

TQ—16 机 算法语言介绍

中国科学院计标中心筹备处

===== 1975·9 =====

序 言

根据我们的意图，编写这份资料的任务，是向用户推广普及TQ-16机的编译系统。为了达到这个总的目的，我们认为这份资料应该包括下列三方面的内容：1) ALGOL60 参考语言的基本方案；2) TQ-16机的出版语言及其硬件表示的详细说明；3) 怎样用TQ-16机的林达语言编写程序的流程图。

大家知道，林达语言是描述数值计算过程的一种方式，它的内容包括运算对象和运算规则的描述，另的说明和语法结构的描述，以及计算过程的计算步骤和执行情况描述等等。为了使这份资料能够尽可能做到通俗易懂，我们所采用的办法是，完全抛开ALGOL60语言的形式化定义，而着重于叙述它的基本概念和基本方法。我们想力图做到使读者在阅读这份资料时，不至于感到它只是形式上的一些硬性规定，而相反地觉得这些规定是很自然的，甚至是不可避免的。为了这一目的，我们采取了下面一些具体的措施。

首先，为了加深读者对基本概念的理解，我们尽可能地依照做字语言的思路，或者按照人们通常所能接受的普通习惯来叙述各种基本概念。特别是我们不惜惜况一些话，来帮助读者时时刻刻都能记住所遵循的这条基本的路线。但是很遗憾的—是，在编写第四章（主要是过程）时，虽然我们很想尽力做到既系统全面又容易接受，可是我们没有得到成功。倘若希望这份资料能够完全令人满意的话，无疑似有些章节在以后需要加以修改。

其次，从用户的角度出发，读者所关心的是用林达语言编写程序的一些基本方法和技巧。为此在整个资料的布局上，我们放弃了通常介绍林达语言的传统方式，而完全按照自然的发展顺序，把另的说明和程序结构放在语言节分的前面来加以介绍。这样做的优点是，一方面可以使读者及早的了解到用林达语言所编写的流

程序的结构。另一方面在介绍各种语句时，我们又可以写出一个小计算问题的完整程序。除此以外，在第六章的最后一节中，我们还进一步的罗列了一些有关提高编写程序质量的技巧。

另外，针对不同读者的需要，我们根据各种语句的常用与不常用，以及由简单到复杂的原则，把它们分成三部分来加以介绍。对于初学林法语言的读者，只要阅读前三章，就能利用常用的语句来编写各种计算问题的程序；对于熟悉ALGOL 60 语言的读者，只须阅读第二章的§2-§4、第三章的§5以及第六章的§1，就能利用TR-16机林法语言来编写他所要计算问题的程序；对于需要使用TR-16机特殊功能的读者，他只要阅读第五章，就能了解到TR-16机的各种辅助语句的功能及其使用方法。

最后，特别需要说明的是，我们过去都不是搞软件工作的，大家也是初学TR-16机的林法语言，对其编译系统也不甚了解，尤其是在TR-16机上还没有林法的实践经验。加上是匆匆上马，边学边想边写，仅仅经过起草、修改，誊写三道手便交去刻印，所以错误和不当之处在所难免，我们诚挚地希望用户和读者多提批评和意见，以便我们在以后的工作中随时加以补充和修改。

目 录

序言	iii
第一部分 IQ-16 电子数字计算机简介	1
第二部分 标法语言和编译系统	5
第三部分 IQ-16 机标法语言介绍	9
第一章 标法语言初步及其基本成分	10
§1. 程序的初步概念	11
§2. 基本符号	13
§3. 标识符、常号、变号	16
§4. 标准函数反函数命名符	19
§5. 表达式	25
第二章 说明和程序结构	33
§1. 简要说明	34
§2. 数组说明	35
§3. 复合语句和分程序	43
§4. 程序结构	47
§5. 注释	50
第三章 标法语言的常用语句	53
§1. N 格式输入语句	54
§2. IO 格式输出语句	56
§3. 赋值语句	57
§4. 转向语句	61
§5. 条件语句	63
§6. 循环语句	69
§7. 空语句	77
§8. 停机语句	78
第四章 开关和过程	81

§1	开关	82
§2	过程说明	85
§3	过程语句	91
§4	函数过程	105
§5	库过程	111
§6	标号的存在域和作用域	112
第五章	辅助语句	115
§1	输入语句	115
§2	输出语句	118
§3	读、写数据语句	129
§4	读、写带语句	131
§5	点灯语句	132
§6	中断语句	133
§7	付本语句	134
第六章	有关编写程序的若干注记	137
§1	编写程序的一般准则	137
§2	程序中的错误	142
§3	编写程序的质易和技巧	146
附录一	TQ-16 初始语句	151
附录二	TQ-16 机标往标法库过程目录	165
附录三	程序编制的限制	179
参考资料		180

第一页

TQ—10 电子数字计算机简介

TQ—10 计算机是以集成电路为主要元件的中型通用电子数字计算机。TQ 分别是图、强两个汉字的汉语拼音 Tu. Qiang 的第一个字母。该机是在 709 计算机的基础上研制成功的，平均速度每秒钟运算十二万次，字长为二进制 48 位，内存共有 32768 个机内字，并配备有两台磁鼓和两台磁带机作为外存贮器。

TQ—10 计算机配有检查程序、管理程序及标准语言 ALGOL60 编译系统。在一般情况下，用户是处于管理程序控制下进行工作，但鉴于该机具有封闭中断系统的特性，于是，在特定的场合下，机内可以不处于管理程序控制下而直接运行。另外，在鼓中还固定贮存有 22 个标准函数程序程序和 30 个标准标准法程序。

在这里我们仅对 TQ—10 计算机的硬设备作一简略的介绍。

一、主机系统结构

从系统结构上讲，TQ—10 计算机的主机由控制、内存和通道三大元件所组成，这三大元件之间的关系可图示如下：

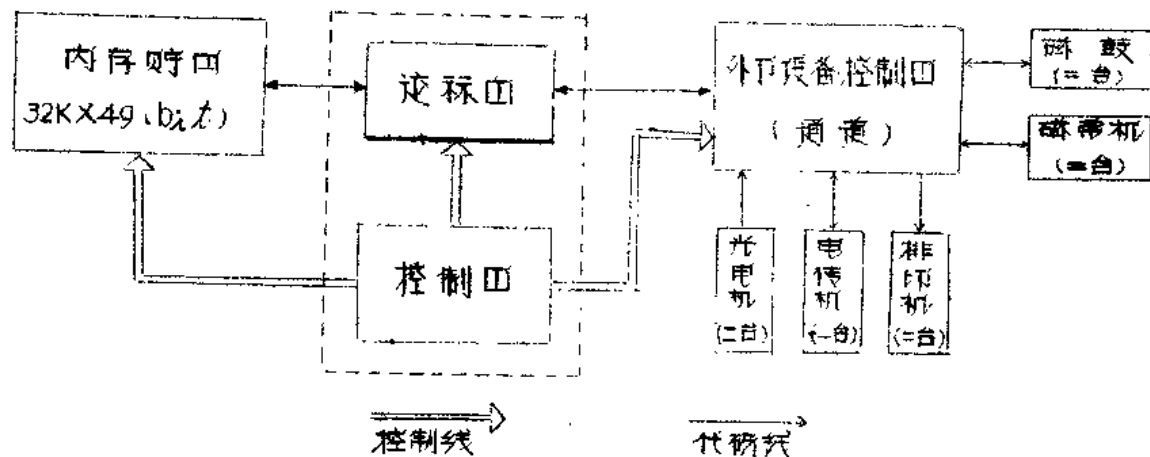


图 1

1. 总控

总控包括选择口和控制口两个部分。

选择口 主要由乘商寄存器、累加寄存器、接受寄存器和一个加法口组成。其功能是在控制口的指挥下，对内存口内的信息进行加工和处理。

控制口 是联系三大部件的主要部件，其中包括一个使用控制台和一个维护控制台。使用控制台是用户应用编辑程序标题时对所口进行人工操作的装置；维护控制台主要是供维护人员和应用程序标题人员使用。

2. 内存

内存就是内存口，它是以集成电路和薄膜电路为主要电路元件，并采用宽温砷心体制成。其容量为32768个单元，单元地址编号依次为0~32767，若采用八进制数编号，则应为000000~077777。存取周期为2微秒，读出时间为1.2微秒，读写方法为二度半法。

3. 通道

通道是连接主机与外部设备的控制口。这些外部设备是：输入机、输出机、电传机、磁鼓和磁带机。由于这五种外部设备公用一个通道，所以，对主机来说，这些外部设备的传送操作是串行的。换言之，当外部设备传送操作时，主机便不执行其它指令，直至一个输入或输出命令完成后，主机方能继续执行下一个指令（由于打洞机空进纸和磁带机反干涉不与内存交换信息，所以这些情形除外）

二. 外部设备情况

1. 输入机

输入机是两台5—8单位电纸带输入机，分上下台，上台编号为0，下台编号为1。输入速度是每秒钟走带2米约800那孔。

2. 输出机

输出机是两台宽 80 个字符的快速打印机，其编号依次为 0 和 1。不同字符的总个数共 64 种，打印速度为每秒钟 10 行。

3. 电传机

电传机是一台 BD-55 型电传打字机，其主要功能是用来输入各种命令和输出某些信息，以便进行人一机通讯。

4. 磁鼓

磁鼓是作为外存储器使用。目前配置两台 9-3 型立式磁鼓，但可以扩充至四台，其编号依次为 0, 1, 2, 3。除 0 号鼓是

固定程序编译系统等软设备以外，其余的鼓在标题时都可以使用。每台鼓分 28 道 512 组，因而有 14336 个单元，其编号依次为 0 ~ 14335。若采用八进制编号，则应为 000000 ~ 033777。鼓的旋转速度是每分钟 15000 转，平均字符时间为 40 毫秒。

5. 磁带机

磁带机也是作为外存储器使用，现配有两台 16 道磁带机，但可以扩充至四台，其编号依次为 0, 1, 2, 3。每台磁带机的磁带长度为 1015 米，每米约有 962 个号，两台磁带机总共约存 192 万个号。磁带机的走带速度 2 米/秒，并采用随机读写的工作方式进行内外信息交换，交换速度为 25 4 周，启动时间为 0.1 秒。

所谓随机读写工作方式，它包含两方面的含义。

第一、程序中时使用的磁带机的台号是形式的，即在机口运行过程中，所有联机的磁带机都允许使用，当目的程序中所需的那台磁带机是正常的，那么，此时形式台号就等于实际使用的磁带机的台号；如果目的程序中所需的那台磁带机发生故障，那么，机口就自动调换磁带机，此时形式台号便不等于实际使用的磁带机的台号。

第二、每台磁带机上的磁带可以划分为任意组，每组容量也是任意的，而且组号只需由小到大排列使用，但不一定是连续。鉴

于机口表示组号的限制，每台磁带机上的磁带最多只能划分成4096组。但是，使用标法语言标题时，组号限制在2~4039。

三、中断系统

TQ-16计算机与T09计算机最大的区别是在于增加了一个中断系统，其目的是用于监督和处理机口内所出现的某些偶然性或系统性的错误，并响应控制台上的人工干预。

1. 中断源

中断源系由一些中断性质标志触发口所构成，它们的内能是显示机口内所产生的一些错误的性质。TQ-16机共配有13个中断源，其中有4个尚未定义，以作备用。

2. 中断计数器

中断计数器是指中断所处的级，并处理中断的入口地址和返回地址。TQ-16机能响应三级中断，但受理程序只处理二级中断，当三级中断产生时，机口立刻停机。

3. 中断屏蔽触发口

中断屏蔽触发口是由控制屏蔽触发口、内存屏蔽触发口和外口屏蔽触发口所组成，其功能是分别指示控制、内存和外口开放中断或封锁中断。

顺便指出，随着科学技术的不断发展，人们对计算机性能的要求愈来愈高，为了提高计算机的工作效率，增加机口的可靠性，因此，对于中断系统的研究，目前已成为计算机设计工作中一个重要课题。TQ-16机所设制的中断系统，只能对机口在信息传输过程中所产生的某些错误和程序执行过程中的溢点进行监督和處理，或当及时报告上机人员进行某些实时处理（如及时电传机人工给出实时处理信息），但是，由于该机是单通道的，因此，它不能实现分时操作。

第二章

译法语言和编译系统

大家知道，使用电子计算机解题，首先要将计算步骤用机内指令（机内语言）形式编成程序，这一工作称为程序设计。但是直接使用机内指令编写程序（称为手编程序）是极其繁琐的工作，需要耗费大量的人力和时间，而且机内语言与数学语言差别很大，难学难写，并容易出错，还不易检查，即使查出错误也难以修改。再加上不同型号的机内有不同的指令系统，所以，程序设计一般需要受过一定专门训练的程序员人员来做，这就大大影响了计算机的推广应用。

为了解决上述矛盾，近年来人们参照数学语言设计了一种译法语言，这种语言比起机内语言来大为方便，它既简单、直观和通俗，又容易写，同时不易写错，即使写错了也容易查找、查出错误后又十分容易修改，因此，一般译法人员只需稍加学习就能掌握，而不需要专门程序员人员来做，这样，也就大大有利于电子计算机的普及和推广。

然而，由于计算机不懂译法语言，所以，用译法语言编写的程序在机内上一般要分两步来实现：

第一步，先把用译法语言编写的程序翻译成等价的机内指令形式的程序，这一过程称为编译阶段。它要由一种称为编译程序的特殊程序来实现。

第二步，执行由编译程序翻译出来的机内指令形式的程序，并对初始数据进行处理，直至得出最后结果，这一过程称为运行阶段。在运行阶段中，需要有若干个子程序（如做初始条件存储分配子程序，下标变量直接地址计算机子程序）陪伴机内指令程序进行工作，这些子程序总起来称为运行系统。

编译程序和运行系统合起来通常称为编译系统，它是人们事先编制好固定存放在机内存储器或外存储器中，使用时由机自动调用执行。这种由机来编程序的方法就是所谓的程序自动化。

计算机配上了编译系统就好像给机增添了一种新的设备，由于这种设备是由一些特殊的程序所组成，所以，相对于机本身的设备（一般称为硬设备或硬件）而言，人们通常就把这种程序称为计算机的软设备或软件。除了编译系统以外，一般地讲，凡是通过固定程序来实现某种操作的系统都称为软件，因此，软件又可叫做程序系统。

用某种语言编写的程序通常称为源程序，翻译后成为机内指令形式表示程序，一般称为结果程序（或称目标程序、目标代码、代码程序和机内代码）。

上述方法的流程图可以图示如下

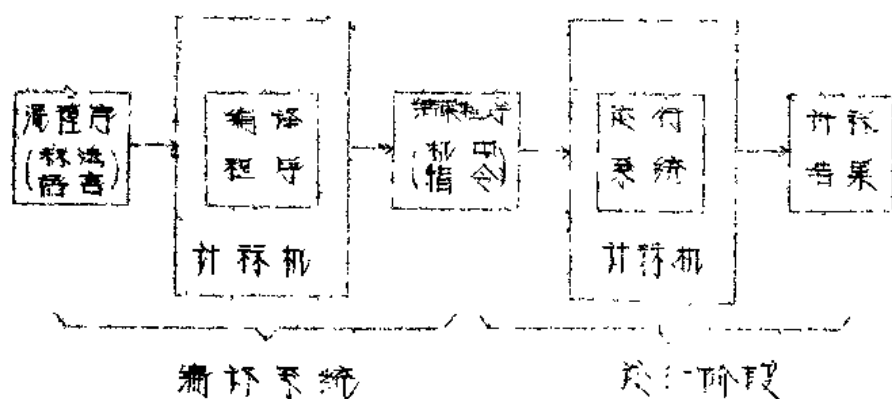


图 2

应该指出，源程序也可以通过解释系统来进行解释执行，即把源程序的语句逐个翻译并立即执行。换句话说，解释系统不是编制结果程序再执行，而是直接解释得出结果，这就象在微型计算机上应用语言解释系统实现语言处理那样。另外，还可以把编译和解释结合起来，即先把源程序编译成一种中间语言程序，然

后，再对此中间语言程序进行解释执行，直到得出最后结果。

从用户使用角度出发，由于标志语言既直观、简单、方便，而且一般又不依赖于具体的机器，所以西方资本主义国家从1958年起，基本上普遍使用标志语言来编写程序，而不再使用手编程序了。我国从1970年开始，也大量应用和推广一些标志语言，如科学院计算机研究所的BCY标志语言、四机部十五所配备的108(乙)机标志语言、科学院数学所计算机站修改的DJS-21机的标志语言和上海计算机技术研究所等单位配置的709机标志语言等，它大大地促进了我国电子计算机的普及和推广工作，使这一先进的计算机工具更好地为我国社会主义建设服务。

但是，事物总是一分为二的，标志语言虽然优越性很多，但用标志语言编写程序、在机器上翻译成的结果程序要比手编程序的质差得多。一般而言，由源程序翻译成的结果程序比手编程序在机器上的执行时间要长两倍，编译系统质差的还要更长些。因此，在某些特定的场合下，手编程序还有其使用的价值。例如，一标就是几个月的大型题目，以及比较常用的定型题目，还是使用手编程序更经济。

标志语言的提出已经有20多年的历史，最早是由H. Rutishauser于1951年提出的，但由于不太适用，因此未引起人们的重视，也从没有在机器上执行过。后来，美国IBM公司在1957年发表了标志语言FORTRAN (formula translation)；西欧在1958年提出标志语言ALGOL 58 (algorithm language)；美国和西欧几个国家在1960年巴黎国际伏羲加工会议上，确定了一种国际通用的标志语言ALGOL 60；1968年国际伏羲加工协会又通过了标志语言ALGOL 68。直到目前为止，已经列出的标志语言约有四百多种，其中除三百多种是专用语言外，约有近百种是通用语言。在这些语言中，最常用的有八种，这就是在科学计算方面用的FORTRAN, ALGOL 60, ALGOL 68,

- 8 -

PL/I (Programming Language/One), BASIC (Beginner's All purpose Symbolic Instruction Code) 和 APL (A Programming Language) 语言; 作数值处理的 FORTRAN (FORmula TRANslation) 语言; 以及作文本加工用的 LISP (list process) 语言等等。

第三部分

TQ-16机译法语言介绍

TQ-16机编译系统是在TQ9机编译系统的基础上做了部分的修改而搞成的，因此，在语言的考虑上，它几乎全保留了TQ9机译法语言的成分和功能。另外，根据标题实践的需要和机中设备的不同，TQ-16机编译人员对TQ9机语言的某些限制和功能作了部分的修改，并增加了若干新的内容。

总的来说，跟TQ9机一样，TQ-16机译法语言基本上是以修改的ALGOL60报告为蓝本，同时考虑到一方面为了减少编译系统的复杂性，并受机中规模和外设设备的限制，删去ALGOL60报告中一些不太常用和影响不大的成分和功能，另一方面，按照实际标题的需要，又适当地增加了一些新的内容。

这里我们仅对TQ-16机新引入的主要语言成分和功能作一点简述，而共删除的成分和若干限制我们将在后面的正文中随时加以叙述和说明。

首先，在ALGOL60报告中，对输入、输出没有作具体的规定，在TQ-16机译法语言中，引进了代码交换的各种方式，特别是输入和输出可以容许多种格式，这是该译法语言的一个明显的特点。另外，在TQ9机语言的基础上，该机还增加了若干显示和逻辑等功能（如点灯、中断等等）。

其次，在TQ-16机译法语言中，定义了22个标准函数，在这些标准函数中，除了有几个是ALGOL60报告中建议必须引进的外，其中多数是国内其它型号机中上所不具备的功能。如象按位左移的二值标准函数，尽管它们在实际标题中用得很少，但当一旦要用时，则只能采用算法语言的成分来实现，这就导致大量占用内存单元，并给用户施加很大困难。

再次，从方便用户的角度出发，在TQ-16机中还引进了30

个标准算法库过程，在下降过程中，基本上包括了经常用到的各种成熟的计算方法，它们都已编成独立的流程说明的形式存放在鼓中，在标题时可以任意调用。

最后，为了流程图的书写简便和减少穿孔的工作量，在TQ-10机中，对于所定义的两个逻辑值和39个等定义符中的绝大多数等定义符，都引用英文缩写形式。而且规定，在书写流程图时，允许采用两种书写形式中的任何一种，或者在同一流程图中可以混合采用两种书写形式。

第一章 算法语言初步及其基本成分

算法语言是描述计算过程和计算规则的一种方式，为了使算法语言比较接近数学语言，而且形式直观，结构简单、概念明确、易于为一般人理解和掌握，因此，人们首先要求它能够直接表达各种计算对象和计算规则，并尽可能地保留数学语言的主要特点及其描述计算对象和计算规则的基本方式。

众所周知，用数学语言表达计算对象和计算规则，主要是一些抽象的符号和各种符号的有序组合，譬如说，数、变量和函数。一般是用阿拉伯数字和英文字母来表示，而各种数学公式则是由一些抽象的符号按一定的规则组合而成等等。算法语言基本上保留了数学语言的这些特点，并且有意识地保留了其数学分析的含义。

在本章中，我们首先以一个简单的例子，来说明用算法语言所编写的流程图的大致形式和基本结构。接着列出TQ-10机所允许使用的64种基本字符，两个逻辑值和39种等定义符。在此基础上，详细叙述用这些基本符号描述数、变量、标准函数、一般函数和各种数学表达式的一般形式和基本规则，从而就构成了算法语言的基本单位——明显的数学公式。

§1. 程序的初步概念

对于实际计算人员来说，他最急于要了解的问题是：如何用基本语言编写程序（即源程序，以后简称程序）？程序的大致形式和基本结构怎样？我们从一个简单的例子入手，先给读者一个基本的概念和初步的认识。

例 已知 R 和 x 的值，要求按公式

$$y = \frac{2Rx}{(R^2 + x^2)^2}$$

计算 y 的值。

如果要在计算机上实现这个问题的计算，那么，人们很自然的想到至少要有以下四个步骤：

1. 首先应该说明 R , x , y 是在什么数域内取值，并在机内给以内存单元地址分配；

2. 然后将 R 和 x 的初始值，分别输入到它们所对应的内存单元中去；

3. 按公式 $y = \frac{2Rx}{(R^2 + x^2)^2}$ 计算 y 的值，并把计算结果存放在 y 的内存单元中去；

4. 最后将 y 的内存单元中的计算结果印在输出纸上。

用基本语言编写这个问题的程序，正是按照上述步骤一步一步的去做。其形式如下：

```

begin
  declare R, x, y;           ..... 说明变量
  #read(0, 'n', R, x);
  y := 2 * R * x / (R↑2 + x↑2)↑2; } ..... 语句部分
  #print(0, '10', y)
end. ^ ^ ^ ^ ^

```

上列程序中各语句的含义和作用如下：

`EREADER, X, Y` 表示 R, X, Y 是在实数域内取值的变量，并对它们在内存储器内进行算术单元，在标准语言中，它称为说明；

`#read(0, 'n', R, X)` 表示将 R, X 的值以通常十进制数的书写形式，由光电带输入机输入到机内存储器 R 和 X 的相应单元中去的输入语句；

`Y := 2 * R * X / (R↑2 + X↑2)↑2` 表示根据给定 R, X 的值，按公式 $Y = \frac{2RX}{(R^2 + X^2)^2}$ 计算 Y 的值，并把计算结果存放到 Y 的相应内存单元中去的计算语句，或称赋值语句；

`#print(0, '10', Y)` 表示将 Y 内存单元中的值，按十进制浮点数的形式由快速打印机印在输出纸上的输出语句；

`ebegin` 和 `ende` 是一对语句括号，表示括在它们之间的一组说明和一组语句是一个完整的计算过程；

`^^...^` 是连写的15个以上的字符 `^`，它表示程序的结束符号；

由此可见，用标准语言所编写的程序，其基本结构大致有如下形式：

`ebegin`

对说明对象及其性质作说明的说明语句；

描述计算过程的语句语句；

`ende^^...^`

从这一简单的例子可以看出，用标准语言 ALGOL 60 编写的程序，主要是由一系列说明和一系列语句所组成，而且所有说明必须放在程序的开头。说明和说明，语句和语句，以及说明和语句之间必须用分号“;”隔开，以表示它们具有各自的独立含义。因此，自然也就有必要用一对语句括号 `ebegin`, `ende` 把它们括起来，这样，才能表示它们是描述一个完整计算过程的整体。

总之，不论是说明还是语句，或者是语句括号，整个程序都