

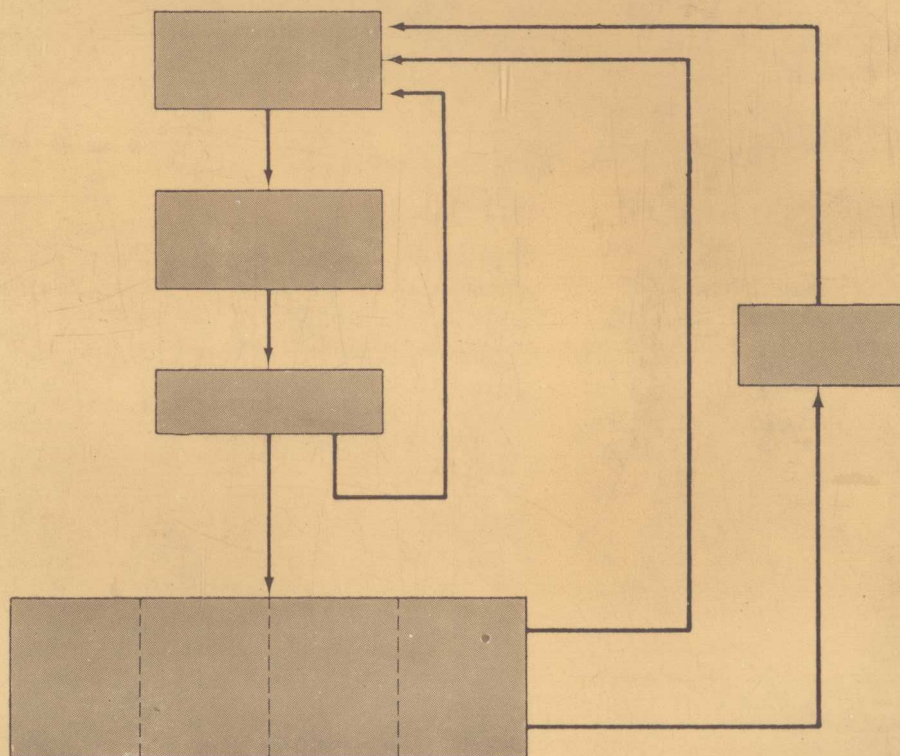
國家科學叢書

計算機系統結構

(第二版)

M. Morris Mano 著

婁子正 譯



COMPUTER SYSTEM
ARCHITECTURE

計算機系統結構

原著者：M. Morris Mano

譯者：婁子正

國家書店有限公司印行

有著作權
不准翻印

計算機系統結構

定價：新台幣叁佰貳拾元整

原著者：M. Morris Mano

譯者：婁子正

總策劃：林洋慈

發行人：林大坤

發行者：國家出版社

總經銷：國家書店有限公司

郵撥：一〇四八〇一帳戶

公司：台北市新生南路一段126之8號三樓

電話：3912425・3926748-9

印刷所：東遠印刷廠

1983年7月初版

行政院新聞局局版台業字第零陸叁貳號

序 文

計算機結構與研究計算機系統硬體設計的計算機工程師及研究軟硬體有關之系統設計的計算機科學家們息息相關。計算機系統為同時包括硬體與軟體的系統。本書主要討論計算機系統的**硬體**方面，但是，與計算機結構有關的軟體部分不致忽略。

通常定義計算機結構，都會加入那些程式師所感興趣之計算機附屬品。此處，我們將那些專業計算機設計師所應該知道的事情考慮進去以定義計算機結構。計算機設計師必然熟悉那造好之計算機的基本硬體藍圖。他們必須具備計算機系統之結構與動作的知識，以及設計它們的方法。因此，如本書中所定義之計算機結構便與數位計算機系統之結構構造及硬體設計有關。

特殊處理機，包括其暫存器在內之實際結構、資料流程、微運算與控制函數，最好利用**暫存器傳送語言**之符號來加以描述。諸如此類之語言，在本書中加以研討，及其與數位計算機之硬體結構及設計的關係，均予以詳細的說明。暫存器傳送語言在許多場合下，均以簡明扼要的方法用來說明各種計算機運算。

本書的計劃為首先敘述比較簡單之內容，然後，介紹比較深的主題。前面六章涵蓋一部簡單數位計算機之計算機結構、設計與程式計劃之基本認識所需的內容。後面六章敘述數位計算機之獨立功能單元，並強調前面部分所未涵蓋之較深主題。

第二版的修改是在後面六章，其中，第七章，修改中央處理單元部分，第十一章，輸入輸出構造全部重寫，第八章，修改微程式控制部分，與第十二章，就記憶器構造的部分加以修改並加入新的內容。其他幾章除了某

些小地方重新組織之外，根本上仍與第一版相同。

第一章介紹當數位系統使用個別閘及正反器來製造時，數位系統之設計所需的基本知識。它涵蓋布耳代數，組合電路與順序電路。爲了使本書保有適當的回響，所以，有必要將這些主題之討論限制於專門介紹該主題之章節中。證實這種劃分方式是正確之理由爲當積體電路之使用取代個別閘與正反器時，數位計算機的設計則採取另一種不同的角度。第一章所包含的內容可作爲認識目前數位系統所需的基礎。

第二章首先列舉積體電路的一般特性，它詳細的涵蓋一些最基本的數位電路，例如：暫存器、計數器、解碼器、多工器、隨機出入記憶體與僅讀記憶體。這些數位電路在後面幾章中作爲較大單元之設計的建築方塊使用。

第三章敘述數位計算機中常見之各種資料型式，並說明它們如何以二進型式表示於計算機暫存器中。本書所要強調的是算術計算中所使用之數的表示法；及符號之二進碼，例如，資料處理中所使用之字母文字，以及其他特殊應用中所使用之獨立符號。

第四章定義暫存器傳送語言，並說明它如何以符號格式被用來表示數位計算機之暫存器間的微運算順序。符號被定義是供算術邏輯與移位微運算以及起動微運算之控制函數之用。本章以較長之篇幅來描述是爲了說明與各種符號以及暫存器傳送語言有關之硬體的含義。

第五章敘述小型基本數位計算機的構造與設計。定義計算機之暫存器並說明計算機指令羣。計算機之描述是以暫存器傳送陳述語作正式之描述，其中，暫存器傳送陳述語表示暫存器之間的微運算，以及起動微運算的控制函數。因此，很明顯可以看出微運算羣能夠被用來設計計算機的資料處理機部分。暫存器傳送陳述語表中之控制函數，提供我們設計控制單元所需之資訊。

第六章使用第五章中所定義之基本計算機的 25 個指令來描述程式計劃計算機經常使用之許多技巧。在符號碼中之微程式範例主要在於敘述許

多基本資料處理的工作。二進程式，符號碼程式與高階語言程式間的關係，均以範例加以說明。由這裏進而了解翻譯程式的需要，如組合程式與編譯程式。組合程式之基本操作連同其他系統程式也一併加以敘述。本章之目的在於介紹計算機軟體的基本概念，但是，並不涉及較深的細節。軟體原理與硬體介紹的知識，應該能夠給予讀者整個計算機系統的整體概念，包括硬體與軟體在內。

第七章討論數位計算機的**中央處理機單元**(CPU)。敘述匯流排構造之處理機與特殊**算術邏輯單元**(ALU)之設計。記憶堆疊的構造以描述它的一些應用來加以說明。各種指令格式連同其定址模式也一併加以描述。在一般計算機中最常見的一些指令以說明其功能的方式而加以列舉。然後介紹被封入一個積體電路包裝中之CPU的微處理機，及其內部與外部之特性分析。本章以一節並行處理與管道處理作總結。

第八章介紹**微程式計劃**的觀念。藉舉例如何產生典型計算機指令羣來說明特殊控制單元的發展。研究微程式順序機以說明**位元分割**型式之LSI元件設計之程序。最後一節討論微程式計劃的優點與應用。

第九章討論**算術處理機**的設計。敘述以帶正負號大小表示之二進整數的加法、減法、乘法與除法。使用暫存器傳送語言來設計算術處理機。算術處理機的構造圖與第五章所設計的計算機相同。然後，定義一個二進計數器，並用它來說明算術運算能夠加以微程式計劃的方法。

第十章敘述其他算術演算法。其中討論帶正負號的補數之二進資料、浮點資料與十進資料的演算法。這些演算法均利用流程圖來敘述，而這些流程圖使用暫存器傳送語言來表示完成這些演算法所需之微運算順序與控制決定。

第十一章說明一些輸入與輸出裝置常用的功能。處理機與I/O裝置間界面的需求，均加以說明，並列舉I/O傳送的各種構造圖。這些包括**非同步傳送**、**直接記憶體出入**與**優先間斷**在內。其他所涵蓋的主題為輸出入處理機、資料通信處理機與多元處理機系統構造。

第十二章介紹記憶器結構的概念，包括**隱藏記憶器**、**主記憶器**與諸如**磁碟**、**磁帶的輔助記憶器**在內。有關之記憶器的內部構造與外部操作均有詳細之說明。記憶器管理之觀念透過**隱藏記憶器**與**虛擬記憶器**系統之硬體需要的陳述也都加以介紹。

每一章都加入一些問題與一張參考資料表。其中有些問題為該章所涵蓋之內容，是作為練習之用。其他的問題為更深一層的性質，或企圖作為解決與數位計算機硬體設計方面有關問題的一些練習。教師所需之解答手冊，可向出版社索取。

本書適用於電機工程、計算機工程或計算機科學等系之計算機結構課程。本書之部分研讀有下列幾種方式：(1)作為計算機硬體構造的第一門課程，可涵蓋第一章到第五章，並由第七、八、九章中加入一些教師認為合適的內容。(2)作為計算機設計的課程，首先複習第五章，作為數位邏輯設計的預備知識，然後，涵蓋第七章到第十二章。(3)作為計算機硬體系統之課程，涵蓋數位計算機的五個基本單元：**處理機**（第七章）、**控制**（第八章）、**算術**（第十章）、**輸出入**（第十一章）與**記憶器**（第十二章）。本書也適合計算機工程師與需要獲得計算機硬體結構之基本知識的科學家自修使用。

M. Morris Mano

目 錄

第一章 數位邏輯電路

1-1	邏輯閘.....	1
1-2	布耳代數.....	5
1-3	映像簡化.....	9
1-4	組合電路.....	18
1-5	正反器.....	25
1-6	順序電路.....	31
1-7	結論摘要.....	41
	參考資料.....	42
	問題.....	43

第二章 積體電路與數位函數

2-1	數位積體電路.....	47
2-2	IC正反器與暫存器.....	55
2-3	解碼器與多工器.....	60
2-4	二進計數器.....	65
2-5	移位暫存器.....	71
2-6	隨機出入記憶體 (RAM).....	74
2-7	僅讀記憶體 (ROM).....	83
	參考資料.....	86
	問題.....	87

第三章 資料表示法

3-1	資料型式	91
3-2	定點表示法	99
3-3	浮點表示法	108
3-4	其他二進碼	111
3-5	偵錯碼	115
	參考資料	117
	問題	118

第四章 暫存器傳送與微運算

4-1	暫存器傳送語言	123
4-2	暫存器之間的傳送	124
4-3	算術微運算	136
4-4	邏輯微運算	142
4-5	移位微運算	152
4-6	控制函數	154
	參考資料	160
	問題	161

第五章 基本計算機的構造與設計

5-1	指令碼	167
5-2	計算機指令	170
5-3	計時與控制	176
5-4	指令的執行	181
5-5	輸出入與間斷	190
5-6	計算機的設計	196

5 - 7	結論摘要.....	202
	參考資料.....	203
	問題.....	204

第六章 計算機軟體

6 - 1	介紹.....	209
6 - 2	程式計劃語言.....	211
6 - 3	組合語言.....	215
6 - 4	組合程式.....	221
6 - 5	程式環路.....	227
6 - 6	程式計劃之算術與邏輯運算.....	231
6 - 7	次常式.....	237
6 - 8	輸出入程式計劃.....	244
6 - 9	系統軟體.....	249
	參考資料.....	258
	問題.....	259

第七章 中央處理機的構造

7 - 1	處理機匯流排的構造.....	265
7 - 2	算術邏輯單元 (ALU).....	268
7 - 3	堆疊構造.....	278
7 - 4	指令格式.....	287
7 - 5	定址模式.....	293
7 - 6	資料傳送與處理.....	301
7 - 7	程式控制.....	309
7 - 8	微處理機的構造.....	321
7 - 9	並行處理.....	333

參考資料	344
問題	346

第八章 微程式控制的構造

8-1 控制記憶器	355
8-2 位址順序	357
8-3 微程式範例	364
8-4 微程式順序機	374
8-5 微指令格式	380
8-6 軟體輔助	387
8-7 優點與應用	389
參考資料	392
問題	394

第九章 算術處理機的設計

9-1 介紹	399
9-2 不帶正負號二進數的比較與減法	400
9-3 加法與減法演算法	406
9-4 乘法演算法	410
9-5 除法演算法	413
9-6 處理機的構造圖	420
9-7 控制的設計	424
9-8 微程式計劃計算器	427
參考資料	437
問題	438

第十章 算術演算法

1 0 - 1	介紹.....	443
1 0 - 2	帶正負號 2 的補數之算術.....	445
1 0 - 3	乘法與除法.....	449
1 0 - 4	浮點算術運算.....	456
1 0 - 5	十進算術單元.....	469
1 0 - 6	十進算術運算.....	474
	參考資料.....	481
	問題.....	483

第十一章 輸出入的構造

1 1 - 1	周邊裝置.....	489
1 1 - 2	輸入 / 輸出界面.....	493
1 1 - 3	非同步資料傳送.....	503
1 1 - 4	直接記憶器出入 (DMA).....	518
1 1 - 5	優先間斷.....	525
1 1 - 6	輸出入處理機 (IOP).....	536
1 1 - 7	多元處理機系統的構造.....	549
1 1 - 8	資料通信處理機.....	559
	參考資料.....	572
	問題.....	574

第十二章 記憶器的構造

1 2 - 1	輔助記憶器.....	581
1 2 - 2	微計算機記憶器.....	586
1 2 - 3	記憶器的層次.....	590
1 2 - 4	結合記憶器.....	594
1 2 - 5	虛擬記憶器.....	601

1 2 - 6	隱藏記憶器.....	609
1 2 - 7	記憶管理的硬體.....	619
	參考資料.....	628
	問題.....	630
	索引.....	637

數位邏輯電路

(Digital Logic Circuits)



1-1 邏輯閘 (LOGIC GATES)

數位計算機，顧名思義，為執行各種計算工作的數位系統。計算機中**數位 (digital)** 一詞含有用變數來表示資訊的意義，而這些變數使用為數有限的**分立值或定值**。這些為元件內部所處理的值能夠保持有限數目的分立狀態。舉例而言，十進數位 0, 1, 2, ..., 9 提供 10 個分立值。事實上，若僅採用兩種狀態，則數位計算機的功能更加可靠。因受元件實際上的限制與人類邏輯對二進制（亦即，**真或假，是或否的陳述**）的趨向，所以，原本即受限於使用分立值的數位元件，更進一步被迫僅能使用兩個值，這就叫做**二進制 (binary)**。

數位計算機使用二進數系統，此系統有兩個數字：0 與 1。每個二進數字叫做**位元 (bit)**。數位計算機中的資訊是用位元組來表示。使用不同的編碼技巧，位元組不僅能夠用來表示二進數，而且能夠表示其他任何的分立符號，例如，十進數位或字母文字。適當的使用二進排列與不同的編碼技巧，二進數位或位元組能被用來發展完整的指令羣，以便執行不同型式的計算。

普通的十進數使用以 10 為底的系統，與其相對照，二進數使用以 2 為底的系統。舉例而言，二進數 101101 表示一定量，若將每個位元乘上一個以 2 為底的整數乘幂，則這值能被轉換成十進數，其表示如下：

$$1 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 = 45$$

2 計算機系統結構

101101 這六個位元表示一個二進數，其十進數等值為 45。然而，在特定的數位系統中，6 位元組也能夠以一個二進碼表示一個文字字母，或以一個控制碼來說明一些決定邏輯。易言之，數位計算機中位元組能夠用來表示許多不同的事物。這與字母中許多相同的字被用來構成許多不同的語言（例如英語和法語）的觀念是一樣的。

在數位系統中二進資訊是以稱為信號（signals）的實際值來表示電的信號，例如在整個數位系統中電壓存在於兩個可辨值之中的任一值上及表示二進變數等於 1 或 0。舉例而言，一特定數位系統能夠將 3V 的信號視為 1 使用，而將 0.5V 的信號視為 0 使用。如圖 1-1 所示，每個二進值對於公認值都有一可接受的偏差值。介於兩容許區之間的中間區域只有在狀態改變時才能夠通過。數位電路的輸入端的二進信號響應也是在指定的容許值之內。

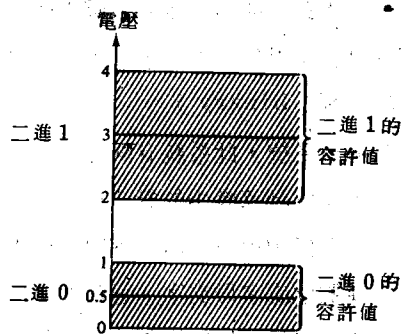


圖 1-1 二進信號的例子

二進邏輯處理二進變數與帶有邏輯意義的運算。它是以代數或表格的形式來描述二進資訊的使用與處理過程。二進資訊的處理是用稱為閘（gates）的邏輯電路所完成。閘是在輸入邏輯需要被滿足時產生二進制 1 或 0 的信號的硬體方塊。不同的邏輯閘在數位計算機系統中是很常見的。每個閘都有一個獨特的圖示符號，而且它的運算能夠用代數函數表示。每

個開之二進變數的輸入 - 輸出關係能夠用表格的型式表示於**真值表**(truth table)中。

八個邏輯開的名稱、圖示符號、代數函數與真值表，列於圖 1-2 中。每個開有一個或兩個二進輸入變數，以 A 和 B 來表示，與一個二進輸出變數，以 x 來表示。AND 開產生 AND 的邏輯接面，亦即：若輸入 A 與輸入 B 均為二進制 1 時，則二進輸出為 1；否則，輸出為 0。這些情形也都在 AND 開的真值表中加以說明。這表說明僅當輸入 A 與輸入 B 均為 1 時，輸出 x 為 1。AND 函數的代數運算符號與一般算術乘法符號相同。我們能夠用一個點介於這些變數之間或在變數之間不用運算符號而連結這些變數。AND 開可擁有兩個以上的輸入，且根據定義，若且唯若所有輸入為 1 時，則輸出為 1。

OR 開產生本含 OR 函數，亦即：若輸入 A 或輸入 B ，或是兩個輸入均為 1 時，輸出為 1，否則，輸出為 0。OR 函數的代數符號為 $+$ ，類似於算術加法的符號。OR 開可擁有兩個以上的輸入，且根據定義，若且唯若任一輸入為 1，則輸出為 1。

反相器 (inverter) 電路顛倒二進信號的邏輯意義。它產生 NOT 或補數 (complement) 的函數。用來表示邏輯補數的代數符號不是分號就是跨於變數符號之上的橫線。本書使用分號來表示二進變數的邏輯補數，然而，跨於字母上的橫線則保留用來表示在第四章中所定義的補數微運算。

反相器圖示符號輸出上的小圓圈表示邏輯補數。三角形符號本身表示緩衝器 (buffer) 電路。緩衝器並未產生任何特殊邏輯函數，因為其輸出二進值與輸入的二進值相同。這種電路僅被用來放大信號。舉例而言，緩衝器將 3V 視為二進制 1，當它的輸入為 3V 時，將會產生 3V 的輸出。然而，輸入電流的供給遠比輸出所產生的電流要低。用這種方法，緩衝器能夠推動其他許多需要大量電流的開，否則，取自接到緩衝器輸入端的少量電流，將無法正常工作。

NAND 函數是 AND 的補數，若以圖示符號表示，它是由一個 AND

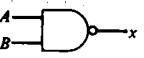
名 稱	圖示符號	代數函數	真 值 表															
AND		$x = A \cdot B$ 或 $x = AB$	<table> <tr><th>A</th><th>B</th><th>x</th></tr> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </table>	A	B	x	0	0	0	0	1	0	1	0	0	1	1	1
A	B	x																
0	0	0																
0	1	0																
1	0	0																
1	1	1																
OR		$x = A + B$	<table> <tr><th>A</th><th>B</th><th>x</th></tr> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </table>	A	B	x	0	0	0	0	1	1	1	0	1	1	1	1
A	B	x																
0	0	0																
0	1	1																
1	0	1																
1	1	1																
反相器		$x = A'$	<table> <tr><th>A</th><th>x</th></tr> <tr><td>0</td><td>1</td></tr> <tr><td>1</td><td>0</td></tr> </table>	A	x	0	1	1	0									
A	x																	
0	1																	
1	0																	
緩衝器		$x = A$	<table> <tr><th>A</th><th>x</th></tr> <tr><td>0</td><td>0</td></tr> <tr><td>1</td><td>1</td></tr> </table>	A	x	0	0	1	1									
A	x																	
0	0																	
1	1																	
NAND		$x = (AB)'$	<table> <tr><th>A</th><th>B</th><th>x</th></tr> <tr><td>0</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>0</td></tr> </table>	A	B	x	0	0	1	0	1	1	1	0	1	1	1	0
A	B	x																
0	0	1																
0	1	1																
1	0	1																
1	1	0																
NOR		$x = (A + B)'$	<table> <tr><th>A</th><th>B</th><th>x</th></tr> <tr><td>0</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>0</td></tr> </table>	A	B	x	0	0	1	0	1	0	1	0	0	1	1	0
A	B	x																
0	0	1																
0	1	0																
1	0	0																
1	1	0																
OR (XOR)		$x = A \oplus B$ 或 $x = A'B + AB'$	<table> <tr><th>A</th><th>B</th><th>x</th></tr> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>0</td></tr> </table>	A	B	x	0	0	0	0	1	1	1	0	1	1	1	0
A	B	x																
0	0	0																
0	1	1																
1	0	1																
1	1	0																
NOR 或等值		$x = A \odot B$ 或 $x = A'B' + AB$	<table> <tr><th>A</th><th>B</th><th>x</th></tr> <tr><td>0</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </table>	A	B	x	0	0	1	0	1	0	1	0	0	1	1	1
A	B	x																
0	0	1																
0	1	0																
1	0	0																
1	1	1																

圖 1-2 數位邏輯閘