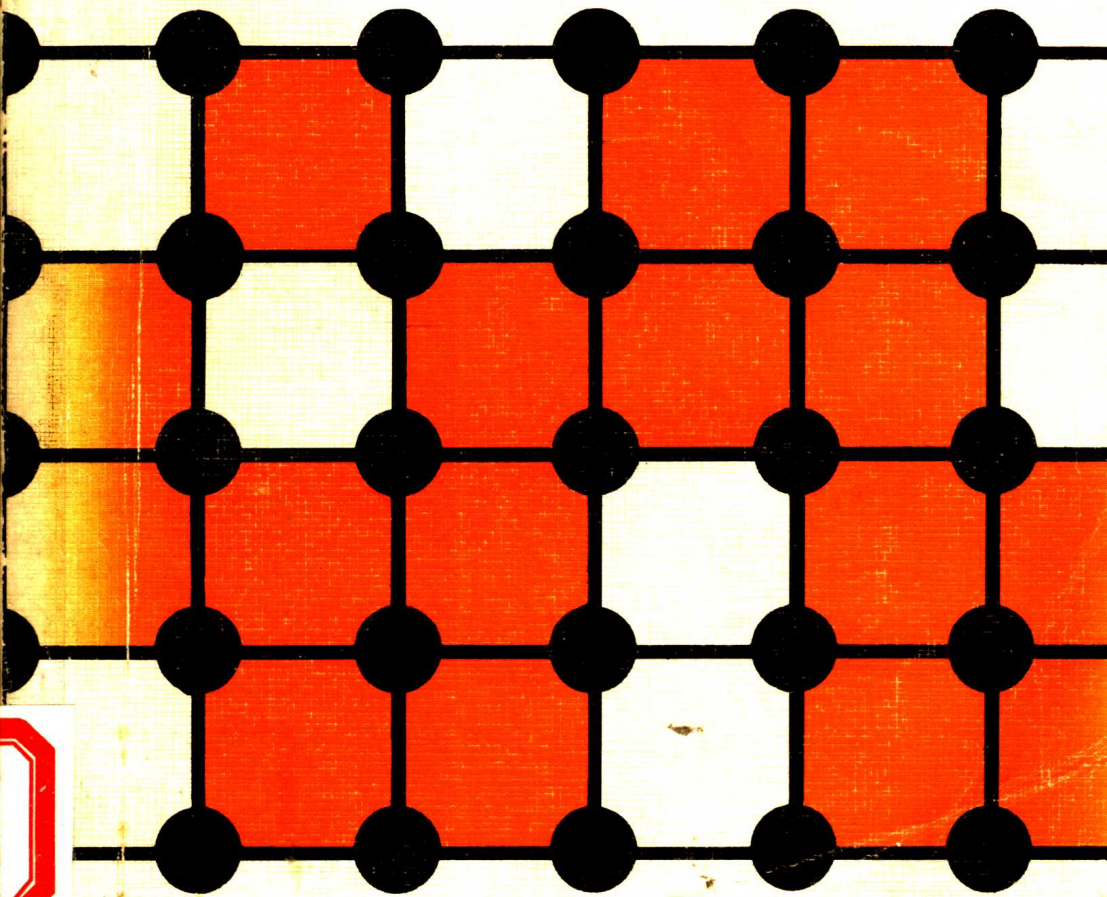


IBM

電子計算機程式語言

修訂版

王瑞材編著

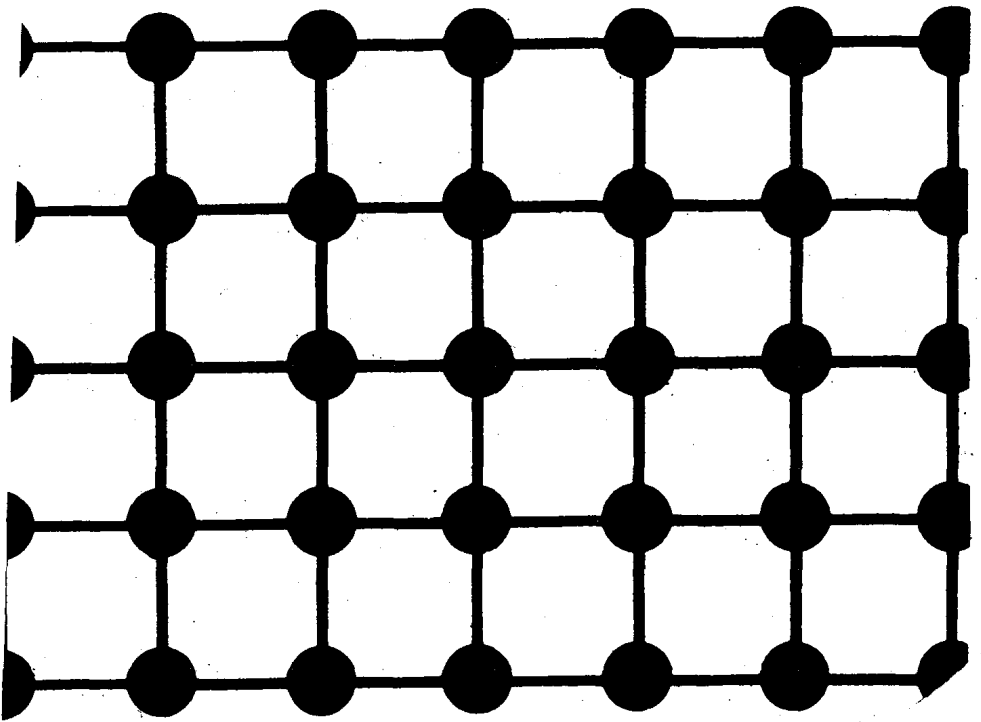


全華科技圖書公司印行

修訂版

電子計算機程式語言

王瑞材 編著 · 劉仲威 助編



全華科技圖書公司印行



全華圖書 版權所有 翻印必究
局版台業字第0223號 法律顧問：陳培豪律師

電子計算機程式語言

王瑞材 編著

出版者 全華科技圖書公司
北市建國北路85巷9號
電話：581-1300-564-1819
郵政帳號：100836

發行者 蕭 而 廊
印刷者 慶福彩色印刷廠
東南亞 港 明 書 店
總經銷 香港九龍彌敦道500號2樓
電話：3-302846-3-309095

定價 新臺幣 150 元
海外定價 港 幣 30 元
再 版 中華民國69年1月

序 言

FORTRAN原名 IBM Mathematical FORMular TRANslating System，始於 1954 年，現發展為 FORTRAN IV，是一種效率較高、用途最廣的電子計算機程式語言。它對於從事科學、工程與商業人員為一種重要的工具語言，因此目前大專院校各科系多將其列為必修或必選的課程。

編著有鑑於此，特參考十多本有關 FORTRAN 書籍與 IBM 各種教學手冊，依多年來教學經驗與不斷改進的教學方式——每一程式語法都先以一簡單的例題介紹，再詳述其規則、用法，最後再列舉一應用例題綜合說明語法的應用以及語法錯誤的結果——編輯而成，如此可免除學者對程式語言格格不入、學而又不知其用，用而不知其所以然的缺點。

本書共分八章，除詳述各種 FORTRAN 語法與特殊規則外，列舉了一百多種 FORTRAN 指述實際的用法，以及深淺漸進的 50 個完整的應用程式例題，以供有關程式設計人員工作參考，實為一本適合於教學與自習參考的書籍。

本書之編輯承本校電子計算機中心程式設計師劉仲威先生大力協助，特在此致謝。又本書所以能順利出版，承許豐妹、崔若蘭、陳燦玉三位小姐、袁承志先生及全體計算機中心同仁的整稿、打卡、校稿，特此致謝。

本書之編著，歷時三年餘，雖經多次修改與校訂，舛誤之處尚祈先進指教。

王 瑞 材 謹識

中華民國六十三年二月於台北工專

圖書之可貴 在其量也在其質

量指圖書內容充實、質指資料新穎够水準，我們就是本著這個原則，竭心盡力地為國家科學中文化努力，貢獻給您這一本全是精華的全華圖書。

目 錄

序

第一章 電子計算機簡介	1
1-1 概述	1
1-2 數字電子計算機之硬體系統與軟體系統	2
1-3 I B M 1130 電子計算機系統	11
1-4 FORTRAN IV (福傳四) 簡介	30
1-5 電子計算機運算流程	56
1-6 結論	58
習題	59
第二章 算術指述	61
2-1 概述	61
2-2 FORTRAN 常數	64
2-3 FORTRAN 變數及數型指述 (TYPE STATEMENT)	70
2-4 算術敘述式	76
2-5 算術指述	85
2-6 結論	89
習題	91
第三章 輸入與輸出指述	95

2 電子計算機程式語言

3-1	概論	95
3-2	輸入指述與輸出指述	96
3-3	輸入指述與輸出格式指述	100
3-4	輸出指述與輸出格式指述	115
3-5	結論	138
	習題	144

第四章 控制指述 151

4-1	概述	151
4-2	無條件GO TO指述	152
4-3	計值GO TO指述	155
4-4	IF指述	158
4-5	PAUSE、STOP、CALL EXIT、END指述	170
4-6	FORTRAN機器和程式指示器之測試	176
4-7	結論	187
	習題	187

第五章 庫存函數及指述函數 191

5-1	概述	191
5-2	庫存函數	194
5-3	算術指述函數	208
5-4	結論	215
	習題	215

第六章 註標變數及其輸出入指述 217

6-1	概述	217
-----	----	-----

6-2	註標變數	220
6-3	註標變數之輸入及輸出	227
6-4	DATA 指述	245
6-5	結論	251
	習題	251
第七章 DO 指述		253
7-1	概論	253
7-2	DO 指述	254
7-3	複合 DO 迴路	264
7-4	結論	278
	習題	279
第八章 副程式		283
8-1	概論	283
8-2	函數副程式	286
8-3	SUBROUTINE 副程式	298
8-4	等值指述與共同指述	316
8-5	程式之分段與CALL LINK 指述	329
8-6	結論	334
	習題	335
第九章 磁碟之資料儲存與磁碟之輸入 / 輸出指述		337
9-1	概論	337
9-2	磁碟之認識	338
9-3	RECORD 與 DATA FILE	339

4 電子計算機程式語言

9-4	DEFINE FILE 指述	341
9-5	FORTRAN 磁碟輸入 / 輸出指述	344
9-6	磁碟 FIND 指述	349
9-7	結論	351
	習題	358

第十章 FORTRAN之錯誤資料與除錯 361

10-1	概論	361
10-2	FORTRAN 編譯程式錯誤之偵測	362
10-3	FORTRAN 輸入 / 輸出錯誤資訊	372
10-4	程式編譯完成後的錯誤之偵測	376
10-5	結論	382
	習題	386

附錄 IBM 1130 DISK MONITOR系統 391

附 1	概論	391
附 2	系統磁碟	392
附 3	SUPERVISOR 與 MONITOR 控制記錄	396
附 4	磁碟使用程式及其控制記錄	401
附 5	FORTRAN 編譯程式及 FORTRAN 控制記錄	412
附 6	SUPERVISOR 控制記錄	416
附 7	結論	418

1 電子計算機簡介

1.1 概述

電子計算機俗稱電腦，一般分爲三類：類比計算機（Analog Computer）、數字計算機（Digital Computer）、綜合式計算機（Hybrid Computer）。一部電子計算機又分爲二大系統：硬體（Hardware）與軟體（Software），所謂硬體系統乃是指計算機的組成結構，而軟體系統則是指控制與操作計算機的程式計劃。

電子計算機自1946年，美國賓夕法尼亞州立大學製造問世以來，至今歷史雖短，但發展非常迅速，在短短的幾年就有四代的演變，這幾代的演進大致如下：

第一代：爲1958年以前的產品，使用真空管，記憶裝置（Memory Unit）爲磁鼓（Magnetic Drum）與低速的磁芯（Magnetic Core），儲存容量有限，計算速度以ms（千分之一秒）爲單位，電路操作慢，

2 電子計算機程式語言

約每秒 30 萬次。

第二代：為 1958 年至 1964 年的產品，使用電晶體，體積大為減小，比第一代小一半以上，其 CPU（中央處理單元）的記憶速度以 μs （百萬分之一秒）為單位。記憶單元的儲存器容量可達 32K 語句（Words），電路操作每秒約可達 100 萬次。

第三代：為 1964 年以後的產品，以微電子學（Microelectronics）為基礎，應用密集邏輯技術 SLT（Solid Logic Technology）製造的積體電路組成，電路操作速度高至 ns（Nanosecond，十億分之一秒）為計算單位，記憶速度最快可達 $62.5 \mu s$ 以上，而其體積約為第一代的五十分之一大小。

第四代：是 1970 年以後，以 LSI（大組合積體電路）所組成的，現正積極發展中。

以上這幾代的演變可以說是計算機硬體（Hardware）系統的演變，至於其軟體（Software）演變亦甚速，其中尤以應用程式（Application Program）及系統程式（System Program）為最。例如，本書所欲討論的 FORTRAN 程式語言，即由最早的 FORTRAN I 演變至現在最普遍的 FORTRAN IV。

1.2 數字電子計算機之硬體系統與軟體系統

硬體系統

數字電子計算機的硬體系統，基本結構包括三大部份：輸入單元（Input Unit）、中央處理單元（簡稱為 CPU）（Central Process Unit）以及輸出單元（Output Unit）。中央處理單元又包括三個部門：控制單元（Control Unit）、算術—邏輯單元（Arithmetic & Logical）以及記憶單元（Memory）或儲存單元（Storage）。輸

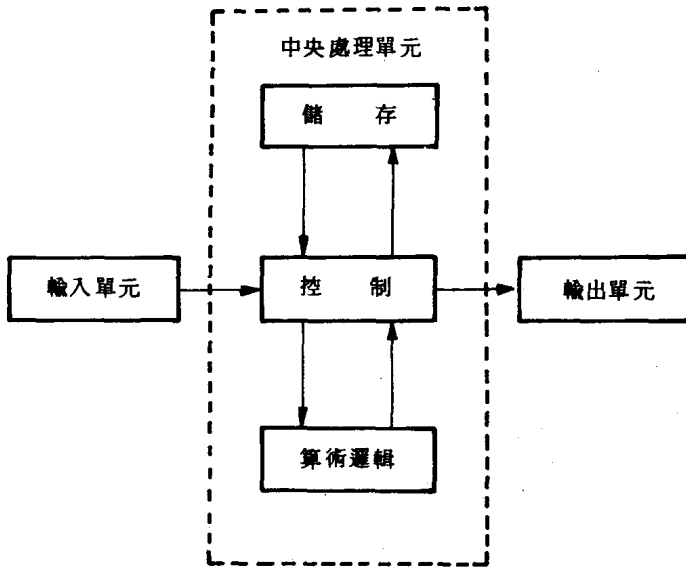


圖 1-1 數字電子計算機之基本結構

入與輸出單元可統稱為輸入/輸出單元 (I/O Unit)。圖 1-1 所示即為一數字電子計算機的基本結構。

輸入單元

此一單元的主要功用是將控制計算機執行工作的指令 (Instruction)、程式計劃 (Program) 以及需要處理的資料輸入計算機，通常這些指令或資料可以一些資料媒體如磁帶 (Magnetic Tape)、紙帶 (Paper Tape)、磁碟 (Disk)、卡片 (Card)、打字機 (Keyboard Typewriter) 以及開關 (Switch) 等輸入電子計算機。因此，此一單元又包括有磁帶機、讀紙帶機、磁碟裝置與讀卡機等等，其中以讀卡機最為常用。

4 電子計算機程式語言

控制單元

控制單元的主要功用是控制並指揮整個計算機的作業。當一系列的指令（我們可稱為程式）輸入計算機時，控制單元就依輸入程式計劃組合作業的順序，訂定指令並發出適當的訊號，指令有關部門執行工作。

儲存單元

儲存單元主要的功用是儲存資料、指令與程式計劃，它分為兩大部門：一為主儲存器（Main-Storage），又稱為磁芯儲存器（Core Storage），一為輔助儲存器（Auxiliary Storage）；輔助記憶單元常用的為磁碟（Disk）或磁帶（Magnetic Tape）；主記憶單元又分為許多記憶位置（Location），每一位置被定以一號碼，稱為地址（Address）以備受控制單元指令儲存資料與指令。儲存單元的儲存容量以語句（Words）為單位，在 IBM 1130 計算機中每一語句由 16 個“筆”（Bit）所組成，每 1 bit 由一磁芯儲存。每 1024 個語句，以 1 K 語句表示，因此一般計算機的大小常以 4 K、8 K、12 K、16 K、24 K 與 32 K、64 K 等等表示之，其中 8 K 亦即表示 8192 個語句。

算術與邏輯單元

算術與邏輯單元主要功用為執行加（Add）、減（Subtract）乘（Multiply）、除（Divide）、比較（Compare）以及邏輯決定（Logic Decision）。此單元又包括二部門：一為累加器（Accumulator）以及記錄器（Register）。

輸出單元

輸出單元的功用為將計算機處理的結果輸出於計算機的資料媒體。可用的資料媒體有報表紙、卡片、紙帶、磁帶或磁碟。因此，此單元又分為印字機（Printer）、打卡機（Card-Puncher）、打字機（Typewriter）、磁碟裝置（Disk）等等。

以上所述是一部數字式電子計算機的基本結構，而其更詳細的結構如圖 1-2。圖中所示當一個程式計劃與資料經由計算機控制部門的指示，由某一輸入單元讀入，並儲存於記憶單元。控制單元一般包括兩部份，一為指令譯碼器（Instruction Decoder），一為控制產生器（Control Generator）。控制產生器控制算術單元的資料處理過程以及提供一資訊於輸入輸出選擇器（In-Out Selector）以控制計算機的輸入與輸出，指令譯碼器則依一連續的方式操作，當一指令被傳至指令譯碼器，指令就在此被翻譯完成，再送入控制產生器，控制產生器用此譯好資料來決定將信號送入算術單元計算或送到輸入—輸出選擇器，使輸入—輸出部門工作。控制產生器亦送一控制訊號到計算機 CPU 的儲存單元（記憶單元）內，這種訊號是用來選擇儲存單元內供給算術單元操作時所需的儲存位置的號碼與資料。同時，控制產生器亦送一訊號至算術單元，以指令算術單元於完成計算工作後將資料傳回計算機儲存單元儲存。因此由於控制產生器專司資訊輸入與輸出記憶單元的轉換工作，對於一儲存單元（或記憶單元）的內部組織應進一步的了解。記憶單元由許許多多的小單元組成，我們可將它看成一組郵政信箱，每一信箱（或者記憶小單元）具有一特定的地址（Address），用以區劃其他的記憶小單元（有許多種不同的名稱：記憶地址、記憶位置與儲存位置），在這一個記憶小單元內，可儲存一數據資料或一個指令，因此如果記憶小單元中儲存的是數據資料，此資料將由計算機的算術單元來處理，如果儲存的是指令則由控制單元取出處理。舉個例子來說，假使要計算機將儲存單元中的某兩個記

6 電子計算機程式語言

憶小單元所存的數加在一起然後再存回儲存單元中的另一記憶小單元中，則計算機的控制產生器將促使這兩個數移至到算術單元中執行相加，然後再指令將結果存回儲存單元中另一個記憶小單元中。當計算機執行上述工作時，前二個資料儲存單元的記憶位置存入算術單元時，在儲存單元內的資料並不被摧毀，仍然保留以備以後所需，因此這一個作用我們可稱為從記憶單元中“讀出”(Read Out)資料。而計算的結果，被存回於儲存單元的另一個記憶位置中，則不論任何情況，原先存於該記憶單元中的資料或資訊都被摧毀。

計算機在計算兩數的總和時，計算機需使用到算術單元的一種記錄器(Register)，這種記錄器是一種特別的記憶裝置(Memory cell)，通常稱為累加器(Accumulator)，這種累加器與桌上計算機(Calculator)很相似。另外計算機在執行工作時，還有兩個記錄器，一是指令記錄器，它是包含在指令譯碼器內，指令記錄器存有在任何

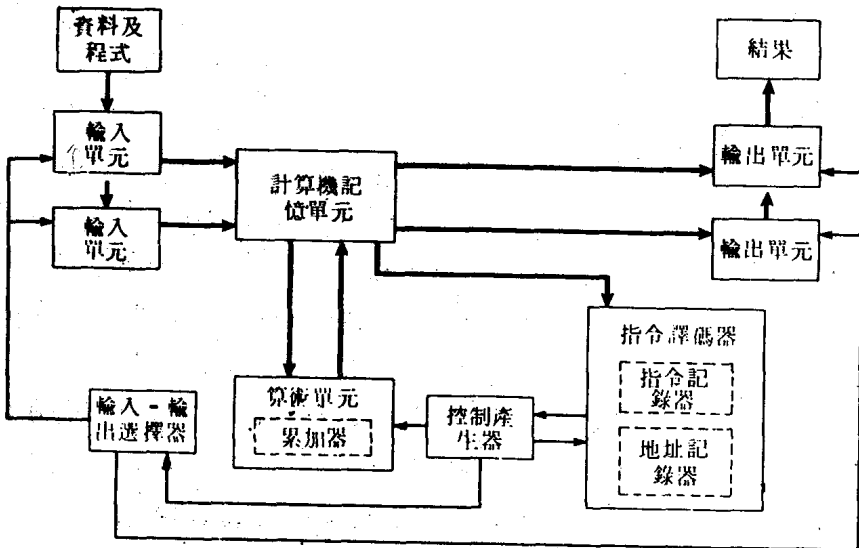


圖 1-2 較詳細的計算機硬體結構

時間均可執行的指令。另外一種是地址記錄器 (Current - Address Register)，也是包含在指令譯碼器內，它的作用在處理記憶器內的指令過程時，能與計算機取得連繫。由於這種地址記錄器的作用可找出下一個從記憶單元傳送至指令譯碼器的指令。

計算機執行上述計算結果，如果需要輸出的話，計算機將由控制單元發出指令，將存於儲存單元的資料結果“讀出”於輸出單元，輸出單元受控制產生器輸出經“輸出—輸入”選擇器的控制訊號而將結果印出於資料媒介體上。

計算機之軟體系統

計算機之軟體系統可分為應用程式 (Application Program) 與系統程式 (System Program)。應用程式是利用各種計算機程式語言組成的一種指令計算機執行某種工程上或商業上的應用之問題的程式計劃，這些程式語言，可分為機器語言 (Machine Language)、符號語言 (Symbolic Language) 以及高階語言 (High Level) 等三種語言，高階語言又分很多種，FORTRAN 即是其中之一。系統程式又分兩組，一組為翻譯程式 (Translator)，另一組為作業系統 (Operating System)，現將軟體系統分類簡述如下：

機器語言

機器語言乃是一連串的數字系統的組合來表示數據或指令的程式語言。

例如：

$$(1) A = B / (C + D)$$

此方程式如以機器語言寫成如下：

1100010000000000 0001010110111010.....Load

8 電子計算機程式語言

```
1000010000000000 0001010101011000 ..... Add
1101001110000000 ..... Store
1100010000000000 0001010111011100 ..... Load
0001100010010000 ..... Shift
1010101110000000 ..... Divide
1101010000000000 0001010111101110 ..... Store
```

計算機執行此指令的計算步驟如下：（設A、B、C及D各變數皆為整數）

- (1) 存入C值於累加器內。
- (2) 將D值加於累加器內所存之值。
- (3) 將累加器內此時所存之值存入一暫時記憶位置。
- (4) 存入B值於累加器內。
- (5) 移累加器以待進行除法。
- (6) 將累加器內之值除以暫存於暫時記憶位置內之值。
- (7) 儲存此時累加器之值於資料變數 (Data Variable) A。

符號語言

由於數字系統中一連串的“1”與“0”不易直接顯示指令的意義，應用程式，若以機器語言寫成則相當地麻煩，並且容易發生錯誤，因此在軟體系統發展一符號語言來寫。符號語言乃是使用一些有意義的符號來代表一指令，例如計算上述方程式(1)，使用符號語言的寫法如下：

```
LD    L    C    ..... Load
A     L    D    ..... Add
STO   3    -126 ..... Store
LD    L    B    ..... Load
```