

電腦工程學

——硬體概論

洪仁正編譯



電腦工程學

——硬體概論

中華民國七十七年十二月初版

編輯者：洪 仁 正

發行人：高 本 釗

發行及印刷行：渤海堂文化事業公司

臺北市羅斯福路一段20號8樓之1

登記證局版臺業字第3694號

郵撥帳號：1094926-2

電話：3 9 2 8 5 1 6

平裝一冊 定價420元

序

本書是一本適用於講授數位系統、數位設計、計算機結構的範本，它涵蓋了數位系統與邏輯設計的所有基本定理，並且對微處理機與以微處理機為基礎的系統提供了介紹的說明。由於微處理機的出現及其漸趨的重要性，使得將這些用途廣泛的元件，儘早地在工程或電腦科學的課程中加以介紹變得更形重要。有關各章特色如下：

- 第一章的主題是邏輯變數與布林代數。它對邏輯閘與邏輯的連結加以描述與分析。在該處我們主要在介紹二進制系統。
- 第二章處理邏輯函數的標準式與卡諾圖。
- 第三章討論基本組合電路，包括解碼器，編碼器，碼轉換器，多工器與解多工器。我們著重於此乃是因為所有這些元件都是在市面上可買得到的積體電路晶片，並且也簡略的討論積體電路的族系。其中也解釋在習慣上我們處理晶片控制端點的特性之方式，而這個討論導致於混合邏輯 (mixed logic) 的規則之討論。在某些運用上，混合邏輯日益受到歡迎。目前逐漸被引入來表示組合邏輯與其他元件的新邏輯符號之例子也介紹了一些。
- 第四章中也詳盡的說明基本儲存元件—正反器，並且小心地區分電栓與正反器的不同。在同位系統中，一個正反器欲要正常的運作所需要的特性也加以討論。將正反器組合成儲存暫存器、移位暫存器與計數器也被提出來研究。再一次地，在此章中所強調的是，所有描述的元件都是可以買到的積體晶片，同時我們也舉了一些這種裝置的例子。
- 第五章是討論算術運算這個主題，而且以加法為主。另外，我們對前瞻進位 (look-ahead carry) 也有說明與分析。

2 電腦工程學（硬體概論）

- 第六章的主題是記憶體。此章包含靜態與動態的 RAM, ROM, PLA, 串聯記憶體以及大容量儲存的記憶體。同時也討論到從記憶體中讀出與寫入的計時（timing）考慮。
- 第七章中介紹同步與基本模式的順序系統之分析與設計，並說明流程圖，狀態圖與表，也描述了消除多餘狀態的程序。
- 第八章中我們提供了控制器的題材。微處理機是由許多的儲存器。工作暫存器，ALU 與控制器組合而成。控制器擁有超人的能力，它能在正確的時間內做正確的事情，並精確地按照正確的順序，當完成一個動作之後，能準確無誤地繼續下一個動作。的確，控制器只是一個特定目的的順序電路，它並沒有包含什麼艱深的觀念。然而，對一位初學者而言，控制器的含糊性無可避免的將是困難的泉源。當沒有具體與特定的例子當作模式時，我們很難接受一般性的控制器描述法。第八章主要希望能提供非初學者一些中肯意見。
- 第九章也是以微處理機為著眼而寫成的，在此處呈現了一個簡單計算機（16指令）的架構。這個結構對日後在更複雜系統中將遭遇到的指令型態提供了一個預先的瀏覽。跳躍（Jump）與次常式呼叫（Subroutine Call）指令在此被提到，同時也用組合語言寫了一些簡單的程式。在此章中也提到以微程式（Microprogramming）控制的主題及一些簡單的例子。
- 第十章中我們選用廣為人知與利用，且深受好評的微處理機 8080（Intel 公司出品）來學習。雖然 8080 已被修改為 8085，但是我們仍然採用 8080，因為它較簡單，也因此適合於介紹性的說明。第十章的主題即是 8080 的架構，指令與程式。

最後，祈本書之出版也能對想一窺數位系統堂奧的初學者有些助益。同時，也要對寫作期間的一些師友提供不少意見深表謝意，並希各方賢達先知能不吝提正賜教。

目 錄

序

第 1 章 邏輯變數的代數	1
1.1 變數與函數	1
1.2 邏輯變數	2
1.3 邏輯變數的值	3
1.4 單一邏輯變數的函數	4
1.5 兩邏輯變數函數	4
1.6 或函數	7
1.7 邏輯系統的實現	8
1.8 邏輯變數的電壓表示	10
1.9 反向器	12
1.10 0 與 1 的表示法	13
1.11 二進制數系統	14
1.12 十進制與二進制之間的轉換	15
1.13 八進制與十六進制數系統	17
1.14 二進數和邏輯變數	19
1.15 布林代數定理	20
1.16 狄摩根定理	25

4 電腦工程學（硬體概論）

1.17 維思圖	25
1.18 二變數函數	29
1.19 至斥或函數	31
1.20 NAND 與 NOR 函數	32
1.21 暗示函數	35
1.22 各種運算間的關係	35
1.23 運算的充分性	36
1.24 NAND 的充分性；NOR 的充分性	37
1.25 布林代數定理的應用例題	38
1.26 附加的例題	41
1.27 邏輯圖	45
1.28 數碼	47
1.29 術語	49
1.30 資料碼	49
習題	50

第2章 邏輯函數

2.1 邏輯函數的標準式：標準的積項和	57
2.2 標準的和項積	60
2.3 全積項與全和項的編號	63
2.4 以全積項與全和項表示邏輯函數	64
2.5 全積項、全和項以及真值表之間的關係	65
2.6 二級的閘結構	67
2.7 使用同一型邏輯閘的結構	70
2.8 卡諾圖（或稱K圖）	75
2.9 利用K圖化簡邏輯函數	80
2.10 也論邏輯相鄰	82
2.11 在K圖中取較大的群	85

2.12	五個及六個變數的K圖	89
2.13	卡諾圖的使用	92
2.14	函數不以全積項表示時的繪圖	100
2.15	不完全指定的函數	102
習 題		103

第3章 基本組合電路

3.1	序論	109
3.2	邏輯電路的族系	111
3.3	TTL系列	113
3.4	CMOS族系	117
3.5	ECL族系	118
3.6	包裝	119
3.7	邏輯操作的開關	119
3.8	邏輯操作的開關或閘；接線及的接法	121
3.9	圖騰柱輸出	125
3.10	三態輸出	127
3.11	IC 閘的例子	128
3.12	控制端點符號：混合邏輯	130
3.13	控制訊號等號	135
3.14	使用於閘結構的混合邏輯	137
3.15	解碼器	140
3.16	編碼器	145
3.17	碼轉換器	151
3.18	多工器	153
3.19	以集極開路與三態輸出來作多工	158
3.20	解多工器	162
習 題		163

第4章 正反器、暫存器以及計數器	169
4.1 導論	169
4.2 用 NOR 閘作的電栓	171
4.3 用 NAND 閘作的電栓	174
4.4 無顫動的開關	176
4.5 閘控電栓	178
4.6 時脈定時	180
4.7 電栓當作儲存元件的限制	183
4.8 主僕式正反器	186
4.9 正反器的計時圖	190
4.10 二相時鐘脈波	192
4.11 JK 正反器	192
4.12 主僕式正反器的捕捉 1 之特性	195
4.13 邊緣觸發的 JK 正反器	197
4.14 D 型正反器	200
4.15 設定、保持以及傳輸時間	205
4.16 暫存器到暫存器的轉移	207
4.17 移位暫存器	211
4.18 移位暫存器另外的特性與移位暫存器的使用	213
4.19 計數器	218
4.20 環形計數器	219
4.21 尾開關計數器	220
4.22 其他種同步計數器	222
4.23 同步計數器速率的比較	225
4.24 任意模數的同步計數器	227
4.25 上下數同步計數器	232
4.26 鎖制	234

4.27 鏈波計數器	234
4.28 計數器的積體電路晶片	239
習 題	241
第5章 算 術	251
5.1 帶正負號的數之表示法	251
5.2 正負數之二的補數表示法	253
5.3 正負數之一的補數表示法	257
5.4 兩個二進制數的加法	259
5.5 串加器	262
5.6 並行加法	264
5.7 簡易的加減計算器	265
5.8 減法器	268
5.9 高速加法器	269
5.10 前瞻進位加法器	270
5.11 使用前瞻進位的位元群	274
5.12 前瞻進位的再度利用	277
5.13 算數邏輯單位	280
5.14 BCD 加法	286
5.15 乘法與除法	288
習 題	289
第6章 記憶體	295
6.1 隨意存取記憶體	295
6.2 半導體隨意存取記憶器的構造	297
6.3 並聯記憶體	299
6.4 一維和二維記憶體內部結構	304
6.5 僅讀記憶體	304

6.6	ROM編碼器的製作	309
6.7	可程式化及可抹去的僅讀記憶體	311
6.8	記憶體內容的改變	312
6.9	記憶器的轉換時間	313
6.10	可程式邏輯陣列	319
6.11	動態隨意存取記憶體	322
6.12	動態隨意存取記憶器的資料感知	323
6.13	動態記憶器的特性	325
6.14	串聯記憶體	327
6.15	電荷耦合裝置；串聯記憶體	331
6.16	記憶體堆疊	335
6.17	巨量儲存器	337
6.18	符號 (Symbolism)	342
	習題	345

第7章 順序電路

7.1	狀態	349
7.2	計數器有如順序系統	351
7.3	模4上下計數器	357
7.4	順序偵測器	360
7.5	摩爾和密理電路 (Moore Mealy Circuits)	368
7.6	多餘狀態的刪除	375
7.7	以分割法做多餘狀態刪除	377
7.8	實例	380
7.9	狀態指定	385
7.10	另外一種設計法	386
7.11	基本模式順序電路	389
7.12	一個分析實例	391

7.13 一個設計實例	393
7.14 競賽	396
7.15 消除臨界競賽	398
7.16 實例	401
7.17 刪除多餘狀態	403
7.18 多餘狀態刪除的另一例子	407
7.19 意外雜波與不同步電路	412
習 題	414
第8章 控制器	429
8.1 暫存器間的傳送	429
8.2 其他運算	431
8.3 暫存器對多重指令的響應	436
8.4 簡單控制器	440
8.5 控制器的實現	444
8.6 以移位暫存器設計的控制器	448
8.7 控制器的條件反應	452
8.8 減法的順序	455
8.9 一個簡單計算機模型	460
8.10 計算機的運算過程	465
8.11 計算機控制器的設計	468
8.12 中斷	472
8.13 連繫交換	476
習 題	482
第9章 計算機	489
9.1 基本結構	489
9.2 指令	495

10 電腦工程學 (硬體概論)

9.3	指令總結	504
9.4	加法和減法	505
9.5	指令 JMP 和 ISZ 的用法	508
9.6	乘程式	510
9.7	說明呼叫次常式的例子	513
9.8	微程式設計	516
9.9	微程式分岐	518
9.10	條件分岐	520
9.11	導管管線	521
9.12	微程式控制器	523
9.13	控制僅讀記憶器的內容	526
9.14	定址法	529
9.15	堆疊器	530
習題		533

第10章 微處理機

10.1	導論	537
10.2	程式設計者眼中的微處理機結構	539
10.3	8080 的一個位元組、二個位元組以及三個位元組的指令	542
10.4	移動資料的指令	544
10.5	直接涉及累積器的指令	545
10.6	算術指令	547
10.7	例題	552
10.8	遞增 - 遞減指令	553
10.9	邏輯指令	554
10.10	比較指令	556
10.11	旋轉指令	556

10.12 取補和設置指令	557
10.13 例題.....	558
10.14 跳躍指令.....	559
10.15 呼叫和返回指令	561
10.16 壓入和彈出指令	565
10.17 例題：乘法	570
10.18 6800 微處理機	572
10.19 6800 定址模式	573
10.20 6800 條件碼暫存器	574
習 題	576
附 錄	581
A ASCII 碼	
B 附圖	
C 8080 指令集	

邏輯變數的代數

1.1 變數與函數

我們已熟悉變數和一變數的函數之觀念了。變數的範圍——即變數 x 所能代入的値之範圍乃是可以無數的方式來指定。例如： x 可定義為從負無限大到正無限大的任意實數；或被限制為從 -17 至 -4 的値；或被限制為正整數 1 至 10 的値……等等。

函數乃是一種法則，藉此我們可以從獨立變數 x 來決定一相依變數 y 的値， y 相依於 x 可寫成 $y = f(x)$ 的方程式。因此，例如假設我們希望 y 被 x 來決定是透過一法則： x 本身相乘，此一乘積再被乘以 5 ，然後加上 3 。於是我們可以將 x 與 y 間函數的關係表示成方程式 $y = 5x^2 + 3$ 。在這簡單的例子中，我們使用數學的乘法和加法運算來決定 y 。然而，那也祇能當可被允許的 x 値之數量是有限時，才可以簡單地以表格列出每一 x 値及相對的 y 値來說明這一個函數。當 x 可能值的數目很少時，以這種表格來表示是極可能且相當方便的。我們現在考慮上述的例子（ $y = 5x^2 + 3$ ），我們限定 x 為整數 $x = 0, 1, 2$ ，與 3 ，如圖 1.1-1 所示， y 與 x 間函數的關係可以用表格的方式來說明。

由這些基本的想法作一簡單的擴展，很清楚地可知道獨立和相依變數沒有必要一定是有數值。例如，假設獨立變數 x 其範圍是一十字路口交通號誌燈的顏色，而設相依變數 y 表示摩托車騎士接近十字路口

口可能的舉動，那麼 x 與 y 間的函數關係可由圖 1.1-2 來表示。其中可由 x 來表示的值以宣告式的敘述來表達“綠燈”或“黃燈”或“紅燈”。類似地，可由 y 來表示的值是“騎士將行駛”等等。

x	$y = f(x)$
0	3
1	8
2	23
3	48

圖 1.1-1 數值函數

x	$y = f(x)$
綠燈	行駛
黃燈	慢下來
紅燈	停止

圖 1.1-2 函數的關係

1.2 邏輯變數

所謂邏輯變數是指具有下列三項明顯特性的變數：

1. 邏輯變數的值只有兩種可能情形。
2. 其值可用敘述（statement）來表示，如上述交通號誌之例題即是。
3. 經由敘述所表示的兩種可能值必須是不互相關聯且互相排斥的。

雖然如前所言，變數沒有必要一定要是數字大小，但是我們也沒有理由一定要排除變數可能是數字的狀況。於是變數 x 可由兩個不同的且互相排斥的值表示，且可敘述為“ x 的值是 7”與“ x 的值是 13”。底下的討論將點出邏輯變數的其他性質。

假設我們要求交通號誌只可以是綠色或紅色，我們排除了燈是黃色的可能性及排除了當號誌改變時綠燈與紅燈都沒有顯示的可能性，在此情況下，圖 1.1-2 的表格內的變數 x 就是一邏輯變數。我們若說“燈是綠色的”，則 $x = \text{綠色}$ ，若說“燈是紅色的”則 $x = \text{紅色}$ 。特別要注意的是，因為這兩種情況是互相排斥的，所以若我們要指出 $x = \text{紅色}$ ，那麼則知由 $x = \text{紅色}$ 表示法也表示為 $x = \text{非綠色}$ 。在一種簡單的記號法中，“非”可以由畫一橫線於邏輯變數值的上端來表示，

因此 $x = \text{非綠色}$ 可寫成 $x = \text{綠}$ ，最後我們得到 $x = \text{綠} = \text{紅}$ 。

1.3 邏輯變數的值

一任意類型的變數，除了可設定為數值外，在一般情形下可以代表任何事物。因此 x 與 y 可代表溫度壓力、時間或速度等等。從數學的觀點來看，在考慮變數間的函數關係時，我們並不關心這變數所代表的事物。因此從方程式 $y = 5x^2 + 3$ 中，假定 $x = 1$ ，我們得到 $y = 8$ 的結果是與 x ， y 所代表的事物十分不相關的， x 變數及 y 變數所能表示的值在意義上與當 x 與 y 變數值均是數字時是相同的。

同樣的，讓我們為這一個邏輯變數的兩個可能值指定兩個名字，使得我們在考慮變數時能與其所代表的事物獨立。任何兩個不同的名字就可以被接納，而能表達互相排斥性觀念的名字更有幫助。為此原由，這些名字可取為“熱與冷”、“進與出”、“高與低”等等。另一組可能的名字就是“真 (true)”與“偽 (false)”，稍後我們將做更進一步註解。因此一邏輯變數，假設為 A ，將有值 $A = \text{真}$ (簡寫成 $A = T$) 或值 $A = \text{偽}$ (簡寫成 $A = F$)。假如 $A = \text{真}$ ($A = T$) 是事實，那麼同樣的我們也可以寫 $A = \text{偽}$ ($A = \overline{F}$)。

現在讓我們回到交通燈號的例子。(因為我們正處理的是邏輯變數，我們將遵守著更通俗的習慣，用 A 與 Z 分別來表示獨立變數與相依變數，而不用 x 和 y)，燈號的顏色與騎士正常的反應之間的函數關係在圖 1.3-1a 中表示。假設關於變數 A ，我們任意的指定值 $A = T$ 為敘述“燈是紅色”，那麼自動地 $A = F$ 代表敘述“燈是綠色”。同樣地，讓我們任意地結合值 $Z = T$ 與敘述“騎士繼續行駛”，那麼燈號的顏色與騎士的反應間的函數關係同樣可以圖 1.3-1b 來表示，假如 A 與 Z 對顏色與騎士行為以不同方式指定，在圖 1.3-1 b 中記錄項目 (entries) T 與 F 出現的式樣就不同，但是顏色與行為間的函數關係則當然不變。

如圖 1.3-1b 中有記錄項目 T 與 F 的表稱為 **真值表** (truth table)。

A	$Z = f(A)$	A	$Z = f(A)$
綠燈	繼續	F	T
紅燈	停止	T	F

(a)

(b)

圖 1.3-1 在 (a) 中的函數關係變成在 (b) 中的真值表

1.4 單一邏輯變數的函數

單一邏輯變數的所有可能函數 $Z = f(A)$ 表示於圖 1.4-1 內的 4 個真值表。為確保我們沒有漏失掉任何可能的函數，我們以下面的方式進行所有可能函數的研討。在 A 行 (column) 我們簡單的列出它的兩個可能值 F 與 T，我們現在在 Z 行中有兩個空缺必須填入記錄，在每一個空缺的位置上我們有二個可能的記錄，因此產生可能不同的 Z 行項目數是 $2 \times 2 = 4$ 。寫出這四行，我們很有信心地確定沒有任何漏失了。在圖 1.4-1 a 中，因為在 Z 行與 A 行下的每一列 (row) 記錄均相同，我們將函數寫成 $Z = A$ ，在圖 1.4-1 b 中函數式為 $Z = \bar{A}$ ，在圖 1.4-1 c 則為 $Z = F$ ，在圖 1.4-1 d 中 $Z = T$ 。讀者可能會認為圖 1.4-1 c 與 d 事實上並不能算是函數。此乃因為在其中之一的情形 Z 是偽，在另一情形 Z 是真，且這兩種情形與邏輯值 A 是不相關的。

A	$Z = f(A)$	A	Z	A	Z	A	Z
F	F	F	T	F	F	F	T
T	T	T	F	T	F	T	T

(a)

(b)

(c)

(d)

圖 1.4-1 單一變數的 4 個函數

1.5 兩邏輯變數函數