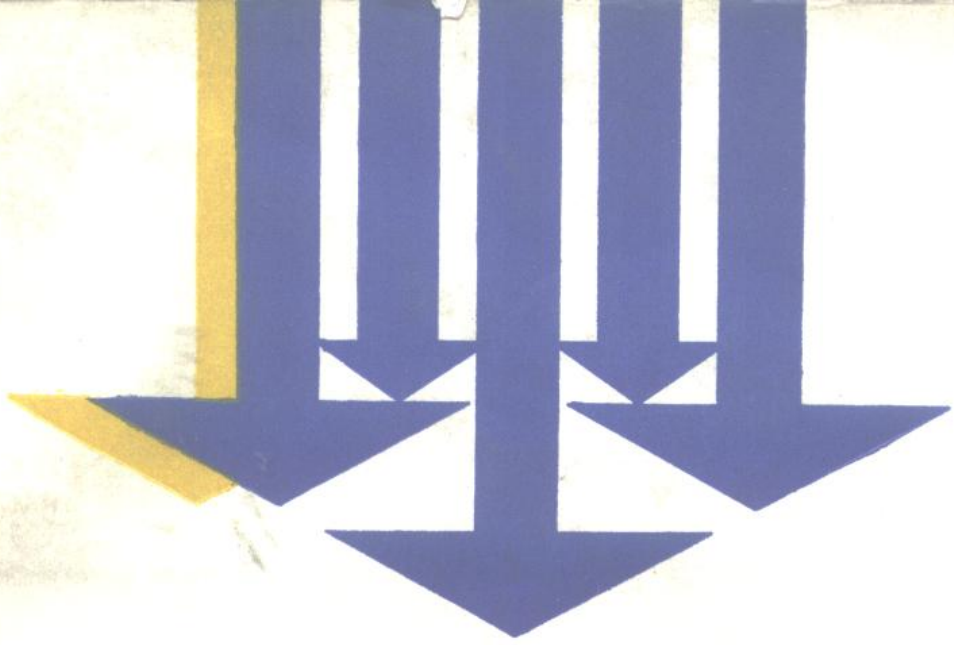


CHILL高级语言自修教程

〔荷兰〕 T.瓦尔克一费 著
张 理 邹勉力 译

人民邮电出版社

CHILL高级语言自修教程

【荷兰】T.瓦尔克一费 著
张理 邹勉力 译

人民邮电出版社

CHILL

A Self-instructional Manual

T.Valk-Fai

PITTC:

Hilversum

The Netherlands

Jauuary 1983

Philips'Telecommunicatie Industrie B.V.

内 容 提 要

CHILL是CCITT推荐的通信用高级程序设计语言。本书是CHILL的基本教程。它由浅入深详尽地探讨了CHILL的理论，又有大量典型的例题和习题，读来引人入胜，对尽快掌握CHILL并进行编程工作有很大的帮助。

本书讲述了CHILL的基本概念，包括模式定义、模式检查、动作语句、过程和程序结构，重点讲述了程序的并发执行——进程的同步和通信，最后讨论了异常和异常处理程序。本书共分三卷，各卷末尾附有习题答案。为了帮助读者全面掌握CHILL，特编写了原书中未讨论的“CHILL性能”一章作为附录。书末另附英汉名词索引。

本书可供高等院校有关专业的师生学习参考。也可供从事通信特别是程控电话交换系统设计的工程技术人员、计算机软件的工程技术人员自修。

CHILL 高级语言自修教程

〔荷兰〕T·瓦尔克—费 著

张理 邹勉力 译

*

人民邮电出版社出版

北京东长安街27号

河北省邮电印刷厂印刷

新华书店北京发行所发行

各地新华书店经售

*

开本：787×1092 1/16　　1990年 4 月　　第一版

印张：20 12/16　页数：166　　1990年 4 月河北第 1 次印刷

字数：522 千字　　印数：1—2500　　册

ISBN 7-115-04167-9/TP·049

定价：8.15 元

译者序

CHILL(CCITT HIGH LEVEL LANGUAGE)是国际电报电话咨询委员会(CCITT)在1980年及1984年Z.200建议书中推荐的高级程序设计语言。CHILL功能强、效率高，利于模块化、结构化，且有并发执行和异常处理能力，又易学易用，正迅速发展成为一种用于通信的国际标准程序设计语言。

CHILL主要是为存储程序控制(SPC)电话交换系统的编程而设计的，但也可适应通信的其它领域，在计算机应用领域也有极大的通用性。现在世界上许多电信、工业部门正纷纷使用CHILL，预计该语言将在计算机通信领域中占重要地位。我国有关部门也正在研究使用该语言，这对消化吸收引进的程控交换设备，对研制新设备，对我国通信语言的标准化和与世界通信的兼容性，均具有重大的意义。

本书是CHILL的基本教程，它由浅入深详尽地叙述了CHILL的理论，并有大量的例题和习题，读来引人入胜，对人们理解和使用CHILL有很大的帮助。

本书共分三卷。第一卷讲述CHILL的基本概念，包括基本模式、动作语句和程序结构的简述。第二卷讲述更为复杂的模式，引入模式定义和模式检查的概念，详述过程和程序结构。第三卷讲述CHILL中并发执行的程序单位——进程，着重讨论了进程的同步和通信，这是CHILL的重点。最后讲述了异常和异常处理程序。

邹勉力同志翻译了本书第一卷和第二卷1至4章。张理同志翻译了第二卷5至6章和第三卷，并编写了原书中未讨论的“CHILL性能”一章作为附录1。全书的校审、译注由张理同志负责。

已发现的原书错误，均作改正。由于译者水平有限，可能仍有差错，敬请读者指正。许多同志关心促进此书的出版，谨此致谢。

张理、邹勉力

译于邮电部第十研究所

全书简目

第一卷

第二卷

第三卷

第一卷

目　　录

前　言

本卷是CHILL自修教程三卷中的第一卷。采用如此编排方式是为了充分地讲授CHILL的基本原理。建议学生们，特别是在解答习题时，结合“CHILL用户手册”来使用本书。

本书不是一本参考手册，本书的目的是使读者通过从头到尾的阅读去获得清晰的概念。

本书尽可能地在最初阶段就开始编写程序。为此，同时考虑到对该语言的重复讲述，偶尔有必要引用尚未解释的资料，但是只在这些资料的有关特性的详细知识不影响弄懂所讨论的内容时才这么做。

每一章都包含理论部分、例子、题解及供学生们练习的习题。习题答案附在本卷的后面。需要强调的是，应该把整章习题全部做完之后，再去核对答案。如果你首先借助于“CHILL用户手册”来检查习题解答，然后再与本卷后面给出的答案做比较，将会收到更好的效果。

请注意，本书并未讨论CHILL的全部概念，仅讨论了菲力浦电信工业公司所用的CHILL中的一些概念。

省略的CHILL性能，可见附录 I 。

1. 单元和值

一个CHILL程序可分成下述三部分：

——程序中所使用的**数据**的描述；

——被执行的**动作**的描述；

——程序**结构**的描述。

对应于每一部分，都有特殊的语句，每个语句的末尾都要标以分号“；”。

本章将介绍一些数据类型和定义数据类型的方法。

1.1 什么是数据？

CHILL的可操作数据是**值**和**单元**。CHILL给出了这两种数据类型的基本区别。

值是操作处理的对象。

单元是逻辑的存储空间，值可以存储在那里，也可以从那里读取。注意，它们不是物理的存储单位！实际上，一个单元可以占用多个物理存储单位。

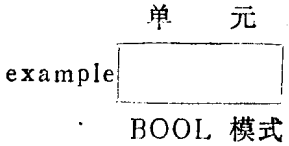
单元附有名字。通过名字可以**访问**单元。它还附有**模式**。单元的模式定义了一组特殊的性质，如单元所含值的集合，单元的大小，等等。

单元是用**单元说明语句**建立的。

例如：

DCL example BOOL;

这个语句建立了取名为example的这样一个单元，它附有模式BOOL：

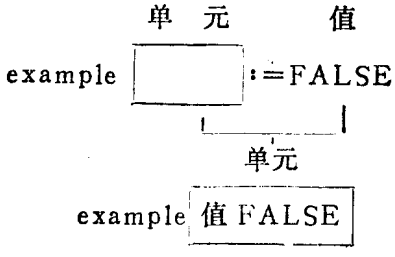


BOOL是布尔模式的预定义名字。DCL是表示说明的专用名字。

我们可在已建立的名为example的单元中存储值，但只能是布尔值。如

example:=FALSE;

这个语句就把布尔值FALSE（它与BOOL和DCL一样，也是一个专用名字）存入单元example中。



“:=”（赋值符号）这个专用符号的意思是，把其右侧的值存入其左侧指定的单元中。

1.2 标准模式INT、BOOL、CHAR

CHILL提供了数种预定义的模式。这里介绍预定义的标准模式中的三个：

—INT：整数模式的单元，可以包含由实现定义的界值之间的整数值。例如：一个32位字长的机器的正整数可以具有界值0至$2^{32}-1$之间的整数。而16位字长的机器，其正整数和负整数的自然界值是－32767和＋32767。当然，也还有定义其它不同界值的整数模式的可能性。例如：(16位机)SHORT_INT有界值－127和＋127（8位），而LONG_INT有界值$-2^{31}+1$和$2^{31}-1$（32位）。

—BOOL：布尔模式的单元，可以包含逻辑值TRUE和FALSE。

—CHAR：字符模式的单元，可以包含CCITT No.5字母表中的字符值（包括空格）。字符字面值必须用单撇号括起来。

1.2.1 整数字面值

可以用四种方法来表示整数字面值。

1)十进制表示法：

1030

1_030

342

99019

1030和1_030表示同一个值。下划线只是用来把数字分组以便阅读，而无其他意义。下划线的这种用法可以用于整数的所有表示法。如下述例子所示。

2)二进制表示法：

B'1

B'101_001

B'1011_0011

B'0

3)八进制表示法：

O'1716

O'373_307

O'0

O'2

4)十六进制表示法：

H'A17E

H'7320

H'BCEF

H'A1_BE_32_4F

1.2.2 布尔字面值

有两种布尔值：

TRUE和

FALSE

1.2.3 字符字面值

包括写在两个撇号之间的CCITT No.5字母表中的全部字符。

例如：'A'

' '

'—'

'7'

'+'

单撇号字符本身，则写作两个撇号″。同样，在组成字符字面值时，也必须将其置于两个单撇号之间，即写成''''。

1.3 名字

在CHILL中，要给出单元名字。根据名字，可以对单元进行访问。

名字必须以字母开始。这个开始的字母后面可以跟多个字母、数字、下划线，或它们的组合。但是，使用的字母和数字都应符合CCITT No.5字母表的规定。名字中的下划线与整数中的下划线不同，它是名字的有效组成部分。因此，Time_table与time table是不同的。一个名字在第一个被禁止的字符出现时就结束了，通常出现的是一个空格，但也可能是一个符号或一个换行。例如，time table被看作为两个名字，即time和table。

例：X325

richard_strauss

first_digit

state_no_325

注意！程序员不能使用本语言中的专用名字（关键字）作为名字。用户手册的附录C中，列出了全部关键字。因此，

例如：CHAR——专用名字

DCL——专用名字

B+EL——含有专用符号+

1XA3——以一个数字开始

S：23——含有专用符号：

作为名字都是非法的！

1.4 单元说明

单元说明建立指定模式的存储单元，该单元附有名字并可存放适当值。最简单的形式是

由关键字DCL，一个名字和一个模式组成。它以一个分号结束。

DCL a_letter CHAR;

这样就建立了一个叫做a_letter的存储单元，模式为CHAR的值就可以存入其中了。

在一个语句中，可以用省略方式说明具有相同模式的几个单元，如：

DCL a_letter, a_digit, no_space CHAR;

这个语句建立了名字分别为a_letter，a_digit和no_space的三个不同的存储单元。这三个存储单元都只用于字符值的存储。

也可以把不同模式的多个说明组合成一个说明语句。如：

DCL object, data CHAR, statement BOOL, number INT;

这个说明语句建立了两个存储字符值的单元object和data以及一个用于存储布尔值的单元statement，还建立了一个存储整数值的单元number。

说明语句的一般形式是：

DCL 名字表1 模式1，名字表2 模式2，…名字表n 模式n;

题解：

1.4.1 建立四个不同的单元：一个用于字符值，一个用于布尔值，两个用于整数值。

解：这两个整数值的单元，可以用一个说明语句来说明。而另外两个单元，则各用一个单独的语句来说明。

```
DCL a, b INT;
DCL C   BOOL;
DCL d   CHAR;
```

但是，也可以将上述三个语句合并为一个语句。

```
DCL a, b INT, C BOOL, d CHAR;
```

1.4.2 建立存储下列值的单元：

TRUE, 'B', B'101, O'304, H'A_17, 'H', -342。

解：这些值中，一个是布尔值——TRUE，两个是字符值('B'和'H')，其它四个则是用不同进制给出的整数值。建立这些单元的说明语句是：

```
DCL bo BOOL,
    le_1, le_2 CHAR,
    nud, nuo, nub, nuh INT;
```

1.5 初始化

说明单元时，可以用两种方法给单元赋初值：

1)DCL subscriber_no INT INIT: =1381;

2)DCL subscriber_no INT: =1381;

这两种初始化方法之间的区别将在第二卷中描述。那时，将使用第二种方法。

```
DCL subscriber_no INT:=1381;
```

等价于：

```
DCL subscriber__no INT;
    subscriber__no: =1381;
```

也可以用一个说明语句，把同一个初值赋给几个单元。

```
DCL a__letter, a__digit, no__space CHAR: ='A';
```

它等价于：

```
DCL a__letter, a__digit, no__space CHAR;
a__digit: ='A';
a__letter: ='A';
no__space: ='A';
```

建议在说明语句中尽可能采用初始化，这样可使程序更容易读，并可减少出错机会，因为确定好的初始值可保护存储单元。

归纳：

DCL	名字	模式:=值;
DCL	名字	模式 INIT:=值;

题解：

1.5.1 建立一个单元，使该单元作为一个计数器，可以存储每小时用户的呼叫次数。设呼叫次数的初始值为零。

解：该单元必须用以存储数字，因此，它必须有INT模式。为了清晰起见，该单元合适的名字应该是call coun ter（呼叫计数器）。初值为零。这样，这个单元的说明语句是：

```
DCL call__counter INT: =0;
```

1.6 只读单元

有时，在程序的执行期间，存储在某个单元中的初值绝对不可更改。为了保护这个单元，防止存放在该单元中的值被更改，必须给该单元的说明语句加上关键字READ。

```
DCL international__exchange READ BOOL: =TRUE;
```

就建立了存储布尔值的单元international__exchange，初值TRUE存入该单元中。在存储单元international__exchange的存在期间，它将始终具有TRUE值，任何要改变这一初始值的尝试都将产生错误信息。

具有只读性质的单元，都必须赋初始值。

归纳：

DCL名字 READ 模式:=值;

1.7 小结

图 I-1 给出本章讲述的数据对象的结构以及具有各种便利的单元说明语句。

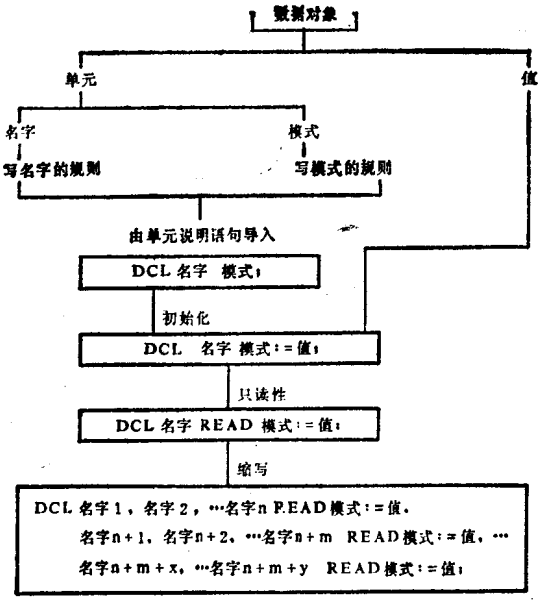


图 Ⅰ-1

1.8 练习

1.8.1 说明下列变量：

变量		性质	初始值	
number of lines	（线数）	整数，常数	982	十进制
line number	（线号）	整数	10100101	二进制
Push button set	（按键话机）	布尔量，常数	TRUE	
active	（活化）	布尔量	—	
digit	（数字）	字符	—	
index	（下标）	字符	'A'	
control channel no	（控制信道号）	整数	A881	十六进制

1.8.2 需要存储六个数字的存储单元（分别单独存储）。还需要两个具有初始值的下标号，初始值是333，其中一个是常数。请用缩写形式建立上述的这些存储单元(参见1.4)。

1.8.3 请指出下列名字表中，哪些是合法名字，哪些是非法名字。如果是非法名字，请说明理由。

a)good__boy

b)*check

c)1__student

d)no2way__traffic

e)black(dark)__colour

f)line: shape

g)ZZZZZ

h)X.X.X.X