63	31	15 7 0	
%rax	%eax	%ax %al	Return value
%rbx	%ebx	%bx %bl	Callee saved
%rcx	%есх	%cx %cl	4th argument
%rdx	%edx	%dx %dl	3rd argument
%rsi	%esi	%si %sil	2nd argument
%rdi	%edi	%di %dil	1st argument
%rbp \	%ebp	%bp %bpl	Callee saved
%rsp	%esp	%sp %spl	Stack pointer
%r8	%r8d	%r8w %r8b	5th argument
%r9	%r9d	%r9w %r9b	6th argument
%r10	%r10d	%r10w %r10b	Caller saved
%r11	%r11d	%r11w %r11b	Caller saved
%r12	%r12d	%r12w %r12b	Callee saved
%r13	%r13d	%r13w %r13b	Callee saved
%r14	%r14d	%r14w %r14b	Callee saved
%r15	%r15d	%r15w %r15b	Callee saved

Figure 3.2 Integer registers. The low-order portions of all 16 registers can be accessed as byte, word (16-bit), double word (32-bit), and quad word (64-bit) quantities.

arguments, returning values from functions, and storing local and temporary data. We will cover these conventions in our presentation, especially in Section 3.7, where we describe the implementation of procedures.

3.4.1 Operand Specifiers

Most instructions have one or more *operands* specifying the source values to use in performing an operation and the destination location into which to place the