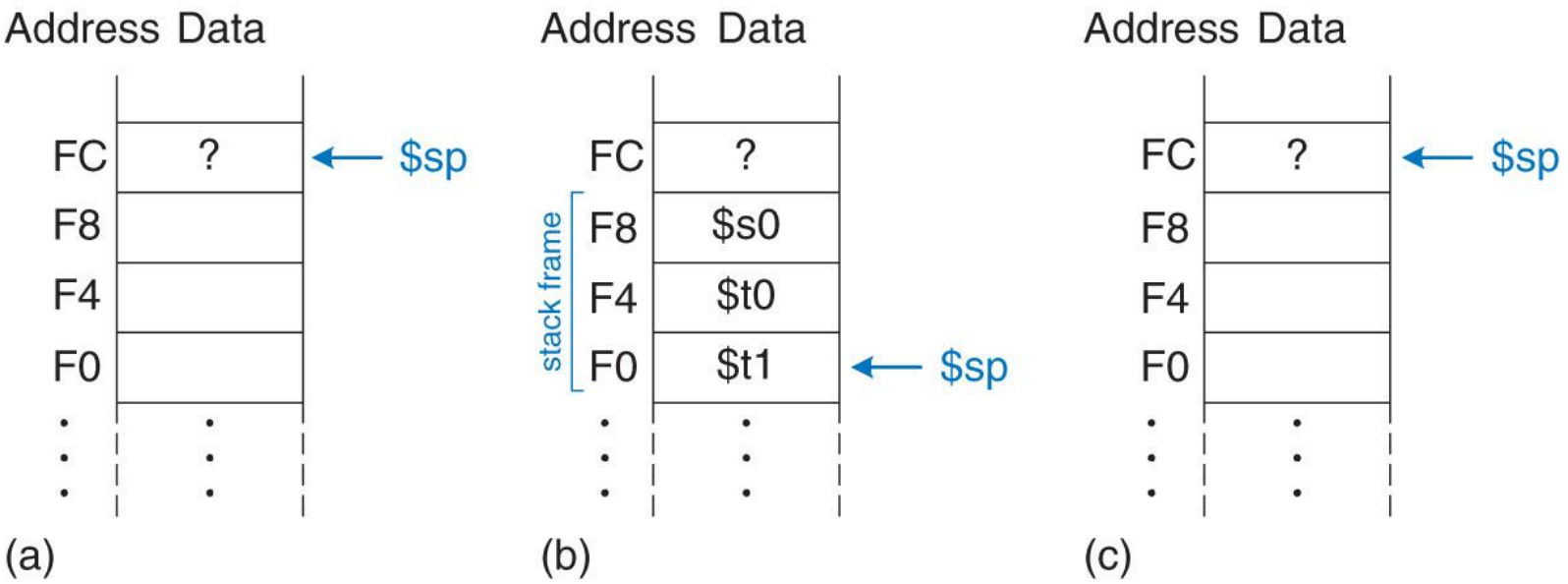


Figure 6.25 The stack (a) before, (b) during, and (c) after diffofsums function call

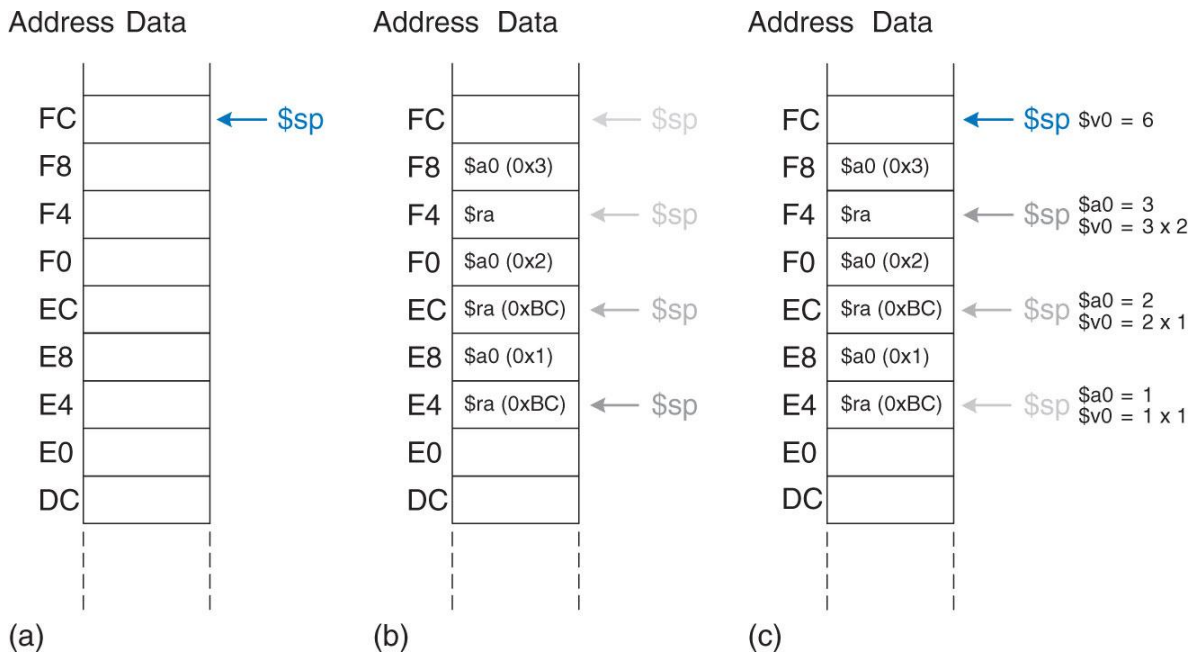


Figure 6.26 Stack during factorial function call when n = 3: (a) before call, (b) after last recursive call, (c) after return

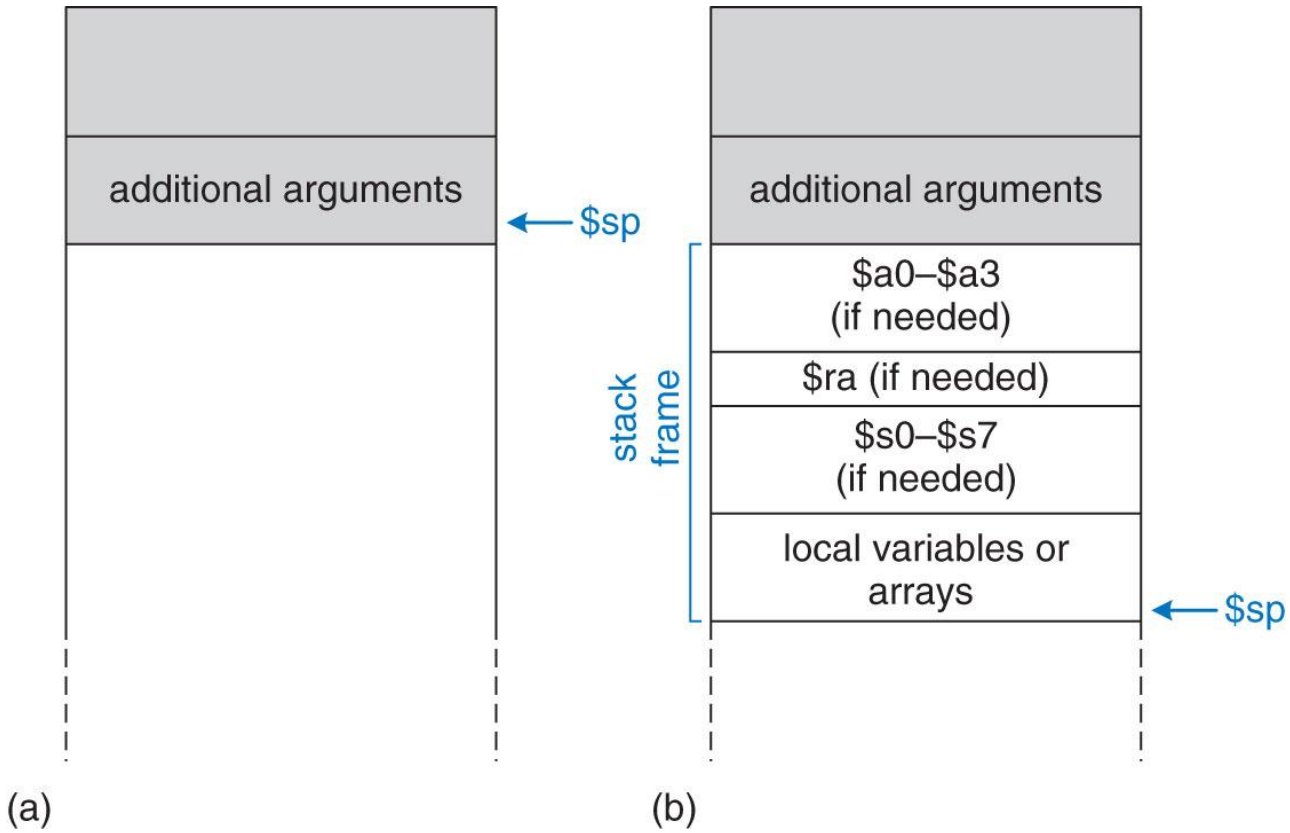


Figure 6.27 Stack usage: (a) before call, (b) after call

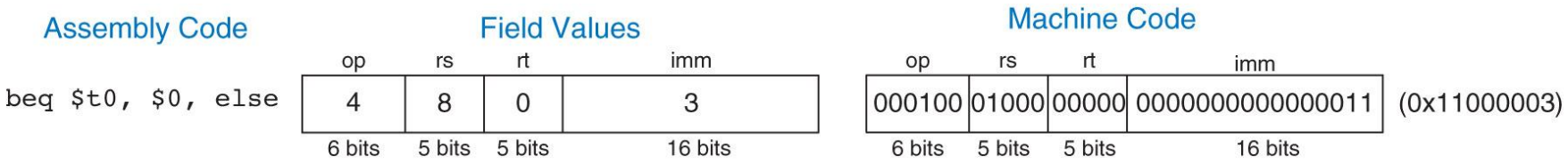


Figure 6.28 Machine code for beq

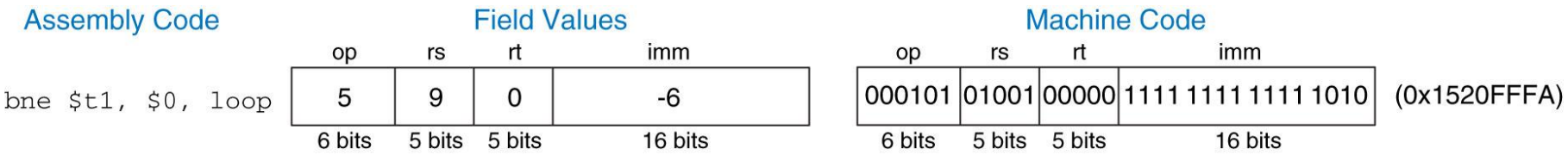


Figure 6.29 bne machine code

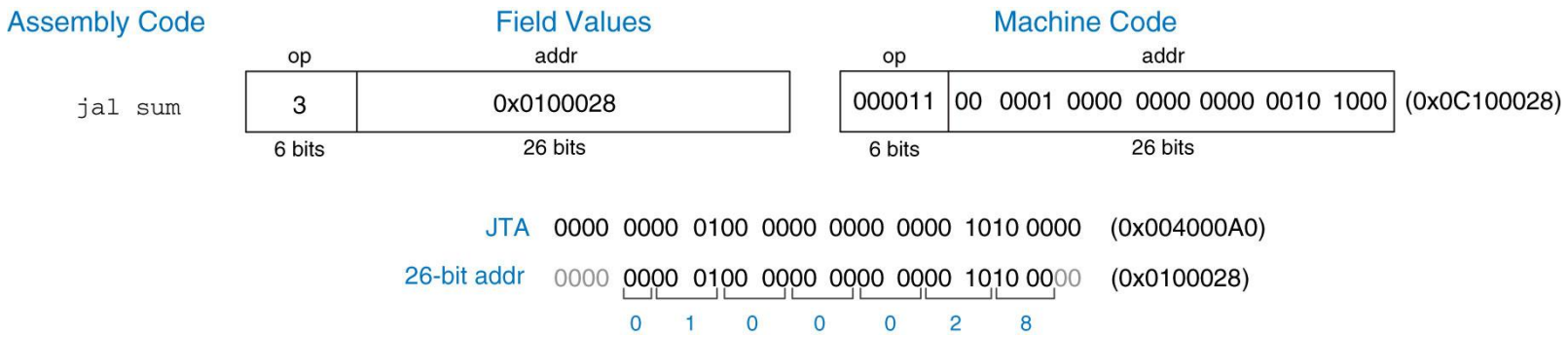


Figure 6.30 jal machine code

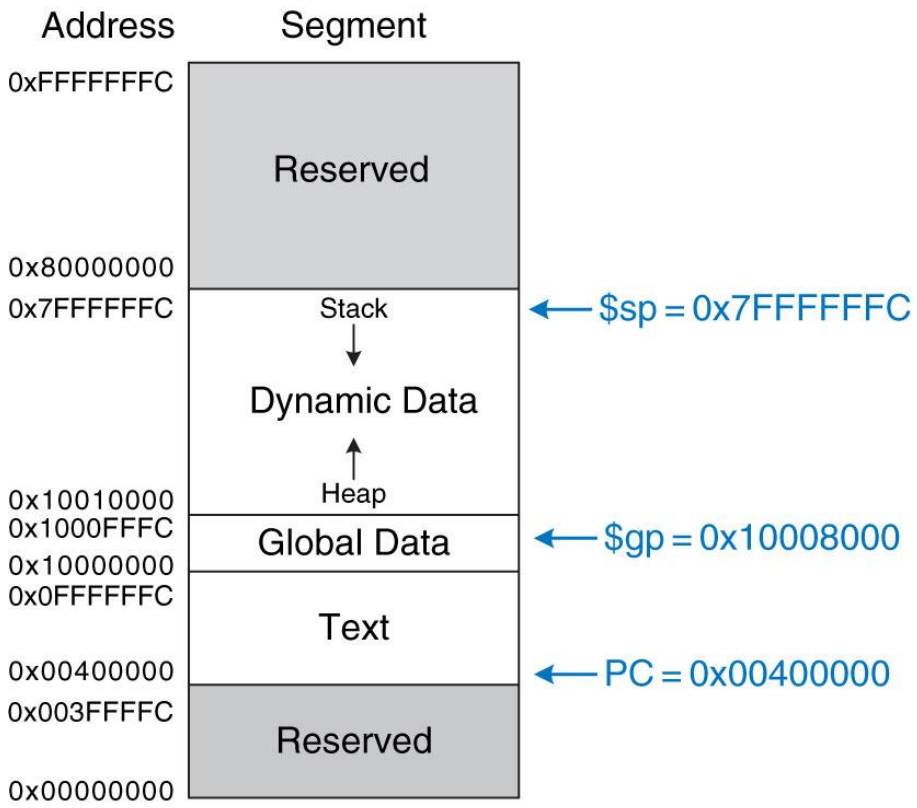


Figure 6.31 MIPS memory map

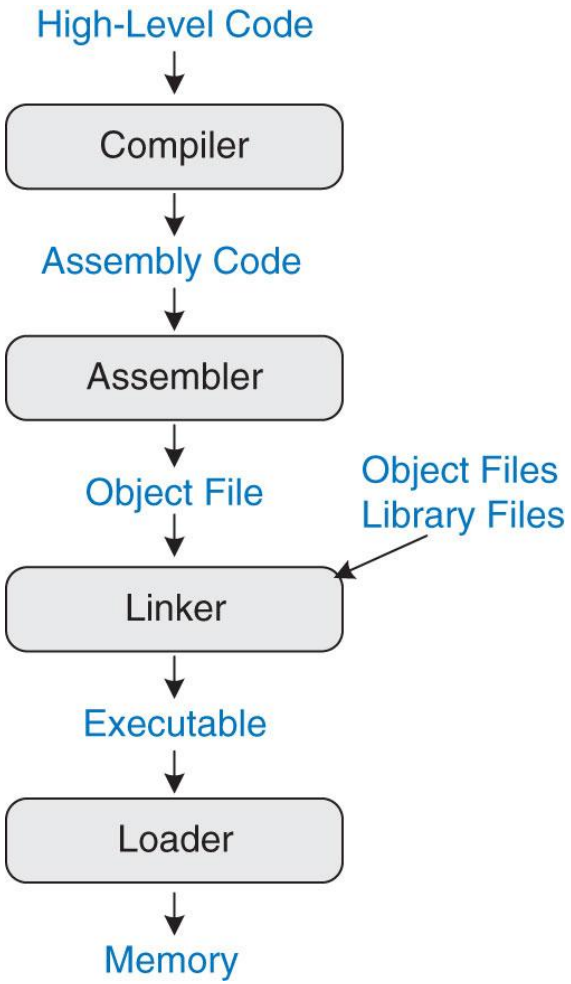


Figure 6.32 Steps for translating and starting a program

Executable file header	Text Size	Data Size
	0x34 (52 bytes)	0xC (12 bytes)
Text segment	**Address**	**Instruction**
	0x00400000	0x23BDFFFC
	0x00400004	0xAFBF0000
	0x00400008	0x20040002
	0x0040000C	0xAF848000
	0x00400010	0x20050003
	0x00400014	0xAF858004
	0x00400018	0x0C10000B
	0x0040001C	0xAF828008
	0x00400020	0x8FBF0000
	0x00400024	0x23BD0004
	0x00400028	0x03E00008
	0x0040002C	0x00851020
	0x00400030	0x03E00008
Data segment	**Address**	**Data**
	0x10000000	f
	0x10000004	g
	0x10000008	y

```
addi $sp, $sp, –4
sw   $ra, 0($sp)
addi $a0, $0, 2
sw   $a0, 0x8000($gp)
addi $a1, $0, 3
sw   $a1, 0x8004($gp)
jal  0x0040002C
sw   $v0, 0x8008($gp)
lw   $ra, 0($sp)
addi $sp, $sp, –4
jr   $ra
add  $v0, $a0, $a1
jr   $ra
```

Figure 6.33 Executable

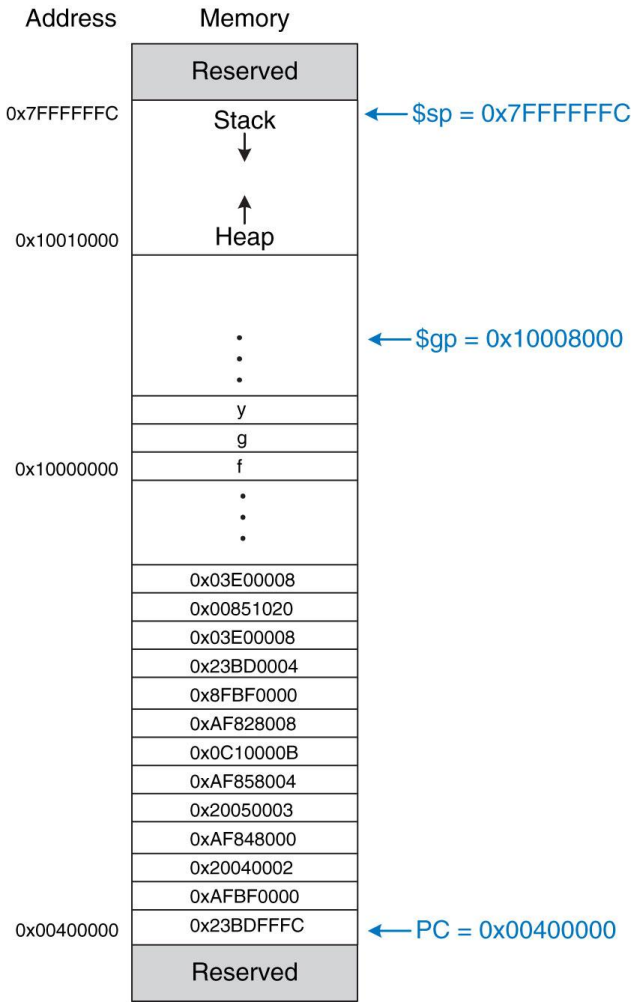


Figure 6.34 Executable loaded in memory

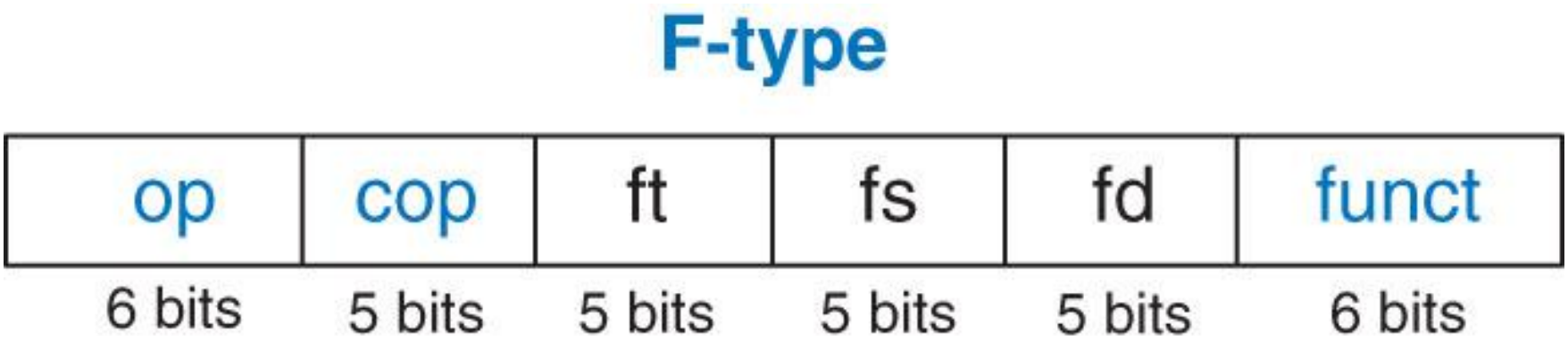


Figure 6.35 F-type machine instruction format

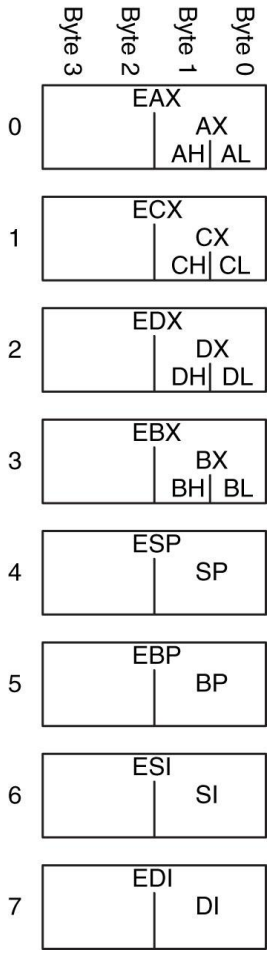


Figure 6.36 x86 registers

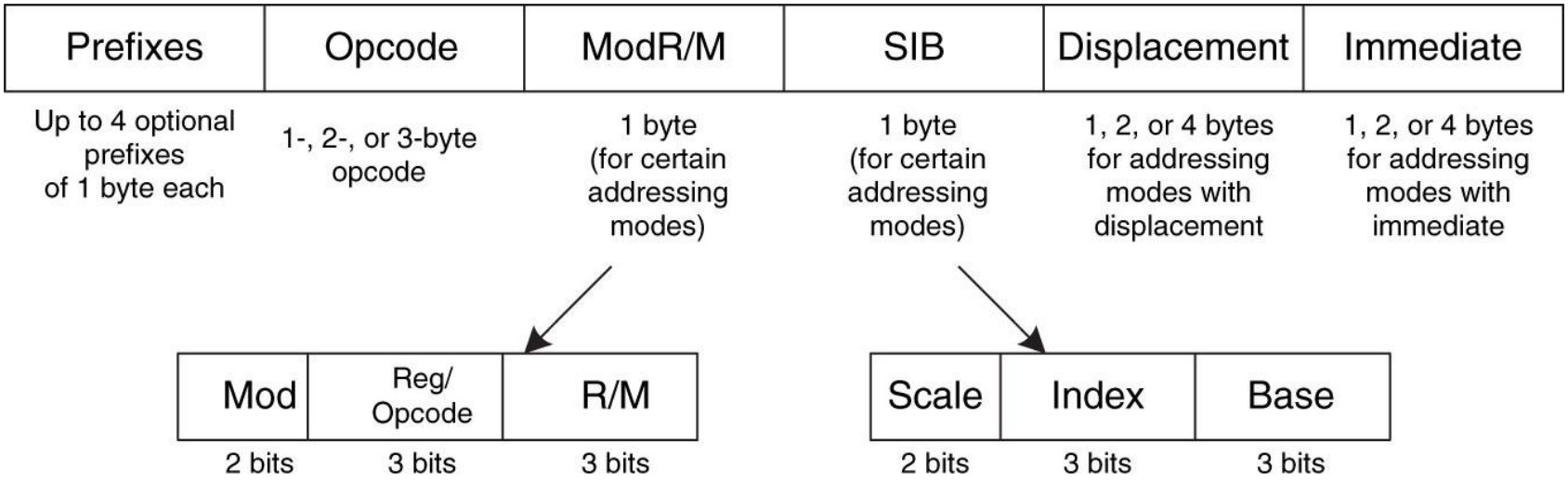


Figure 6.37 x86 instruction encodings

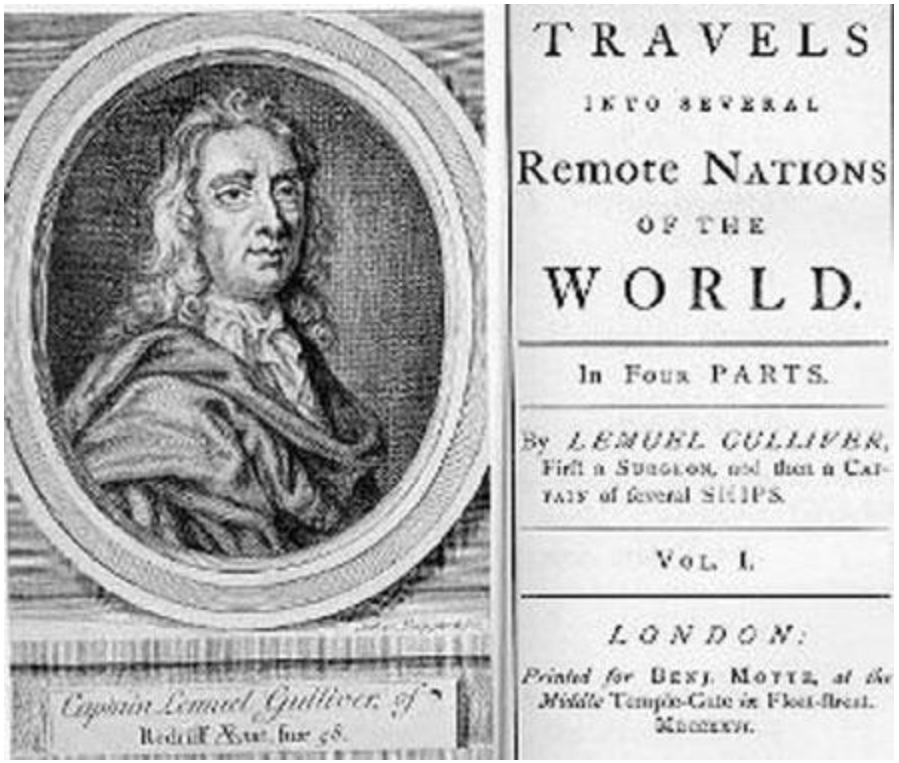
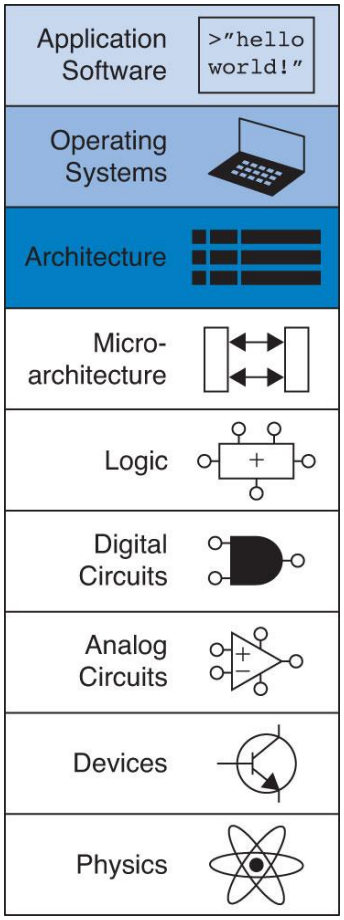


Figure M 01

Figure M 02

Figure M 03

Figure M 04



UNN Figure 1