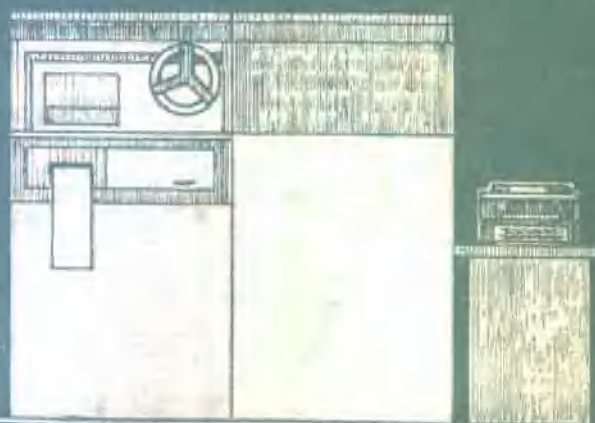


DJS-154

# 计算机原理

上册

北京工业学院



## 出版说明

一九七四年底，本专业74级工农兵学员入学后，为了深入开展教育革命，我们和七三八厂实行厂校挂勾，以该厂DJS-154机为对象，理论联系实际地进行教学。当时，根据教学和其它方面提出的需要，由厂校双方组成三结合小组编写出版了《154小型多功能电子数字计算机原理》上、下两册。参加编写工作的，有七三八厂的工人师傅、技术人员、厂办七二一工人大学的学员，有北工的教员、工农兵学员。此外，还有许多同志热情地为编写教材提供图纸资料，或帮助整理、审阅初稿。因此这是厂校挂勾、开门办学和社会主义大协作的一个可喜成果。因为国内比较系统地叙述微程序控制计算机的教材还不多，所以这个教材出版后受到了各方面的欢迎。但由于教材编写时154机还处于试制阶段，缺陷较多，同时编写也很匆忙，因此教材不可避免地存在一些问题。现在，154机已正式定型和批量投产，因此，我们对原教材进行了修订、改编，成为现在这部《DJS-154计算机原理》，以满足培训用户和各单位学习它的需要。

全书除绪论外，共分十二章。第一章《计算机中数的表示》和第二章《逻辑代数基础》，是学习计算机原理的预备知识。其后的第三章到第十二章，分别叙述154机的运算方法和运算器，内存贮器，只读存贮器，指令系统，微程序控制，中断和数据通道，外部设备，过程输入输出装置，外存贮器，起停电路和控制面板。分上、下两册出版。考虑到这部教材的针对性，凡与154机无关的内容，154机目前尚未配置的设备，一般都没有加以介绍。154机采用的TTL中、高速数字集成电路，在常见的晶

体管电路书籍中都有说明,这本书里也不再介绍。关于 154 机电源系统的描述,已收入我们另外编写的一本《计算机电路》(补充教材)里面。

我们水平很低,经验不足,对 154 机的实践经验和理论分析都不够。因此,本书的缺点和错误肯定还是不少的。热诚地欢迎大家提出批评意见。

**北京工业学院五二专业**

1977 年 10 月

# 目 录

|                        |     |
|------------------------|-----|
| 绪 论                    | 1   |
| 第一节 电子数字计算机的工作方式及其组成   | 1   |
| 第二节 电子数字计算机的使用         | 7   |
| 第三节 DJS-154 机概要        | 9   |
| 第一章 计算机中数的表示           | 12  |
| 第一节 进位计数制              | 12  |
| 第二节 进位制数之间的转换          | 15  |
| 第三节 小数点的表示法            | 18  |
| 第四节 机器数与真值             | 20  |
| 第五节 154 机中数的表示         | 29  |
| 第六节 小结                 | 33  |
| 习 题                    | 35  |
| 第二章 逻辑代数基础             | 37  |
| 第一节 逻辑代数的基本运算          | 37  |
| 第二节 逻辑代数的基本公式          | 38  |
| 第三节 逻辑表达式的简化           | 43  |
| 第四节 小结                 | 53  |
| 习 题                    | 53  |
| 第三章 运算方法及运算器           | 56  |
| 第一节 定点运算方法             | 56  |
| 第二节 浮点运算方法             | 84  |
| 第三节 十进制运算方法            | 94  |
| 第四节 运算器                | 100 |
| 第四章 内存贮器               | 108 |
| 第一节 磁心和磁心存贮信息的原理       | 108 |
| 第二节 电流重合法磁心存贮器         | 129 |
| 第三节 154 机内存贮器          | 141 |
| 第四节 存贮器的自检             | 169 |
| 第五节 内存分调面板和辅助线路        | 181 |
| 附录 154 机内存各点实测波形       | 184 |
| 第五章 只读存贮器              | 187 |
| 第一节 变压器绕线式只读存贮器存贮信息的原理 | 188 |
| 第二节 逻辑组成及原理            | 188 |

|     |                    |     |
|-----|--------------------|-----|
| 第三节 | 固定体 .....          | 199 |
| 第四节 | 分调工作与联调工作 .....    | 200 |
| 第六章 | 指令系统 .....         | 201 |
| 第一节 | 指令的结构 .....        | 201 |
| 第二节 | 指令系统介绍 .....       | 204 |
| 第三节 | 操作地址的形成 .....      | 213 |
| 第四节 | 154 机指令系统一览表 ..... | 222 |
| 第五节 | 机器对指令的识别和处理 .....  | 226 |

# 绪 论

## 第一节 电子数字计算机的工作方式及其组成

电子数字计算机是一种自动的电子设备，它不但能完成加、减、乘、除等算术运算，并且能对参与运算的数据和计算结果进行数量大小的比较和符号正负的判断等（或称逻辑运算），同时具有“记忆”（或存储、寄存）的功能，因而可以自动、快速地解算各种数学问题和逻辑问题。它已日益广泛地应用于我国工业、农业、国防、科学文化等各个领域。我们学习计算机，就是为了使这一工具能更好地为我国社会主义革命和社会主义建设事业服务。

为了了解计算机的构成以及它是如何自动解题的，先考察一下人是如何利用算盘算题的。假设要计算

$$y = \frac{ax+b}{c}$$

我们分析这个问题，确定计算步骤，即

第一步：取  $a$

第二步：作乘法，求  $ax$

第三步：作加法，求  $ax+b$

第四步：作除法，求  $\frac{ax+b}{c}$

只要给定了  $a$ 、 $b$ 、 $c$  和  $x$ ，利用算盘按上述“步骤”进行规定的运算就能得到要求的结果。这里，用纸和笔记录原始数据、中间结果和计算步骤，用算盘实现运算，而整个计算过程都是由人来做的。对于一般复杂的数学问题，也可利用数值分析的方法，将它转变为一系列基本算术运算，解题过程还是类似的。

电子数字计算机是模拟上述解题过程的自动电子装置，为此它必须具备下列最基本的设备：

1. 机器必须能记录、保存和读出“计算步骤”、原始数据和计算结果，好象人用以记录的纸和笔一样，这就是记录装置，一般又叫做存贮器。
2. 机器必须能进行数的运算，象算盘一样。这就是运算器。
3. 机器必须按规定的“计算步骤”控制执行上述操作，这就是控制器。

计算机作为一个完整的机器，除了具备存贮，运算和控制的能力（“主机”功能），还必须把编制好的“计算步骤”和原始数据送进机器，把运算完的结果以某种方式送出来为人们所使用。这就是人们和机器直接打交道的部分。前者一般称为输入设备，后者一般称为输出设备，统称外部设备。

以上各部分都是由集成电路或少量分立元件（晶体管、电阻、电容等）以一定逻辑关系装焊成的，因此还需要有电源，以保证电路的正常工作。

综上所述，电子数字计算机的主要组成部分有：存贮器、运算器、控制器、外部设备与电源等，其相互关系如图1所示。

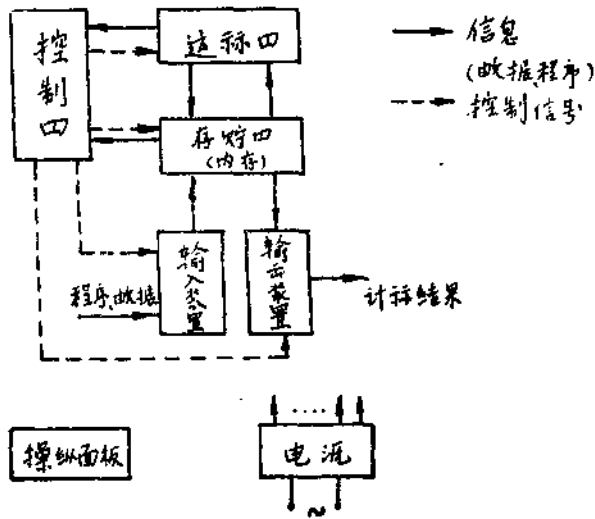


图1 数字计算机组成框图

其中输入装置在控制器的控制下将“解题步骤”和原始数据等输入到存贮器保存起来。计算开始后，存贮器在控制器控制之下随着解题过程的推移，将解题步骤与数据读出，分别送到控制器和运算器。经过运算器的运算后又把运算的中间结果或最后结果送到存贮器保存起来。控制器还控制存贮器将所得的最后结果送到输出装置以某种方式输出。

图1 仅仅表示出电子数字计算机的主要组成部分及其相互关系。实际上每一部分的具体结构和相互关系是相当复杂的，且随任务而异。下面对机器各部件略作说明。

### 1. 运算器

运算器是电子数字计算机主要组成部件之一，它能执行基本算术运算（加、减、乘、除）和逻辑运算（逻辑加、逻辑乘、逻辑比、移位等），根据任务的需要也可执行一些特殊运算，如开平方等。

运算器既然要对两个数进行运算，就至少需要两个寄存器（A、B）存放参与运算的两个数，还有一个功能产生器，它接受两个数（a、b），在控制器发来的控制信号作用下完成各种

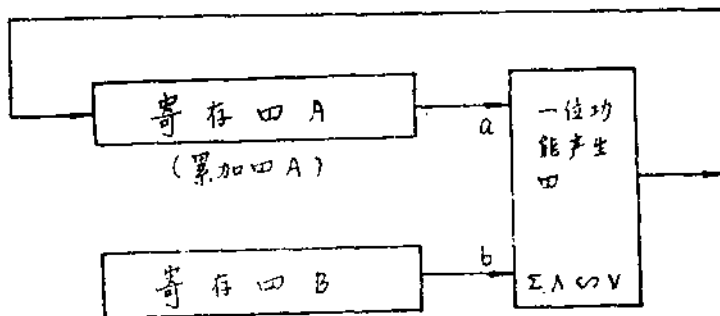


图2 串行运算器

运算,并将运算结果传送到其中一个寄存器去(这个寄存器一般称为累加寄存器,简称累加器)。对完成算术运算来说功能产生器就是全加器。

运算器可分为串行和并行两种。串行运算器对数  $a$ 、 $b$  的运算是一位接着一位进行的,因此它只需要一个一位功能产生器,如图 2 所示。其优点是设备节省,结构简单,缺点是运算速度低。

154 机是全并行运算。并行运算器对数的运算是所有位同时进行的,因此对 16 位数的运算,就需要一个 16 位的功能产生器,如图 3 所示。其优点是运算速度快,缺点是设备较贵,结构比较复杂。

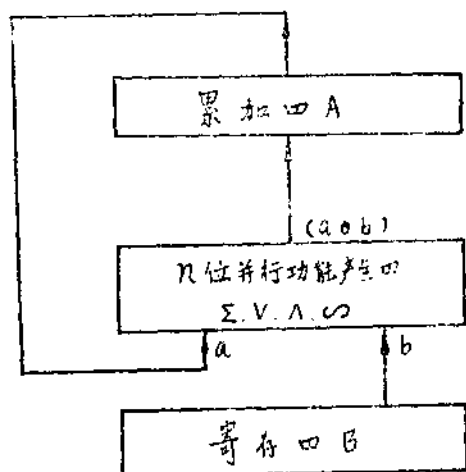


图 3 并行运算器

在单位时间内运算器所能完成的运算次数,称为运算器的运算速度。显然,对于不同的运算,完成的时间是不一样的,如加法、减法比乘法、除法快的多,这是因为在电子数字计算机中执行乘法、除法是通过一系列加减法来实现的。通常所讲的运算速度是指各种运算和操作的平均速度。而常说的机器速度还包括取指令、取操作数的时间。现代电子数字计算机运算速度约为每秒几万次到几千万次。154 机定点加法  $4.8\mu s$ , 定点乘法  $16.2\mu s$ 。浮点加  $33\mu s$ 。

## 2. 存贮器

存贮器在现代电子数字计算机中起记存信息的作用。其功能是接收并存储“解题步骤”(指令代码)、原始数据、中间结果与最后结果等等。计算过程中将需要的指令代码传送到控制器,将数码传送到运算器,从而保证机器能按照解题步骤自动的工作。

因为在电子数字计算机中的数是以二进制表示的,所以凡具有两种不同的稳定状态,且能控制与识别其为某一种稳定状态的器件都可作为“存贮元件”。如触发器某一输出端的高电位与低电位,磁心材料的两种剩磁状态等等,其中一个稳定状态表示“1”,另一个稳定状态表示“0”。在现代电子计算机中常用的是磁心,它是用磁性材料制成的小圆环,外径只有  $0.5\sim 1.2\text{mm}$  左右,它在不同方向的一定大小的电流作用下,会处于两个不同的剩磁状态,从而实现二进制信息的记存。将信息存入存贮元件的过程叫做写入。在已存入信息的磁心中,以一定方向通过一定大小的电流,在穿过磁心的另一条线上就会有不同的感应电压输出,从而实现所存信息的输出。从存贮元件中将所存信息取出的过程叫做读出。

154 机用  $0.8\text{mm}$  磁心,读时间  $1.2\mu s$ ,读写周期  $2.4\mu s$ 。

一个存贮元件记存一位二进制数,若干个存贮元件才能构成一个存贮单元,通常一个存储单元存放一个数或一条指令的代码,也有几个单元存放一个数的。一个存储单元所能记存的二进制数的位数叫做字长,一般是 16 位, 20 位, 24 位, 32 位, 40 位, 60 位不等。154 机的基本字长是 16 位,因之一个存贮单元是 16 位。存贮器中所有存贮单元的数目叫存贮容量,它反映存贮器的记忆本领。随着计算机任务的不同,容量差别是很大的。小的容量只有几个单元,大者上百万,甚至更大。154 机容量为  $4K \times 16$ ,  $8K \times 16$ ,可扩充至  $32K$ 。

为了在解题过程中便于写入或读出所需要的代码,将存贮器所有单元按顺序用二进制编

号。每个存贮单元所对应的号码就称为它的地址，这正好象一幢房屋的房间号码就是它的地址一样。要将一个数码写入存贮器，首先要确定写入到哪个地址的存贮单元中，然后进行写入操作。以后在解题过程中需要这个数时，就从这个地址中将数读出来。

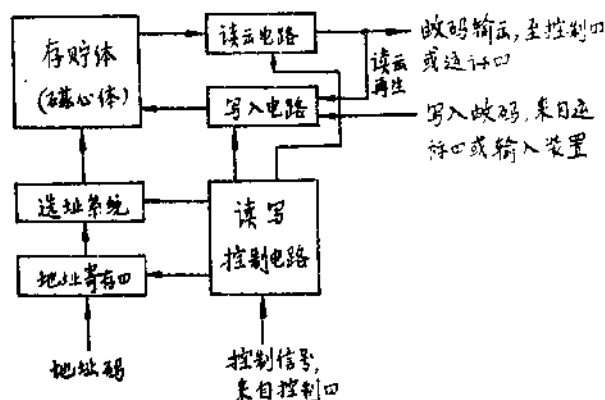


图4 存储器框图

存储器(图4)除了存贮体外，还有地址码寄存器、选址系统、写入电路、读出电路和控制它们工作的读写控制电路。地址寄存器用来寄存地址码，它通过选址系统选择存储体中对应于地址码的存储单元，准备写入或读出。要写入时运算器送来的数码传送到写入电路，在控制电路的控制下实现向所选定的存储单元的写入。在读出时，从选址系统所选择的那个存储单元中将所有信息传送到读出电路，并送至控制器或运算器。

存贮单元所存的信息被读出后，其原来的状态就被破坏了，但它原来的信息以后可能还有用，这就要求在读出所存信息之后，读出电路的输出再传回到写入电路，使之重新写入，这个过程一般叫做重写或再生。

从启动存贮器工作，直到读出与写入操作完毕的时间，叫做存取周期或读写周期。目前磁心存储器的存取周期已做到几个微秒，乃至几百毫微秒。154机磁心存储器读写周期是 $2.4\mu s$ 。

### 3. 控制器

控制器是全机的指挥机构。它根据计算程序实现对各部件的控制，以完成整个解题过程。所谓计算程序实质上是一系列有一定顺序的计算命令，一般一条命令表示一种操作，指出对哪一个数据进行何种操作。因为电子数字计算机中的数一般是以二进制表示的，因此在机器中，每一条命令也用二进制代码表示。这种命令在计算机术语中叫做指令，表示一条指令的二进制代码叫做指令代码。用以解题的指令序列叫做程序。

不同计算机的指令码的形式(一般叫指令格式或指令结构)是不一样的，同一台计算机也可能有几种不同的指令格式。尽管如此，它们所包含的信息内容却不外乎有两部分：一部分表示该指令所要进行的操作，如取数、加法、逻辑、存数等等，这部分代码叫做操作码；另一部分表示参加运算的数存放在什么地方，或从哪里将数取来进行规定的运算，这部分叫地址码。操作码的位数由该计算机所能执行的操作种类而定，如操作码为四位二进制码，则最多可有16种操作。地址码的位数常决定于存储器的容量，设地址码的位数为 $N$ ，则存储器的容量通常为 $2^N$ 。显然最简单的指令格式为：

|     |       |
|-----|-------|
| 操作码 | 地 址 码 |
|-----|-------|

上述讨论, 数是直接从地址码所规定的存储单元中取出或写入的, 称为不变址方式。通过编排程序的实践, 感到采用变址方式更方便, 即操作数不是从指令的地址码所确定的存储单元中取出, 而是将地址码加上一个修改量后才是真正要取数的那个地址。此过程称变地址或变址, 所加上的修改量叫做变址量, 存放变址量的寄存器叫做变址器。变址指令中的地址码叫形式地址, 经过变址后的地址码叫做操作地址。

总结上述, 控制器的功能是:

- ① 控制输入设备将解题程序的指令代码和原始数据等输入到存贮器的指定单元中。
- ② 控制存储器在读指令时, 按照预定的次序读出指令码, 在执行指令时, 根据操作地址读出或写入数据。
- ③ “翻译”指令的内容, 明确该指令所要求的各部件操作内容。
- ④ 发出一系列控制信号至各个部件, 控制它们执行指令所规定的操作。
- ⑤ 控制输出设备将解题结果以某种形式输出。

控制器为完成上述作用, 应具有以下主要组成部分(如图5):

#### ① 指令地址计数器(或称中央计数器)

程序的各指令是按顺序存放在存储器的各单元中的, 也就是说程序中一连串指令是连续顺序排列的。因此在控制器中设有一个计数器, 在解题开始时, 它的状态是第一条指令的地址, 以后每读出并执行一条指令, 这个计数器就自动“+1”, 这样就能保证指令的顺序执行。正因为这个计数器随时指明将要执行的指令地址, 所以叫做指令地址计数器。

但是指令的读出和执行并不一定是顺序的, 有时要突然跳到另一个指令地址去执行另一段程序, 这样的指令地址的转移叫做无条件转移, 对应的指令, 叫做无条件转移指令。有时当执行到某一指令后要根据运算的结果来选择下面的计算方向, 满足某个条件就跳到另一个指令地址去执行指令, 不满足此条件则继续按顺序执行原来的指令序列。这种根据条件来决定指令地址的转移的叫做条件转移, 对应的指令叫做条件转移指令。因此指令地址计数器不但能顺序计数, 而且能接受新的指令地址, 以实现指令的各种转移。

#### ② 指令寄存器

控制器在执行某指令时, 是根据该指令码的内容来实现对各部件的控制的。因此在控制器中, 设有指令码寄存器, 它将从存储器读出的指令码暂时寄存起来, 作为控制执行这条指令

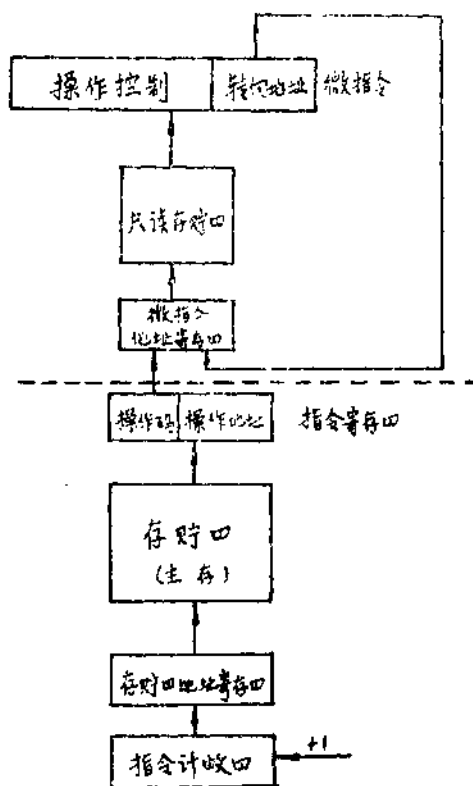


图5 微程序控制器

的根据。

### ③ 微指令地址码寄存器和只读存储器

计算机执行每一条指令，实际上是分成许多步完成的，每一步执行一个或几个微操作。如果我们将每一步所要控制执行的微操作用二进制代码来表示，就构成一条微指令。微指令包含两部分内容，一部分是应该执行的微操作的代码，它表示要执行哪些微操作；还有一部分是确定下一条微指令的转向地址。有时下一条微指令的地址要根据反馈信号来确定。这样，执行一条指令就变成执行若干条微指令的微程序了。我们编制出所有指令微程序，并将它们组合起来，存储在一个高速只读存储器中，由微指令地址码寄存器中的地址来选择并读出对应的微指令。要执行某一条指令时，使微指令地址码处于该指令的第一条微指令的地址（此地址往往由这条指令的操作码决定），从只读存储器中读出第一条微指令，根据其代码发出有关的微操作控制信号至有关部件，使之执行所规定的微操作。该微指令在反馈信号作用下，将下一条微指令地址传送到微指令地址码寄存器中，从而保证在执行完第一条微指令后，接着读出第二条微指令，并执行第二条微指令，如此继续下去，直到对应于这条指令的微指令全部执行完毕为止。

上述的控制方式叫做微程序控制。由于微程序都是以代码的形式存储在只读存储器内，所以按这样的结构组成的控制器被称作是存储逻辑的控制器。154机就是微程序控制的计算机。

除存储逻辑外，还有所谓组合逻辑的控制器。

在组合逻辑的控制器内，除必须设有指令地址寄存器、指令寄存器外，尚设有节拍发生器，通过节拍发生器给出的电位，将机器指令所对应的微操作序列分配在不同的节拍电位内进行。而各种指令在不同节拍内所产生的相同的微操作，以指令特征、节拍电位、操作脉冲为变量，通过逻辑代数综合简化，由“与”、“或”、“非”等逻辑电路组合起来，以实现运算器、存储器及外部设备的控制。

控制器的结构和机器的指令系统是直接相关的，对于不同的计算机，控制器的结构有很大差别，在此不再详述。

### 4. 输入输出设备(外部设备)

电子数字计算机的输入设备一般分两种，一种是通用输入设备，如纸带输入机，它利用机械或光电转换的方法，将纸带上有孔或无孔（即“1”或“0”）变成电信号输入到计算机中；又如键盘输入机，人按动键盘上的某一按键，通过机电的方法向机器输入一个与该按键相对应的代码。另一种是模拟——数字(A/D)转换装置，它将要输入的电压、转角等模拟量变换成对应的数字量送入机器。

同样的，电子数字计算机的输出设备也有两种，一种是通用输出设备，如打印机、X-Y记录仪或显示装置等等，机器通过这些设备将解题结果以数字、符号、曲线或图形的形式打印或显示出来。还有一种是数字——模拟(D/A)转换装置，它将要输出的数字量变换成与其相对应的模拟量如电压、转角等等输送到有关设备。

与内存相对应还有“外存”，外存主要是用磁带、磁鼓、磁盘。外存用于存储大批的数据和指令，作为后续存储器来用。它好比“仓库”，我们把暂时不用的数或指令存在仓库中，需要时再成组的提取到内存。外存容量特大，但速度较低，因此也只能成批的与主机进行数据交换，其作用也相当于输入输出设备，它们统称外部设备。

## 第二节 电子数字计算机的使用

### 一、编制程序

根据求解的问题, 由人给机器规定计算步骤(程序)。程序由一系列的命令(指令)构成。指令按顺序在内存中排列。一般情况下机器按次序逐条执行。也可以通过指令改变执行顺序。每条指令用一个二进制代码表示。

为了便于检查, 可首先根据规定的机器指令系统的文字符号编写成“符号程序”, 上机器算题时, 再将这种事先编好的“符号程序”翻译成代码程序, 机器执行代码程序并算出结果。这种“符号程序”比较直观, 便于检查。

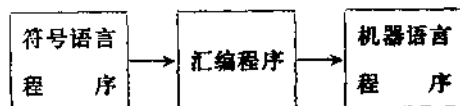
熟练的程序设计者, 也可直接编写代码程序, 它也是按机器的指令系统编写的, 其中指令的操作码、地址码是真实代码, 故又有代真程序之称。这种代码程序凿孔成纸带, 经纸带输入机器后便可直接执行运算。但编写繁琐, 不易检查, 出错也不便修改。

例如计算  $y = \frac{ax+b}{c}$ , 编制如下程序:

| 符号程序(文字程序) |   |     |                             | 代码程序(代真程序) |   |        |
|------------|---|-----|-----------------------------|------------|---|--------|
| K+0        | ↓ | (a) | 取数 a                        | 1200       | 5 | 1301   |
| K+1        | × | (x) | 乘 X, 得 ax                   | 1201       | 3 | 1304   |
| K+2        | + | (b) | 加 b, 得 ax+b                 | 1202       | 1 | 1302   |
| K+3        | ÷ | (c) | 除 c, 得 $y = \frac{ax+b}{c}$ | 1203       | 4 | 1303   |
| K+4        | ↑ | (y) |                             | 1204       | 6 | 1305   |
| K+5        | Ω | --  |                             | 1205       | 0 | 0000   |
| 操作码        |   |     |                             | 地址分配(单元分配) |   |        |
| 0          |   |     | 停机                          |            |   |        |
| 1          |   |     | +                           |            |   | 1301:a |
| 2          |   |     | -                           |            |   | 1302:b |
| 3          |   |     | ×                           |            |   | 1303:c |
| 4          |   |     | ÷                           |            |   | 1304:x |
| 5          |   |     | ↓                           |            |   | 1305:y |
| 6          |   |     | ↑                           |            |   |        |

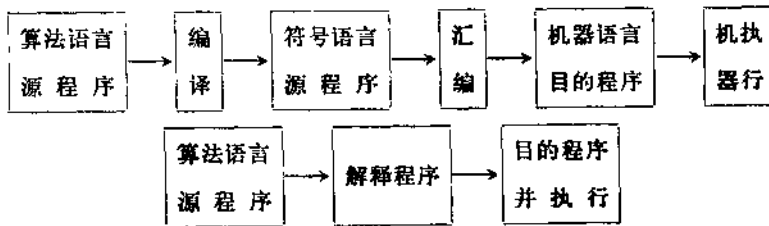
### 二、算法语言的概念

机器本身只“认识”用二进制代码表示的指令。显然, 直接按机器指令系统——“机器语言”编制程序是一件很繁琐的工作。这样编出的“天书”别人很难翻认和检查, 隔一段时间连自己也记不清了。为方便使用, 现代电子计算机都配有“汇编程序”一类的服务性程序, 它能把用符号语言(如+(c)等)写的程序, 汇编成机器语言的代码程序, 其两者之间是逐条对应的, 故各机器不能通用。



近十几年来，又发展了许多“算法语言”。它是根据界乎数学语言和机器语言之间的某种程序设计语言（一种特定的算法语言，其规则与人们常用语言更接近）来编写源程序，这种语言源程序在机器上不能直接执行，是经过事先编好的“编译程序”把源程序翻译成代码程序，然后在运行系统的配合下执行代码程序得出所要求的结果。编写语言程序是很直观的，大大简化了编写程序的工作，易于查错，出错也便于修改。但是在机器上必须首先“翻译”或“解释”成代码程序才能执行，因此较直接执行代码程序化费的机器时间要长些。

算法语言可分为两种类型：一种是先将用算法语言编写的“源程序”，由“编译程序”翻译成符号语言的程序，再经“汇编程序”汇编成机器语言的目的程序，交机器执行。另一种是用“解释程序”逐条解释用算法语言编写的源程序，并直接由机器执行。无论哪一种，源程序都无须与机器指令逐条相对应。



目前算法语言已有若干种使用在不同的机器上和流行于不同的行业中。较常用的有以下三种：ALGOL60，FORTRAN，BASIC。

前两种语言，即 ALGOL60 及 FORTRAN，是属于上述的第一种类型。而 BASIC 语言是属于第二种类型。后一种解释性的算法语言容易掌握，使用方便，但需占用较多的机器时间。如 NOVA1200 机本来运算平均速度达每秒几十万次，而使用 BASIC 语言后每秒只能进行约一千次的算术运算。

国内目前使用 ALGOL60 的较多。一般都是针对所用机器的特点，略作一些修改。我们 154 机上述三种语言都已配用。

作为一个完整的现代电子数字计算机系统，除了需要包括一定的基本部件——通称为“硬件”或“硬设备”，为了方便使用，扩大其功能，还必须配备成套的服务性程序，如汇编程序、算法语言编译程序、解释程序，以及标准子程序库和必须的检查、诊断程序等等。这些预先编好的服务性程序通常称为“软件”或“软设备”。一般是先记录在磁带等外存中，或作孔成纸带。软设备通常由厂家与硬设备同时设计和生产出来，是计算机的一个重要组成部分。从发展趋势来看，软设备可能显得比硬设备更重要。特别是小型机，主机结构很简单，许多功能（包括乘、除法）都靠软设备实现。

### 三、数字计算机解题步骤

#### 1. 数学描述

针对问题的物理过程，抓住主要矛盾，进行数学描述，建立起完整的数学公式或方程。

#### 2. 数字加工，选择数值计算方法

现代电子数字计算机基本上只会进行加、减、乘、除等简单的四则运算。通常遇到的各种数学问题必须采用相应的数值计算方法归结为四则运算（例如定积分的计算用辛浦生公式化为有限项的求和）。计算方法的选择对于取得可靠的结果，简化程序，节省机器时间有决定性的意义。

此外, 机器能表示的数有一定范围, 超过了界限, 机器就要“溢出”。在编制程序之前必须消除造成“溢出”的各种可能性。

### 3. 画框图和编程序

把经过分析加工的解题步骤用流程图表示出来就叫框图。这里包括各种可能遇到的情况及处理方式(分支)和计算中所需要的重复(循环)。然后根据框图编写程序。一般的程序至少包括原始数据输入, 计算本身和计算结果输出三部分。

### 4. 试算及查错

即使经过认真检查过的程序也难免有错, 因此必须进行试算。有些错误可以在试算过程中发现、改正。有些则需要下机后检查。为查错方便, 有时需要尽量多打印出一些中间结果。往往在试算时有意识地选取一些容易直接用手验算的参数(可能与物理问题并不对应), 比较机器与人计算的结果, 以证明程序的正确性。

### 5. 正式计算

## 第三节 DJS-154 机概要

DJS-154 是一台字长 16 位的小型多功能电子数字计算机, 适用于生产过程控制、数据处理及小型科学计算和工程设计。它采用微程序控制, 接口灵活, 并有可选长指令部件。全机由中央处理部件、内存贮器、外部及外围设备三大部分组成。它们之间的关系见系统框图(图6)。

中央处理部件由只读存贮器、运算器组成。

只读存贮器用来存放微程序, 做为微程序控制存贮器。存贮体由 MXD4000 的磁环组成。为提高稳定性, 采用双体结构, 使其交叉工作。为提高可靠性及主机运算速度, 设有缓冲寄存器、容错逻辑线路(每一地址允许连续出错二次)及执行寄存器。将读出下一条微指令的时间(或校错及其重读时间)重叠于执行微指令的过程之中。被执行的微指令采用分组编码控制。只读存贮器容量为 512 个单元, 字长 53 位, 存取周期为 600nS。

运算器基本字长 16 位, 加可选长指令部件后可实现双倍字长的运算。

运算器主要由十六位并行全加器及相应的寄存器组成。全加器的进位链采用四位一小组、十六位共分四组的组内并行、组间串行的快速进位方法。

运算器的各寄存器通过全加器连接。每位全加器都有一个四与非门作为它的输出开关, 可将全加器运算结果直接输出, 或左移一位、右移一位, 或按位加输出, 这样就赋予了各寄存器以移位的功能, 省去了各寄存器间的信息传递通道, 简化了寄存器的结构。

内存贮器采用三度三线电流重合法, 存取周期  $2.4\mu s$ , 基本容量 4K, 且以容量 4K 为模, 最大容量可扩到 32K。

内存贮器为并行读出和写入信息, 设有十八位重写寄存器(其中十六位存放信息, 一位奇偶校验, 一位保护)。重写寄存器不仅用来再生或写入信息, 而且还用作主机和外部设备交换代码的接口寄存器。

内存贮器不但能在程序控制下与中央处理部件累加寄存器 A 交换数据, 而且可以通过数据通道直接与高速外部设备交换数据。

主机与外部及外围设备的联接采用总线结构, 设备通过标准接口挂在 50 条输入输出母线上。设备的增减, 可通过增减接口来实现, 无须改变主机逻辑。接口允许有三个缓冲寄

指令

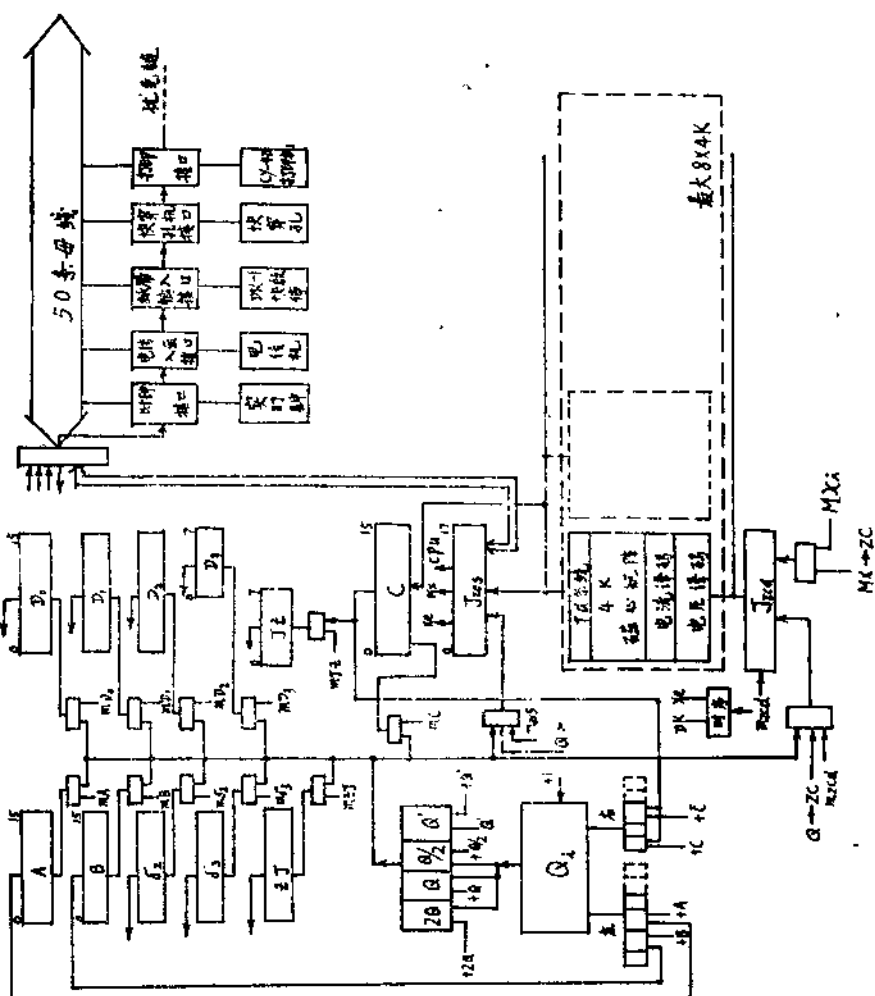
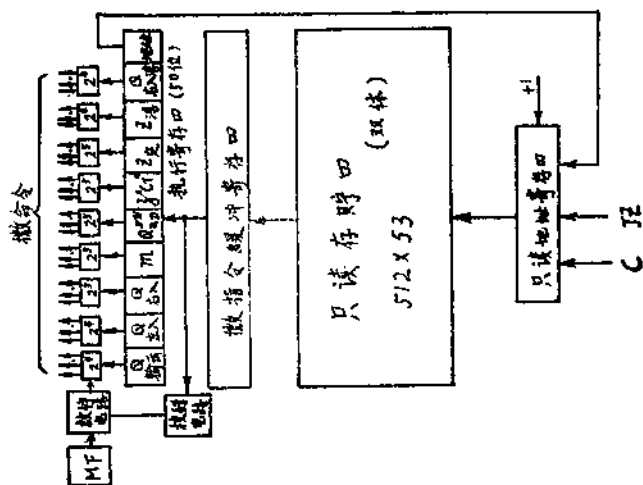


图 0 151 机系统框图

存器。标准接口控制逻辑包括有：设备占线触发器，完成触发器，请求中断触发器，中断屏蔽触发器，设备码选择线路，优先链输入、输出及优先设备码产生等线路。中断级别的高低可用设置设备的屏蔽状态来任意调整。多重中断由程序处理。一般慢速设备由程序中断和主机累加寄存器A交换信息。高速设备及外存用数据通道形式直接和内存快速交换数据。输入输出最高交换频率为 333KC。

电源系统采用 1000 周交流机组或逆变器集中供电。

主机由基本设备柜和主机柜两个机柜组成。基本设备柜内有 CY-4D 打印机, DR-1 电容式纸带输入机及全机稳压电源。主机柜由两个组合块组成，每个组合块有 20 个插件位置。第一个组合块为中央处理部件及内存贮器，第二个组合块为外部及外围设备的标准接口。

154 机的指令系统共包含 63 条指令，并有广义指令可用以根据需要扩充机器的指令系统。几种基本指令的执行时间为：

定点加法指令  $\geq 4.8\mu S$

定点乘法指令  $16.2\mu S$

浮点加法指令  $\geq 33\mu S$

# 第一章 计算机中数的表示

一台电子数字计算机，不管它是用于科学计算的，还是用于数据处理的，还是用于工业控制的，都免不了要与大量数据打交道。它要对输入的各种原始数据进行运算、加工，其最后结果也需以数据形式输出。因此，计算机中数的表示方法就直接影响着计算机的结构和性能。而要了解 and 掌握一台电子数字计算机，也必须首先了解计算机中数的表示方法，作为学习计算机的基础。所以，在这第一章中，我们将讨论各种不同的进位计数制，重点讨论数字计算机中经常用到的十进计数制、二进计数制、八进计数制；它们之间的相互转换；讨论计算机中小数点的表示方法，引出定点和浮点的概念；讨论数的原码、补码、反码和移码四种表示法，这四种码的一些基本性质，为进一步学习 154 机的运算方法打下基础，最后给出 154 机中数的十一种形式。

## 第一节 进位计数制

在生产劳动和日常生活中，我们最常用、最熟悉的的就是十进制数，它的数值部分是用十个不同的数字符号 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 来表示的，我们把这些数字符号叫做数码。同一数码处于不同位置(或数位)代表的意义是不同的。例如 1974.11，在这个数中，小数点左边的第一位 4 代表个位，表示它本身的数值，左边第二位 7，表示  $7 \times 10$ ，左边第三位 9 是百位，表示  $9 \times 100$ ；左边第四位 1 是千位，表示  $1 \times 1000$ 。而小数点右边的第一位 1 表示  $\frac{1}{10}$ ，第二位 1 表示  $\frac{1}{100}$ ，因此这个数可以写成：

$$1974.11 = 1 \times 10^3 + 9 \times 10^2 + 7 \times 10^1 + 4 \times 10^0 + 1 \times 10^{-1} + 1 \times 10^{-2}$$

一般地说，任意一个十进制数 S (为方便起见假设是正的)都可以表示为：

$$S = K_n(10)^n + K_{n-1}(10)^{n-1} + \dots + K_1(10)^1 + K_0(10)^0 \\ + K_{-1}(10)^{-1} + \dots + K_{-m}(10)^{-m}$$

或 
$$S = \sum_{i=-m}^n K_i(10)^i \quad (1-1)$$

其中  $K_i$  可以是 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 十个数码中的一个，它由 S 决定；m, n 为正整数。括号内的 10 称为计数制的基数。

所谓进位计数制(简称进位制)，就是说它是按进位方式计数的。所谓某进位制的基数，就是在该进位制中可能用到的数码量。当基数为十，每位计满十，向高位进一，即“逢十进一”，这种进位制就是十进制。除了十进制，在生产和生活中我们还碰到非十进制的计数制。比如时间，六十秒为一分，六十分为一小时，它是六十进制的。再如通常说的“一打铅笔”，就是十二支，它是十二进制的。基数为二的进位制，即二进制，是电子计算机中广泛采用的进位制。

在二进制中，每个数位只可能取“0”和“1”这两个不同的数码，而且是“逢二进一”的。下面列出几个简单、常用的数字的十进制与二进制表示的对照表。