第5章

计算机系统

5.1 概　　述

计算机本质上是实现了信息处理的自动化，虽然最早期的计算机主要是用来求解数学问题的，但是随着计算机科学技术的不断发展，计算机可以将人类的更多任务自动化。在此，以模拟抛硬币统计其中正背面次数为例，具体分析一下计算机是如何完成自动化计算任务的。

图 5-1 是通过 C 语言编程实现模拟 turns 次抛硬币的过程，这段代码是开发者根据 C 语言的语法规则编码出来的。当单击编译运行的按钮时，程序会弹出运行窗口，接收用户输入抛硬币的次数，然后模拟抛硬币的动作，计算正面的次数和背面的次数，除了在计算机询问用户需要抛几次时，用户需要输入次数之外，后面每一次抛硬币、计算和记录正背面的次数，都是由计算机自动完成的，如图 5-2 所示。

coins.cpp

```
#include<stdio.h>
#include<time.h>
#include<stdlib.h>
int main(){
    int turns; //抛硬币的次数
    int ftimes=0;//正面次数
    int btimes=0;//背面次数
    printf("请输入抛硬币的次数：");
    scanf("%d",&turns);
    srand(time(NULL));
    while(turns--){
        if(rand()%2==0){
            btimes++;
            printf("第%d次抛到背面硬币\n",turns);
        }
        else{
            ftimes++;
            printf("第%d次抛到正面硬币\n",turns);
        }
    }
    printf("一共抛到%d次正面硬币\n",ftimes);
    printf("一共抛到%d次背面硬币\n",btimes);
}
```

图 5-1　程序片段

在程序执行过程中，计算机是如何根据键盘敲击按键来获取抛掷硬币次数信息的？又是如何模拟每一次抛掷硬币过程，生成抛掷结果并展示给用户看的？最后又是如何完成统计计算的？作为用户的开发者，只是给了计算机一段代码，计算机只能理解、存储的数据形式是二进制，而对于近乎自然语言的程序代码，计算

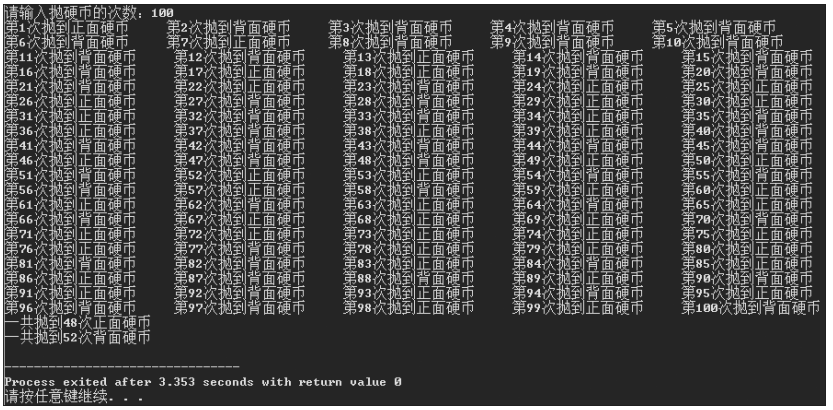

图 5-2　程序执行结果

机又是如何理解的？为了揭示程序代码的执行过程，打开集成开发环境 CPU 窗口，可以看到更为复杂的汇编代码，虽然可读性不好，但是语句更加细致地描述了计算机内语句执行的过程。汇编语言是较为接近机器码的编程语言，在 CPU 窗口（见图 5-3）中，可以读到 push、mov 等熟悉的单词。每条语句前有一串十六进制数字，代表了这条语句在计算机存储器存储的位置，后面尖括号内的数字代表了指令的相对存储位置，冒号后面的内容为指令具体内容，窗口右侧为 CPU 中特殊的存储器、寄存器的使用情况。可见，计算机执行一段代码是一个在复杂系统中综合协调的过程。因此，计算机系统是个复杂的、综合的、软硬件结合的系统。

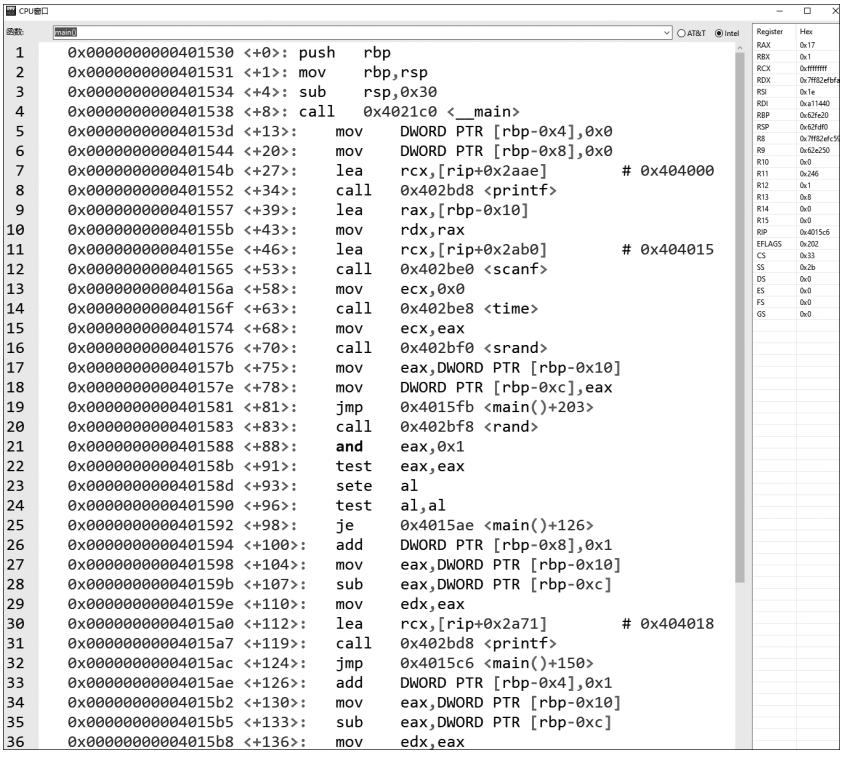

图 5-3　CPU 窗口内容

计算机可以识别的机器语言全部都是二进制代码，以下这段程序的可执行文件即为二进制代码，如图 5-4 所示。注：图 5-4 中截取的机器码仅为部分代码。

```
0000000  4D 5A 90 00 03 00 00 00 04 00 00 00 FF FF 00 00
0000010  B8 00 00 00 00 00 00 00 40 00 00 00 00 00 00 00
0000020  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0000030  00 00 00 00 00 00 00 00 00 00 00 00 80 00 00 00
0000040  0E 1F BA 0E 00 B4 09 CD 21 B8 01 4C CD 21 54 68
0000050  69 73 20 70 72 6F 67 72 61 6D 20 63 61 6E 6E 6F
0000060  74 20 62 65 20 72 75 6E 20 69 6E 20 44 4F 53 20
0000070  6D 6F 64 65 2E 0D 0D 0A 24 00 00 00 00 00 00 00
0000080  50 45 00 00 64 86 12 00 F4 50 25 65 00 EE 01 00
0000090  F1 05 00 00 F0 00 27 00 0B 02 02 18 00 1E 00 00
00000A0  00 20 00 00 00 0C 00 00 00 15 00 00 00 10 00 00
00000B0  00 00 40 00 00 00 00 00 00 10 00 00 00 02 00 00
00000C0  04 00 00 00 00 00 00 00 05 00 02 00 00 00 00 00
00000D0  00 A0 02 00 00 06 00 00 CE B2 02 00 03 00 00 00
00000E0  00 00 20 00 00 00 00 00 00 10 00 00 00 00 00 00
00000F0  00 00 10 00 00 00 00 00 00 10 00 00 00 00 00 00
0000100  00 00 00 00 10 00 00 00 00 00 00 00 00 00 00 00
0000110  00 80 00 00 58 08 00 00 00 00 00 00 00 00 00 00
0000120  00 50 00 00 34 02 00 00 00 00 00 00 00 00 00 00
0000130  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0000140  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0000150  20 A0 00 00 28 00 00 00 00 00 00 00 00 00 00 00
0000160  00 00 00 00 00 00 00 00 14 82 00 00 D8 01 00 00
0000170  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0000180  00 00 00 00 00 00 00 00 2E 74 65 78 74 00 00 00
0000190  60 1D 00 00 00 10 00 00 00 1E 00 00 00 06 00 00
00001A0  00 00 00 00 00 00 00 00 00 00 00 00 20 00 50 60
00001B0  2E 64 61 74 61 00 00 00 90 00 00 00 00 30 00 00
00001C0  00 02 00 00 00 24 00 00 00 00 00 00 00 00 00 00
00001D0  00 00 00 00 40 00 50 C0 2E 72 64 61 74 61 00 00
00001E0  40 08 00 00 00 40 00 00 00 0A 00 00 00 26 00 00
00001F0  00 00 00 00 00 00 00 00 00 00 00 00 40 00 50 40
0000200  2E 70 64 61 74 61 00 00 00 34 02 00 00 00 50 00 00
0000210  00 04 00 00 00 30 00 00 00 00 00 00 00 00 00 00
```

图 5-4　部分机器码

计算机系统由计算机硬件和软件两部分组成，是计算机硬件系统和计算机软件系统的有机结合整体。计算机硬件部分包含 CPU、内存储器、外部输入/输出(I/O)设备等；计算机软件部分包含基础软件和应用软件，及相应的支持文档。计算机系统的软件部分和硬件部分是相辅相成、互为依托、缺一不可的有机整体。如果没有人类的智慧(软件)，只有一堆计算机硬件，也不过是一堆废件。同样，如果只有软件，没有硬件，计算机系统也不过是空中楼阁，想象中的系统。

5.1.1　计算机的基本组成

以 5.1 节的抛硬币事件为例，计算机想要自动化完成抛掷 n 次硬币的过程，首先需要询问用户要抛的次数，即 n 的值，这个交互过程需要有设备可以输入 n 的值给计算机，所以计算机需要一个输入设备来完成用户设定次数的输入操作，如键盘。然后，当用户输入抛掷次数 n 后，计算机通过模拟和运算产生了结果，这个结果是需要反馈给用户为用户所见的，所以计算机需要一个输出设备展示其操作的结果，如显示器。至此可以分析出计算机的组成至少包括输入设备和输出设备(见图 5-5)，输入和输出设备可以很好地完成计算机与用户的交互功能。

计算机可以根据用户的明确指令即程序，自动执行用户编写的程序内容，以完成用户的具体需求。以抛硬币为例，图 5-1 是用户编写的模拟抛掷硬币并完成结果正背面次数统计计算的程序，计算机需要对这段程序进行理解，而这个理解的过程就是对程序代码进行编译

或解释的过程，也就是说把人类理解的程序语言，变成计算机可以执行的机器语言，计算机才可以执行，即改为图 5-4 中计算机可以执行的二进制代码。如果想了解机器具体怎样执行程序语句，可以对图 5-6 中的汇编语句进行具体的分析。

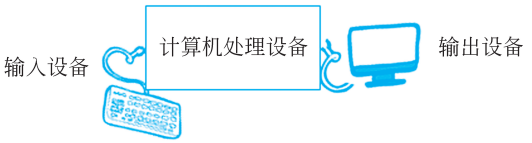

图 5-5　计算机基本组成

```
0x0000000000401530 <+0>:
0x0000000000401531 <+1>:
0x0000000000401534 <+4>:
0x0000000000401538 <+8>:
0x000000000040153d <+13>:
0x0000000000401544 <+20>:
0x000000000040154b <+27>:
0x0000000000401552 <+34>:
0x0000000000401557 <+39>:
```

图 5-6　汇编语句部分内容

左边十六进制数字和括号内数字之间的关系参见式(5-1)。

$$0x000000000040153\text{hex}(x)=0x0000000000401530+<+x> \tag{5-1}$$

hex(x)：将 x 值转换为十六进制的数，如 hex(13)=d。

结合图 5-7 继续分析。

0x0000000000401530	push　rbp
0x0000000000401531	mov　rbp,rsp
0x0000000000401532	
0x0000000000401533	
0x0000000000401534	sub　rsp,0x30
0x0000000000401535	
0x0000000000401536	
0x0000000000401537	
0x0000000000401538	call　0x4021c0　<__main>
0x0000000000401539	
0x000000000040153a	
0x000000000040153b	
0x000000000040153c	

图 5-7　语句的存储情况分析

可以看到汇编语句是存放在连续的存储空间中的，这个直接存储可执行语句的空间叫做内存，也就是说计算机执行的指令不是凭空出现的，而是实际存在于一个空间中的。计算机就是依次读取这个空间中的内容来进行具体操作的，所以计算机要有存储装置用来完成指令集和数据的存储操作。当然随着计算机处理问题规模的不断扩大，涉及的指令和数据不断增加，需要更大空间的存储器以完成这些内容的存储操作。以人脑来类比或许更加容易理解计算机的存储器，即人脑本身可以记忆很多事情，但是由于需要学习和记忆的东西非常多，仅靠大脑已经无法容纳更多的知识和事实了，因此人们进一步把需要记忆的内容记录在笔记本上，当需要时，可以通过翻阅笔记本来帮助回忆相关的内容。所以，人的知识是存储在自己的大脑和外部的笔记本等可记录知识的实体中的。同理，计算机同样需要两种类

型的存储器，短期快速但是易丢失的和长期较慢但是不易丢失的存储器，也就是内存和外存。

内存和外存的存在，使计算机完成了指令和数据的存储，即记住了这些指令，但是仅凭记忆是无法完成指令蕴含的实际操作的。需要进一步的具体执行，即理解这些指令的内涵以完成对应的操作，并达到指令的预期目标——自动化地完成任务。同样类比人脑，人仅记住知识和事实也是不够的，还要理解并能够在适合的场合对知识加以应用，即人脑要根据学习到的知识处理接收到的信息，做出合理正确的决策或响应。与之对应，计算机内也有一个类似于人脑理解和处理信息的机构，可以根据指令完成对相关数据的操作，从而得到预期的结果。那这个具有理解和处理信息的机构就是计算机的大脑——运算器和控制器，它们可以完成具体的运算和控制功能。

根据上述分析，计算机的组成呼之欲出。时至今日，计算机的组成结构依然是 1946 年美国科学家冯·诺依曼提出的以存储程序为核心而设计的计算机体系结构。这种体系结构，在不改变机器结构的前提下，通过编写不同的指令，就可以完成多种不同的功能。冯·诺依曼提出的解决思路是将各类基础的运算，如算术运算——加减乘除、逻辑运算——与或非，以及一些其他的简单操作，如跳转、比较、输入、输出等，都转化为指令。众多指令便构成了计算机中的一组基础指令集合。计算机要实现的各类功能，都可以由一条条简单的指令组合而成。这些指令一条一条按照顺序执行，形成一组指令序列。机器结构是不变的，变化的只是指令序列，计算机就是根据不同的指令序列，实现各种不同的功能。

图 5-8 描绘了冯·诺依曼计算机的体系结构，包括 5 个部分：存储器、运算器、控制器、输入设备和输出设备。

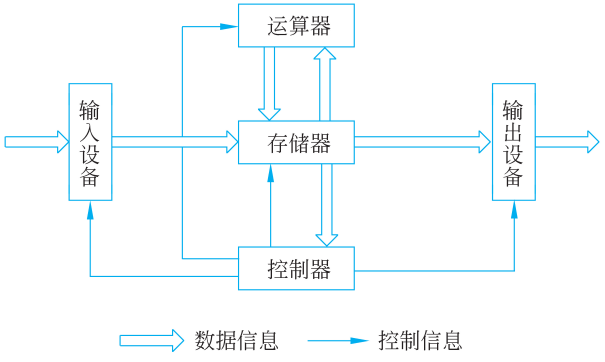

图 5-8　冯·诺依曼计算机

存储器用于存放计算机要处理的信息。这些信息可能原本就保存在计算机系统中，也可能是运行程序时由用户提供的。这类外部的输入信息，例如，用户通过鼠标单击屏幕的某个位置、通过键盘输入一些文字，就需要通过输入设备来接收。计算机得到信息后，通过运算器来实现相应的运算操作，如逻辑运算和算术运算。控制器用于指挥计算机各个部件之间的协同工作。在现代计算机硬件组织结构中，控制器和运算器是最为重要的两部分，合称为中央处理器(CPU)。当计算机功能运行完毕或者某一任务执行完成之后，由输出设备输出计算机的处理结果，例如将运行结果显示在显示器上、利用打印机打印文档等。这五部分是计算机的基本硬件，也是支撑现代计算机的骨架。除了这五部分之外，图 5-8 中也显示了

数据信息和控制信息的流向。例如，数据由用户输入，先被送入存储器保存起来，之后被送入运算器并执行相应的程序，完成后由输出设备显示结果数据。而控制信息，则全部由控制器发出，传送到各个其他部件，通知各个部件在何时需要做何种操作。

现代计算机硬件组织结构与冯·诺依曼计算机的体系结构基本一致，在保留基本部件的基础上，布局上有一些差异。图 5-9 展示了典型的计算机硬件组织结构。其中，冯·诺依曼计算机中的控制器和运算器被放置在现代计算机中的 CPU 内。并且随着现代工艺的发展，各种器件规模体积逐渐缩小，运算器浓缩为了算术逻辑单元，控制器变成了控制单元。冯·诺依曼计算机中的存储器对应现代计算机硬件中的内存。除内存外，现代计算机中很多其他部件也具备存储数据信息的功能，如 CPU 内的寄存器组、外部存储器等。输入设备和输出设备在两种系统中一一对应。各类总线，用于传输控制器下达给各部件的命令、提供各类数据在各部件之间的传输通道，对应于冯·诺依曼计算机体系中的数据信息和控制信息的流向。

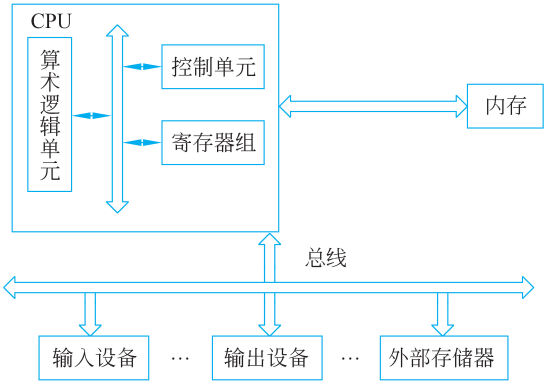

图 5-9　典型计算机硬件组织结构

冯·诺依曼计算机体系结构具备三点特征。

（1）二进制表示。数据和程序均以二进制的形式存储在计算机内。

（2）程序与数据一样存放在存储器中。在冯·诺依曼计算机体系结构形成之前，数据和程序是分别存放和处理的，数据存储于内存中，程序视为控制器内的部件。在冯·诺依曼计算机体系中，程序和数据以同样的形式存储于内存。

（3）指令逐条自动执行。在控制器的控制之下，实现“取指令、执行该条指令、自动取下一条指令”这一系列操作。

上文讨论的计算机体系结构都是单机体系结构。除单机体系结构外，为了提高计算机的运算性能，还可以由多个计算机构成并行处理结构、集群结构等。

5.1.2　计算机的工作原理

计算机处理各类任务、实现各种功能，通过执行程序来完成。程序的运行离不开 CPU 中的指令。指令是 CPU 执行的最小单位，CPU 的工作过程就是执行各类指令。

指令由操作码和操作数两部分构成。操作码表明该条指令要执行什么动作，如是执行加法运算还是执行输入操作，或是将数据转移到内存某个位置。操作数则用来指明该动作

的作用对象，如被转移的数据、参与算术运算的数据等。指令的长度可以是固定的也可以是可变的。指令的长度通常同 CPU 一次存取的数据长度相关。CPU 通过数据总线一次可以存取、传送的数据，称为一个字。一个字通常包括一个或多个字节。字的长度，称为字长。指令的长度是一个或多个字长。例如，没有操作数的指令，其长度为一个字节；只涉及寄存器的指令，其长度为两个字节。

计算机能够识别的指令，是仅有 0 和 1 构成的二进制串，称为机器指令，又叫机器码，图 5-4即为程序段编译后的机器码。二进制指令符合机器的特点，但不符合人类的读写习惯，所以直接用二进制的机器指令编写程序非常枯燥，且很难书写和阅读。因此，在指令中引入助记符帮助人们理解和使用，这就是汇编指令，即图 5-3CPU 窗口中的内容。汇编指令中的助记符很容易理解，对于操作码而言，一般采用该操作动作的英文缩写，例如，加法操作的助记符是 ADD、比较操作的助记符是 CMP 等；对于操作数而言，具体的数值直接用阿拉伯数字表示，寄存器通常用英文字母 R 加上该寄存器的编号表示。图 5-10 表示了加法指令 ADD R1、R5、R1。该条指令将寄存器 R1 中保存的数值加上寄存器 R5 中的数值，再将结果保存回寄存器 R1。

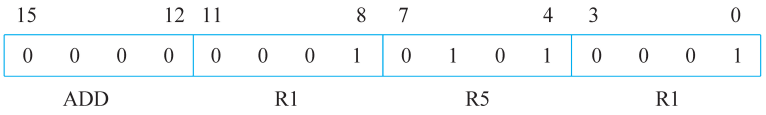

图 5-10　加法指令示例

该加法指令长度为 16 位。最左边四位是操作码 0000，代表加法操作 ADD；该指令后 12 位是操作数。因为加法指令涉及三个操作数——两个加数及求和结果，操作数部分被划分为三段，分别代表第一个加数寄存器 R1、第二个加数寄存器 R5 和结果寄存器 R1。二进制数 0001 代表寄存器 R1，0101 代表寄存器 R5。图 5-11 中最下面一行是该指令的汇编形式，计算机无法直接执行汇编指令，需要将其转换成上一行的机器指令才可以执行。

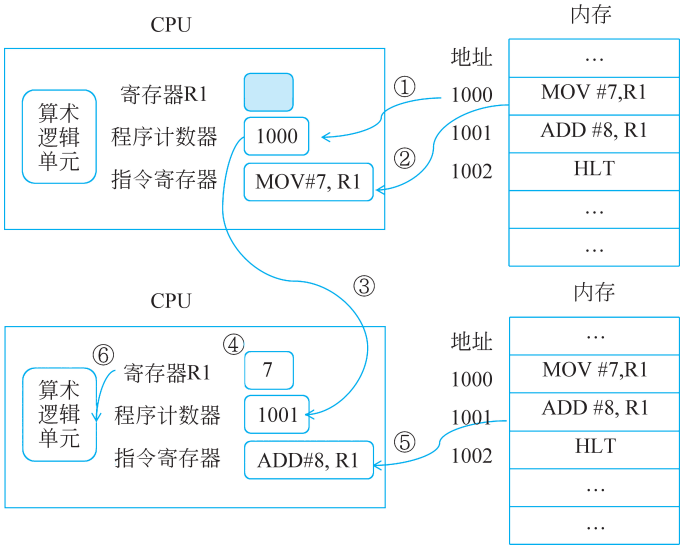

图 5-11　CPU 工作过程示例

计算机的工作依靠指令的执行。CPU 中的各个部件协同工作，共同完成指令的执行。CPU 一般包括 3 个主要部分：算术逻辑单元(arithmetic and logic unit，ALU)、控制单元(control unit，CU)和寄存器组(registers)。算术逻辑单元接收参与算术运算或逻辑运算的操作数，并根据指令的要求，执行相应的运算。控制单元既要分析指令所做的操作，又要传送指令内容和操作数，同时还要负责发送控制信号，协调 CPU 其他部件共同工作。寄存器组，顾名思义，是一组用于保存临时数据的结构，包括通用寄存器和专用寄存器。通用寄存器一般保存的是参加运算的操作数及运算结果，而专用寄存器则保存计算机当前工作相关的信息，专用寄存器中最为重要的有指令寄存器和程序计数器。指令寄存器(instruction register，IR)存放从内存中读取的指令。程序计数器(program counter，PC)保存下一条将被执行的指令所在的内存地址。

下面以简单的加法计算“7＋8”为例，详细描述 CPU 完成这一运算的整个工作过程。如图 5-11 所示，该加法运算程序包含 3 条汇编指令：先将第一个加数 7 送至寄存器 R1，之后执行加法计算，将结果保存至寄存器 R1 中，最后运算结束，停止操作。这 3 条汇编指令顺序保存在内存中，地址从 1000 至 1002。

开始执行这段代码时，先将这段代码中的第一条指令的地址取出，保存在程序计数器内，即程序计数器内的值设为 1000。接下来，根据程序计数器内保存的内存地址，找到内存中地址 1000 的内存单元保存的内容“MOV ＃7，R1”，把该指令放入指令寄存器。同时，程序计数器指向下一条要被执行的指令所在的内存地址，即增 1 变为 1001。MOV 指令经过译码，提取出指令中的操作码、操作数等信息，在控制单元的控制下，将寄存器 R1 置为 7，该条指令执行结束。接下来，继续根据程序计数器的值寻找下一条要执行的指令。程序计数器的值为 1001，代表要执行的指令保存在内存地址为 1001 的内存单元。重复上次执行的操作，将新指令“ADD ＃8，R1”取出放入指令寄存器。执行 ADD 指令时，操作码被译码为一系列的控制码，调动算术逻辑单元执行加法运算、数据传输等操作。这种指令逐条执行的过程将会持续下去，直到遇到 HLT 指令，操作停止。当遇到控制指令时，程序的执行不再按顺序。程序计数器的值将会发生改变，而不再是自动增加 1，从而实现跳转到其他位置执行新的指令。

5.2　计算机硬件子系统

计算机系统的硬件部分是指计算机的物理实体，包括 CPU、内存储器及必要的 I/O 设备，输入设备如键盘、鼠标、手写板等，输出设备如硬盘、显示器、打印机等。这些硬件承载着计算机的物理实体，是计算机系统能够运行的实体保证。

5.2.1　中央处理器

中央处理器(central processing unit，CPU)作为计算机系统的运算和控制核心，是信息处理、程序运行的最终执行单元。

CPU 出现于大规模集成电路时代，处理器架构设计的迭代更新及集成电路工艺的不断提升促使其不断发展完善。从最初专用于数学计算到广泛应用于通用计算，从 4 位到 8 位、16 位、32 位处理器，最后到 64 位处理器，从各厂商互不兼容到不同指令集架构规范的出现，

CPU 自诞生以来一直在逻辑结构、运行效率及功能外延上飞速发展。

CPU 已经有 40 多年的历史，通常将其分成六个阶段，如表 5-1 所示。

表 5-1 CPU 发展的六个阶段

阶 段	处理器位数	代 表 产 品	关 键 事 件
第一阶段（1971—1973 年）	4 位和 8 位	Intel 4004	1971 年 CPU 的诞生；1978 年 8086 处理器，奠定了 X86 指令集架构
第二阶段（1974—1977 年）	8 位	Intel 8080	指令系统较完善
第三阶段（1978—1984 年）	16 位	Intel 8086	指令系统较成熟
第四阶段（1985—1992 年）	32 位	Intel 80386	1989 年 80486 处理器实现了 5 级标量流水线，CPU 的初步成熟，传统处理器发展阶段的结束
第五阶段（1993—2005 年）	奔腾系列微处理器	Pentium 处理器	1995 年 11 月 Pentium 处理器首次采用超标量指令流水结构，引入了指令的乱序执行和分支预测技术，大大提高了处理器的性能
第六阶段（2005—2021 年）	多核心、更高并行度	Intel 酷睿系列处理器和 AMD 的锐龙系列处理器	—

为了满足操作系统的上层工作需求，现代处理器进一步引入了诸如并行化、多核化、虚拟化及远程管理系统等功能，不断推动着上层信息系统向前发展。

CPU 的工作通过执行指令来完成。执行指令的步骤顺序称为指令周期。CPU 的工作分为以下五个阶段：取指令阶段、指令译码阶段、执行指令阶段、访存取数和结果写回。对于 CPU 而言，影响其性能的指标主要有主频、CPU 的位数、CPU 的缓存指令集、CPU 核心数和每周期指令数(instruction per clock，IPC)。所谓 CPU 的主频，指的就是时钟频率，它决定了 CPU 的性能，可以通过超频来提高 CPU 的主频以获得更高性能。而 CPU 的位数指的就是处理器能够一次性计算的浮点数的位数。通常情况下，CPU 的位数越高，CPU 进行运算时的速度就会变得越快。21 世纪 20 年代后，个人计算机使用的 CPU 一般为 64 位，这是因为 64 位处理器可以处理范围更大的数据并原生支持更高的内存寻址容量，提高了人们的工作效率。而 CPU 的缓存指令集是存储在 CPU 内部的，主要指的是能够对 CPU 的运算进行指导以及优化的硬程序。一般来讲，CPU 的缓存可以分为一级缓存、二级缓存和三级缓存，缓存性能直接影响 CPU 处理性能。部分特殊职能的 CPU 可能会配备四级缓存。

Intel 和美国超威半导体公司 AMD 是国外研发 CPU 芯片的两家知名公司。根据 Intel 产品线规划，截止到 2021 年 Intel 十一代消费级酷睿有五类产品：i9/i7/i5/i3/奔腾/赛扬。此外还有面向服务器的至强铂金/金牌/银牌/铜牌和面向 HEDT 平台的至强 W 系列。根据 AMD 产品线规划，截止到 2021 年 AMD 锐龙 5000 系列处理器有 ryzen9/ryzen7/ryzen5/ryzen3 四个消费级产品线，此外还有面向服务器市场的第三代霄龙 EPYC 处理器和面向 HEDT 平台的线程撕裂者系列。

国内现有的几种主流 CPU 芯片如下所示。

1. 龙芯系列

龙芯系列芯片是由中国科学院中科技术有限公司设计研制的，采用 MIPS 体系结构，具有自主知识产权，产品现包括龙芯 1 号小 CPU、龙芯 2 号中 CPU 和龙芯 3 号大 CPU 三个系列，此外还包括龙芯 7A1000 桥片。

龙芯 1 号系列 32/64 位处理器专为嵌入式领域设计，主要应用于云终端、工业控制、数据采集、手持终端、网络安全、消费电子等领域，具有低功耗、高集成度及高性价比等特点。其中龙芯 1A 32 位处理器和龙芯 1C 64 位处理器稳定工作在 266～300MHz，龙芯 1B 处理器是一款轻量级 32 位芯片。龙芯 1D 处理器是超声波热表、水表和气表的专用芯片。2015 年，新一代北斗导航卫星搭载着我国自主研制的龙芯 1E 和 1F 芯片，这两颗芯片主要用于完成星间链路的数据处理任务。

龙芯 2 号系列是面向桌面和高端嵌入式应用的 64 位高性能低功耗处理器。龙芯 2 号产品包括龙芯 2E、2F、2H 和 2K1000 等芯片。龙芯 2E 首次实现对外生产和销售授权。龙芯 2F 平均性能比龙芯 2E 高 20％以上，可用于个人计算机、行业终端、工业控制、数据采集、网络安全等领域。龙芯 2H 于 2012 年推出正式产品，适用于计算机、云终端、网络设备、消费类电子等领域需求，同时可作为 HT 或者 PCI-e 接口的全功能套片使用。2018 年，龙芯推出龙芯 2K1000 处理器，它主要是面向网络安全领域及移动智能领域的双核处理芯片，主频可达 1GHz，可满足工业物联网快速发展、自主可控工业安全体系的需求。

龙芯 3 号系列是面向高性能计算机、服务器和高端桌面应用的多核处理器，具有高带宽、高性能、低功耗的特点。龙芯 3A3000/3B3000 处理器采用自主微结构设计，主频可达到 1.5GHz 以上。2019 年面向市场的龙芯 3A4000 为龙芯第三代产品的首款四核芯片，该芯片基于 28nm 工艺，采用新研发的 GS464V 64 位高性能处理器核架构，并实现 256 位向量指令，同时优化片内互连和访存通路，集成 64 位 DDR3/4 内存控制器，集成片内安全机制，主频和性能再次得到大幅提升。

龙芯 7A1000 桥片是龙芯的第一款专用桥片组产品，目标是替代 AMD RS780＋SB710 桥片组，为龙芯处理器提供南北桥功能。它于 2018 年 2 月份发布，目前搭配龙芯 3A3000 及紫光 4GB DDR3 内存应用在一款高性能网络平台上。该方案整体性能相较于 3A3000＋780e 平台有较大提升，具有高国产率、高性能、高可靠性等特点。

2. 上海兆芯

上海兆芯集成电路有限公司是成立于 2013 年的国资控股公司，其生产的处理器采用 x86 架构，产品主要有开先 ZX-A、ZX-c/ZX-C＋、ZX-D，开先 KX-5000 和 KX-6000，开胜 ZX-C＋、ZX-D、KH-20000 等。

开先 KX-5000 系列处理器采用 28nm 工艺，提供 4 核或 8 核两种版本，整体性能较上一代产品提升高达 140％，达到国际主流通用处理器性能水准，能够全面满足党政桌面办公应用，以及包括 4K 超高清视频观影等多种娱乐应用需求。

开先 KX-6000 系列处理器主频高达 3.0 GHz，兼容全系列 Windows 操作系统及中科方德、中标麒麟、普华等国产自主可控操作系统，性能与 Intel 第七代的酷睿 i5 相当。开胜 KH-20000 系列处理器是兆芯面向服务器等设备推出的 CPU 产品。

3. 上海申威

申威处理器简称 SW 处理器，出自 DEC 的 Alpha 21164。采用 DEC Alpha 架构，具有