

DSP 數位控制簡介

1. 簡介
2. DSP的硬體架構
3. 以DSP為基礎的數位控制系統
4. 微處理器 / 數位信號處理器的控制功能
5. 數位控制系統設計
6. 參考文獻

1. 簡介

數位信號處理器(Digital Signal Processor,DSP)的設計主要即在於以數位計算的方式進行信號處理，因而先天上即具有強大的數值計算能力，主要應用於電腦繪圖、儀器量測、影像語音、控制通訊等領域。DSP也可視為一個具有強大計算能力的微處理器，因此舉凡微處理器可以應用的場合，如需要更快速的計算能力，則可考慮使用DSP。但值得注意的是，單晶片微控器(microcontroller, mC)已廣泛應用於工業控制領域，其關鍵主要在於完整的I/O界面，而一般的DSP並不具備這些功能。

在工業控制的應用領域，如機器人控制、磁碟機與光碟機的控制、伺服控制等[54],[56], [58], [59]，採用DSP來實現數位控制器的主要著眼點即在於其快速的計算能力，表1.1列出一些主要處理器與DSP在執行一個典型的PID控制迴路所能達到的取樣頻率[23]。

由於DSP的快速計算能力，因而具有廣泛應用的潛力，採用DSP來實現需要複雜計算的數位控制系統，以達到高性能控制系統的需求即為未來發展的趨勢。因此有眾多的半導體製造商投入高性能DSP的開發，表1.2列出一些廠商所提供的DSP晶片，表1.3列出一些支援廠商所提供的軟體發展工具。

表1.1 PID數位控制取樣頻率

Microprocessor	Clock	fs
8086	8 MHz	2 KHz
Z8000	5 MHz	2 KHz
68000	10 MHz	4 KHz
32010	25 MHz	34 KHz
320C14	25 MHz	43 KHz
320C25	25 MHz	70 KHz

表1.2 一些典型的DSP

Signal processors		Type	Available
NEC	μ PD 7720	U	
Texas Instr.	TMS 32010	U	
Fujitsu	MB 8764	U	
STC	DSP 128	U	
Texas Instr.	TMS 32020	U	
Texas Instr.	TMS 320C25	U	1982
Nat. Semi.	LM 32900	U C	1983
Analog Dev.	ADSP 2100	U C	1984
Phillips	PCB 5011	U	1985
Thomson	TS 68930	U	1985
Motorola	DSP 56000	U	
Nat. Semi.	LM 628	A	
NEC	μ PD 77220	U	
NEC	μ PD 77230	U F	

U universal
 C processor core (external memory)
 A algorithm-specific
 F floating-point arithmetic

為了推展DSP的應用，一些DSP的製造商通常會提供評估模組(evaluation kit)，這些評估模組通常包含一個以DSP為核心的電路板及相關的支援軟體；這些電路板一般具有記憶體、高解析度的類比/數位及數位/類比轉換器。支援軟體則包含組合語言組譯器(assembler)、C語言編譯器(compiler)、載入器(loader)與偵錯器(debugger)等。DSP雖然均具有快速的計算能力，但其種類繁多，硬體架構亦不同，對各別應用的適用性也有所不同，因此設計工程師在選用DSP時，不僅需要考慮其計算速度、價格、軟體支援能力、與供貨能力等，其應用上的特殊功能亦是重要考慮因素。例如TI的TMS320C14特別適合於數位控制的應用，而TM320C40則是為列陣信號處理而設計的。

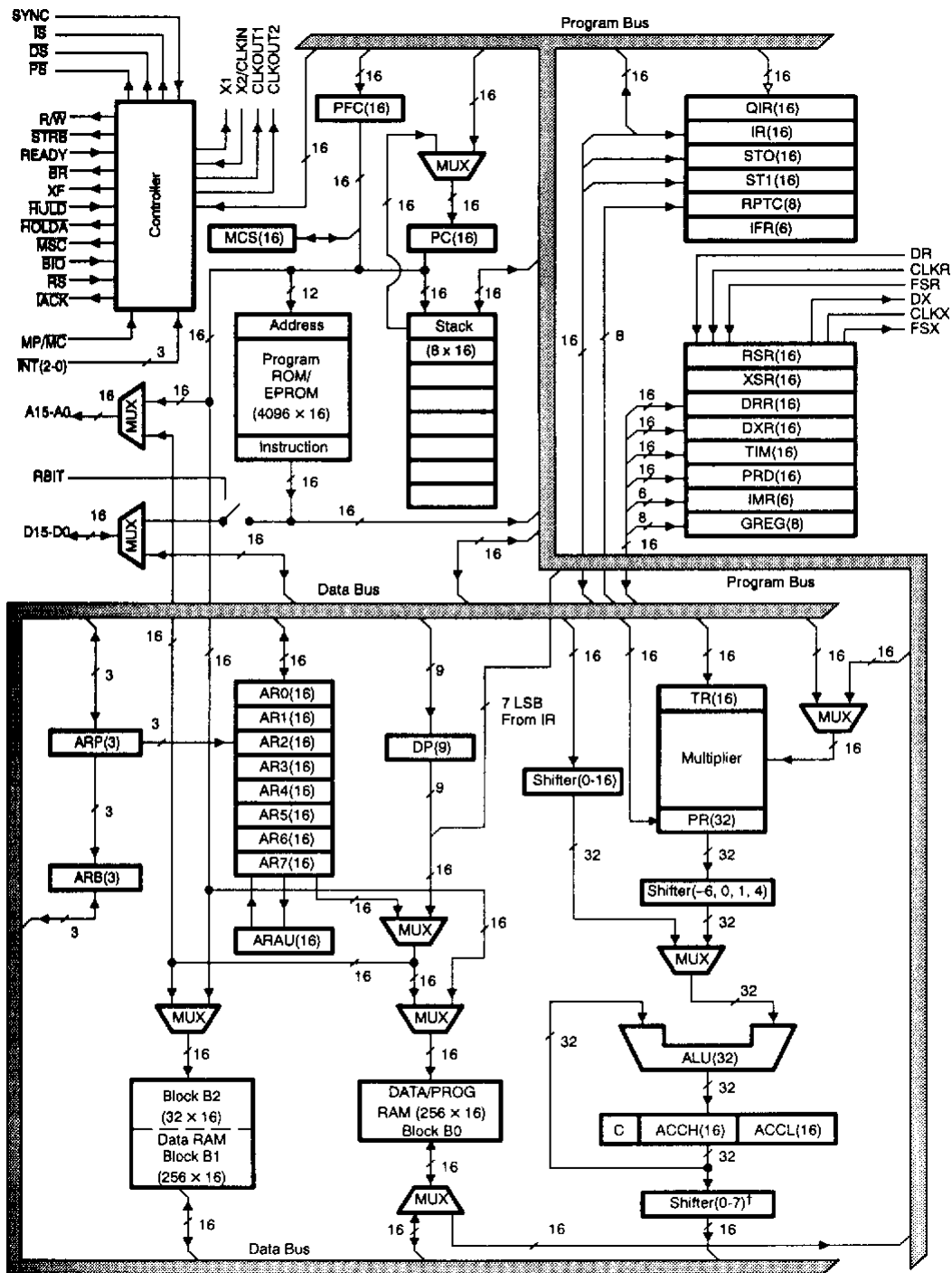
表1.3 DSP與其支援發展工具

Manufacturer	Evaluation kit	Price (US\$)	DSP chip	Included software
Analog Devices	EZ-Kit	\$499	ADSP-2101	C compiler, simulator, and demo programs for DTMP, FFT, filter, speech processing, graphics, and music applications.
Array Microsystems	a66550/8K	\$9900	a66111	Control software, code generator, and an assembler.
AT&T Microelectronics	ATT93C010EK	\$2499	ATT93C0101	Two assemblers (microcontroller and DSP chip), sample DSP chip, sample disk-servo and head-actuator code.
	MP3210	\$4995	Two DSP32102	C host drivers, demos, and utilities. An optional Multimedia software package with a C compiler, simulator, and applications library costs \$3000.
GEC Plessey	FFT Processor Evaluation System	\$5380	PDSP16510	Graphics-based software control program, C-routine library.

Motorola	DSP56156 Application Development System	\$3000	DSP56001	Assembler, linker, and simulator cost \$495 for a PC or Apple Macintosh; \$5500 for Sun3 workstations.
	M68HC16Z1EVB	\$375	68HC16Z1	Control software, macro assembler, filter-design package, real-time kernel, floating-point routine, and a C compiler demo.
SGS-Thomson	ST1893X HDS	\$10,000	ST18932	Macro assemble, emulation-control software
Star Semiconductor	SPROC1ab Development System	\$8950	SPROC-1400	Block-diagram editor and compiler, signal-processing and filter-design libraries, loader, and a debugger.
Texas Instruments	Multimedia Developer's Toolkit	\$5000	TMS320C313	C compiler; debugger; nad algorithms for image and speech compression/ decompression, music functions, and fax.
	TMS320C1XEVM	\$795	TMS320C16	Assembler, linker, and demo software.
	TMS320C2XEVM	\$995	TMS320C26	Assembler, linker, and demo software.
	TMS320C3XEVM	\$995	TMS320C304	Assembler, linker, and demo software.
	TMS320C5XEVM	\$995	TMS320C50	Assembler, linker, and demo software.
Zilog	Z86C9500ZC0 Evaluation Kit	\$200	Z86C951	Sample assembly-language routines; assembler available from third-party vendors.
	Z86C66 V.22bis Modem Kit	\$200	Z89C65 or Z89C661	2400-bps modem or speech-compression firmware.
Notes:				
1. This processor contains both a general-purpose microcontroller and a DSP chip.				
2. Also available from Ariel				
3. Evaluation board also works with TMS320C31 and TMS320C40 DSP chips.				
4. Also available from Atlanta SignalProcessors				

2. DSP 的硬體架構

DSP為了達成其快速的計算能力，其晶片在設計時通常採用16位元的字元長度，並以32位元設計中央處理單元(CPU)內的累積器(accumulator)與暫存器(register)等。在CPU架構設計上亦多有不同，有的採用pipeline架構，有的則採用哈佛架構(Harvard architecture)或特殊的硬體架構。



TMS320C25/E25 Block Diagram

NOTE: Shaded areas indicate a bus.

圖2.1 TMS320的硬體架構

圖2.1所示為德州儀器所製造的數位信號處理器TMS320系列的硬體架構，以16位元為基本資料處理單元，採用16位元的資料與位址匯流排與32位元的暫存器，其核心為32位元的算數運算邏輯單元(ALU)與累積器(ACC)，並以硬體方式來實現不同的功能，如乘法器、位移器等，以提升其

計算速度。在整數計算方面，為了減少數值計算所衍生的問題如溢位(overflow)與尺規劃(scaling)等，採用硬體式的位移暫存器。表2.1列出了TMS320系列在硬體架構上的特色。

表2.1 TMS320硬體架構特色

Feature	Benefit
Single-cycle instructions	Execute advanced control algorithms in realtime
Pipelined architecture	Controls high-bandwidth systems
Harvard architecture	Simultaneously accesses data and instructions
Hardware multiplier	Minimizes computational delays
Hardware shifters	Have larger dynamic range
16-bit word length	Minimizes quantization errors
32-bit registers	Minimizes truncation errors
Hardware stack	Supports fast interrupt processing
Saturation mode	Prevents wrap-around of accumulator

TMS320系列的DSP晶片為了加快其計算速度，在設計上一些常用的數學計算指令往往以硬體方式實現，例如差分方程式的z-1運算元(operator)可以單一指令DMOV完成，而MACD指令則可同時完成四個操作：兩個數的相乘、移動資料、與前一比資料累加、與載入下一個暫存器。由於這些特殊指令的硬體設計，使得DSP在運算速度上較一般的微處理器快上10倍以上。

為了說明DSP與一般微處理器的不同，可從其硬體架構上瞭解。圖2.2所示為目前已廣泛應用於工業控制的16位元單晶片微控器(microcontroller) Intel 80C196KA的硬體架構圖，圖2.3則為TI公

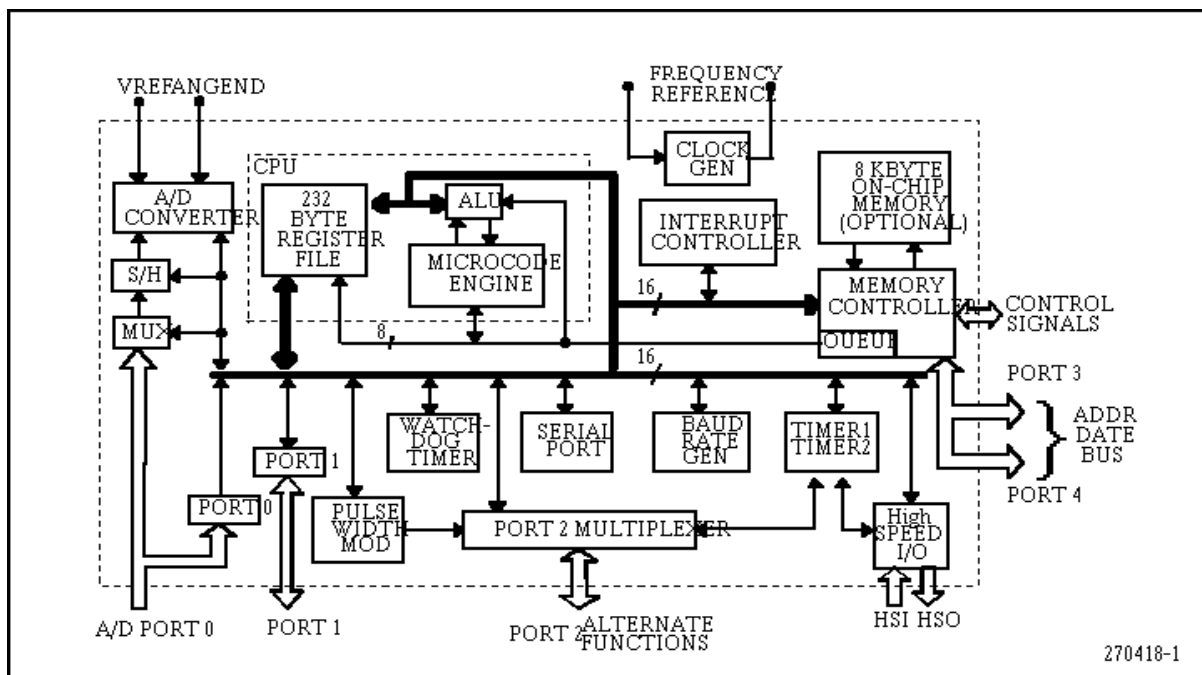
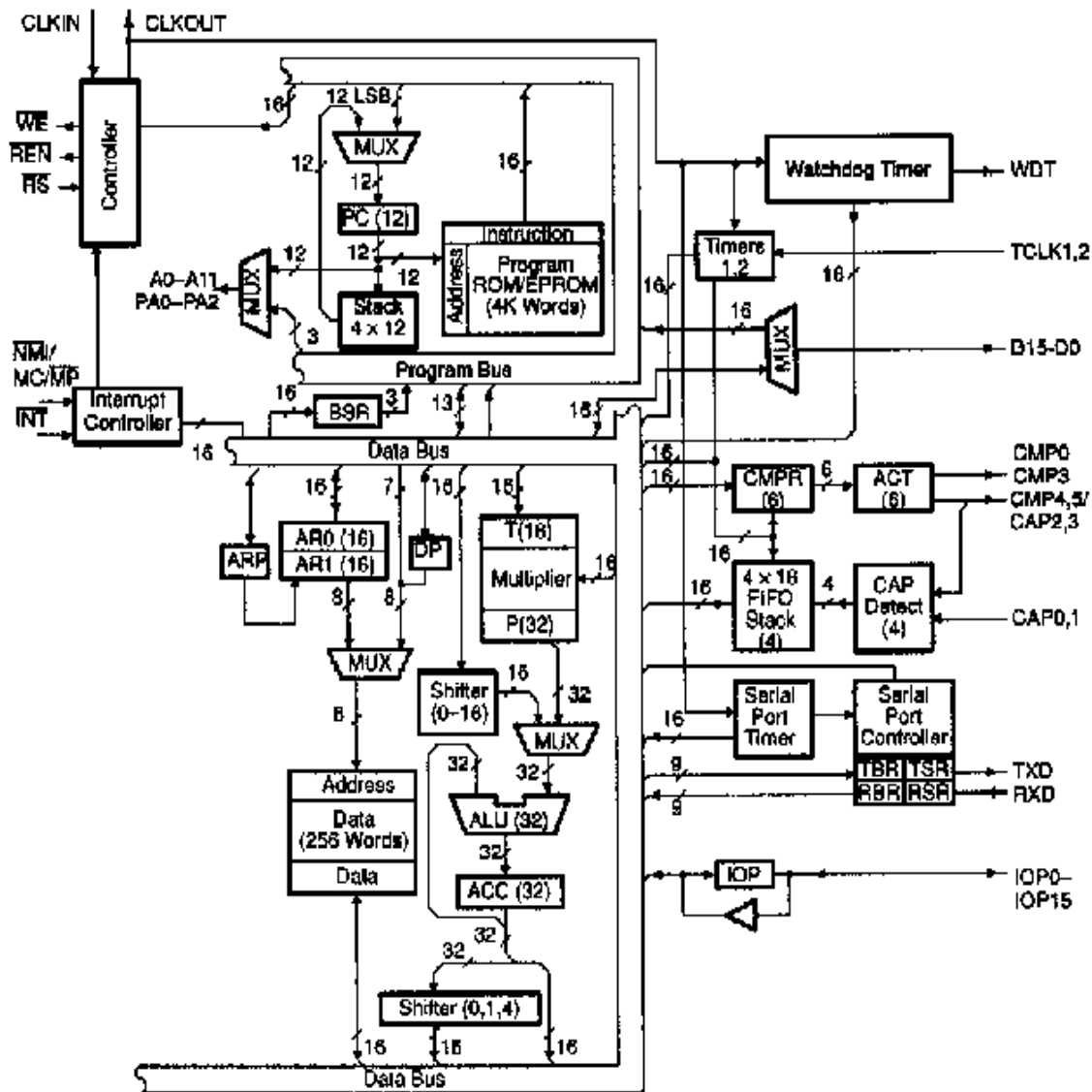


圖2.2 單晶片微控器80C196KA(Intel)的硬體結構圖

司為了應用DSP於工業控制所特別設計的32位元DSP控制器 TMS320C14。微控制器一般在設計上不僅包含CPU、RAM與ROM或EPROM(視其型號而定)，主要在其I/O能力的增強，例如80C196K即具有A/D轉換器、PWM脈寬調變輸出、串聯介面、數位I/O與可程式化計時器等。

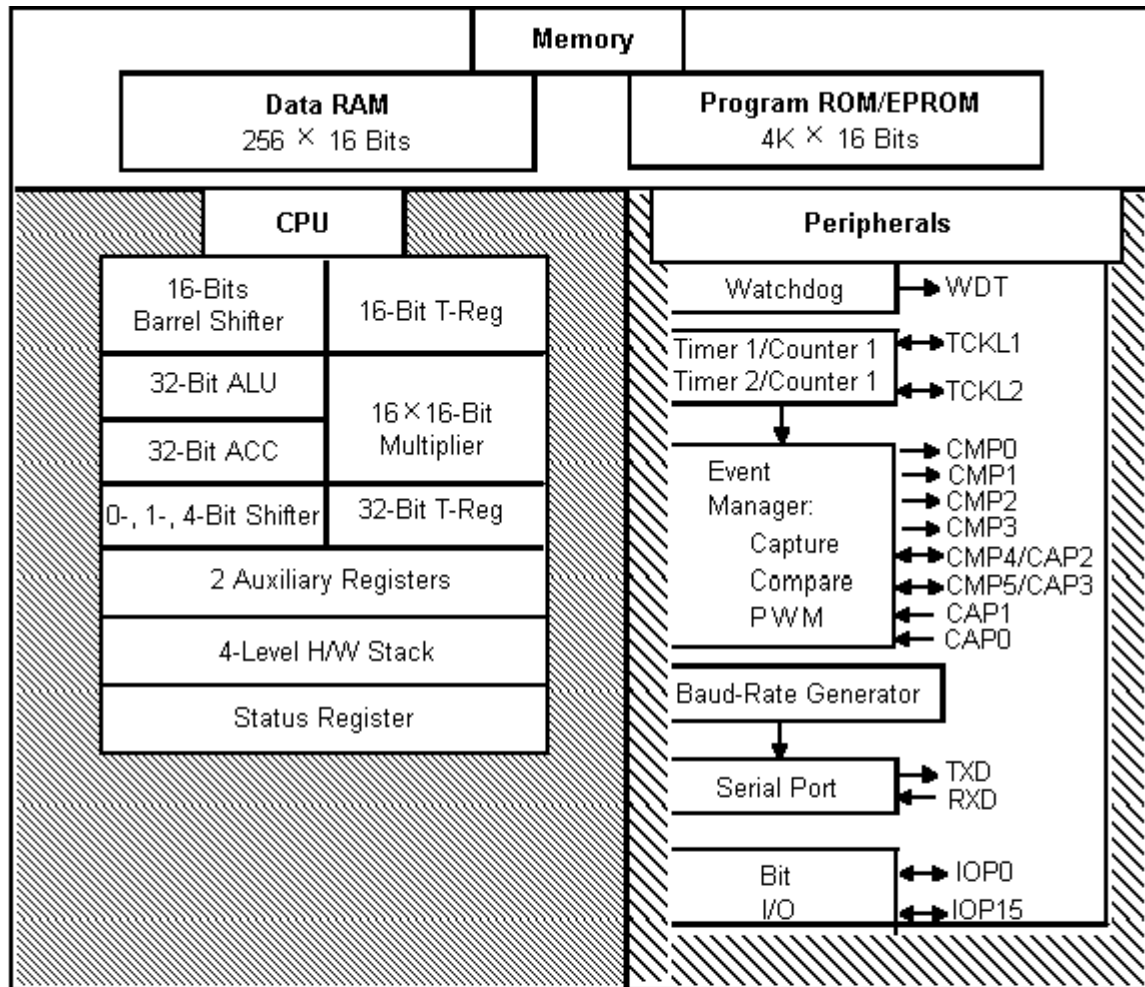


Legend:

- ACC = Accumulator
- ACT = Action Register (Bit Selectable)
- ALU = Arithmetic Logic Unit
- ARP = Auxiliary Register Pointer
- AR0 = Auxiliary Register 0
- AR1 = Auxiliary Register 1
- BSR = Bank Select Register
- CAP = Capture
- CMPR = Compare Register

- DP = Data Page Pointer
- IOP = Input/Output Port
- P = P Register
- PC = Program Counter
- RBR = Receive Buffer Register
- RSR = Receive Shift Register
- T = T Register
- TBR = Transmit Buffer Register
- TSR = Transmit Shift Register

圖2.3 TMS320C14(TI)的硬體架構圖



- 160-ns instruction cycle
- 100% object code compatible with TMS320C15
- 4 16-bit timers
 - 2 general-purpose timers
 - 1 watchdog timer
 - 1 baud-rate generator
- 16 individual bit-selectable I/O pins
- Serial port - UART
- Event manager with 6-channel PWM D/N capability
- CMOS technology
- 68-pin PLCC and CLCC packages

圖2.4 TMS320C14的功能方塊圖與特色

由於包含這些週邊I/O界面電路，因此利用微控制器來實現數位控制器就大幅簡化了相關的硬體界面電路設計，同時由於大量生產也就降低了價格。其硬體架構也與一般傳統泛用型微處理器類似，因此也就有著類似的組合語言指令與定址模式(addressing mode)，較易為一般已熟悉泛用型微處理器的工程師所接受。但是也因為受到傳統微處理器硬體架構的限制，在數值計算的處理上就無法與特別為處理數值計算所設計的DSP相比，因此在實現以數值計算為主的數位控制器時，在計算速度上即無法與之相抗衡。

數位控制器的數值計算能力為什麼如此重要呢？主要是因為如果具有較快的計算能力，則控制器的控制定律(control law)則可在較短的時間內完成，也就能有較高的取樣頻率(sampling frequency)，整個控制系統的頻寬(bandwidth)也可因而加寬，使系統具有較佳的動態響應，而提高整體性能。由此可見取樣頻率所造成的影響，因此數位控制器的數值計算能力是相當重要的。又因為控制定律通常是以差分方程式(difference equation)的型式出現，在型式上與數位濾波器(digital-filter)是一樣的，因此特別適合DSP的計算，也就因為這個原因，採用DSP來實現數位控制器就具有更佳的效果。值得注意的是提高硬體計算能力祇是實現高性能控制系統的一環，其他如控制法則的設計、感測器的選擇、控制結構的決定、軟體實現的技巧等均是重要的考慮因素。DSP與一般微處理器比較時，其優劣可參考表2.2。

表2.2 DSP與微處理器的比較

	DSP	Microprocessor/Microcontroller
Advantages	Signal processing architecture High performance Advanced control techniques Additional functions	On-chip peripherals Supervisory functions Familiar architecture
Disadvantages	Limited peripherals	Low performance Computation delay Numerical problems

德州儀器公司所生產的TMS320C14，即是採用320C10系列的硬體架構，為了實現以單晶片DSP完成數位控制器所設計的，因此在經濟效能上採用TMS320C14來設計數位控制器也可以說是一種理想的選擇。TM320C14可取址到晶片本身4K字元(word)的ROM或EPROM，250字元的RAM與32 bit的ALU與累加器，工作於25 MHz的時鐘頻率時，其指令週期為160nsec，16bit的整數乘法可以在一個指令週期完成。TMS320C14具有6個PWM輸出與4個16bit的計時器：其中兩個為泛用型計時器、一個為看門計時器(watchdog timer)、另一個則用來產生串聯介面的字元率產生器(baud-rate generator)。PWM輸出與計時器在控制應用上，尤其應用於馬達控制、電力轉換系統均是不可或缺的。

DSP在數值計算上的性能如與一般泛用型的微處理器與微控器比較，其速度上的差異可參考表2.3。如將320C14與80C196比較，則執行一個PID迴路的計算，DSP快了12倍，由此可見採用DSP來實現數位控制器在數值計算上所得到的優勢。

表2.3 DSP與一般為處理器的比較

FEATURE	320C14	320C25	80C196	68000	68020	UNIT
Instruction cycle time	160	100	333	400	120	ns
Frequency	25	40	12	10	24	MHz
Multiply (16×16) 32	0.16	0.1	2.2	7.0	1.0	ns
PID loop	2.2	1.3	27.0	25.0	4.8	μs
Matrix multiply (3×3) (3×1)	4.3	2.7	24.3	65.2	9.5	μs

德州儀器公司所發展的DSP晶片，可分為五代如圖2.5所示[13-18]。每一代晶片彼此間均具有目標碼(object code)相容的能力，有些晶片彼此間的腳位亦同。TMS320C1X、TMS320C2X、與TMS320C5X三個系列均為固點算數(fixed-point arithmetic)之DSP，採用16位元哈佛架構，內部為32位元之ALU與累加器，以硬體完成 $16 \times 16 = 32$ 位元之整數乘法，除了跳躍(branch)指令以外，其餘指令均可於一個指令週期(instruction cycle)完成，計算能力約介於5 MIPS~28.5 MIPS之間。另外兩個系列：TMS320C3X與TMS320C4X則為浮點算數(floating-point arithmetic) DSP，採用32位多匯流排(multibus)架構，浮點計算採用40位元擴展精度，具有以硬體實現的浮點乘法器與算數邏輯單元。尤其值得注意的是TMS320C4X為平行數位信號處理器，具有6個通訊埠、6個DMA副處理器、兩個計時器與16 G-byte 的定址能力，適合以平行式之硬體架構完成平行計算法則，指令週期為40nsec，執行速度約為275 MOPS (million operations per second)，通訊速度可達320Mbyte/sec。

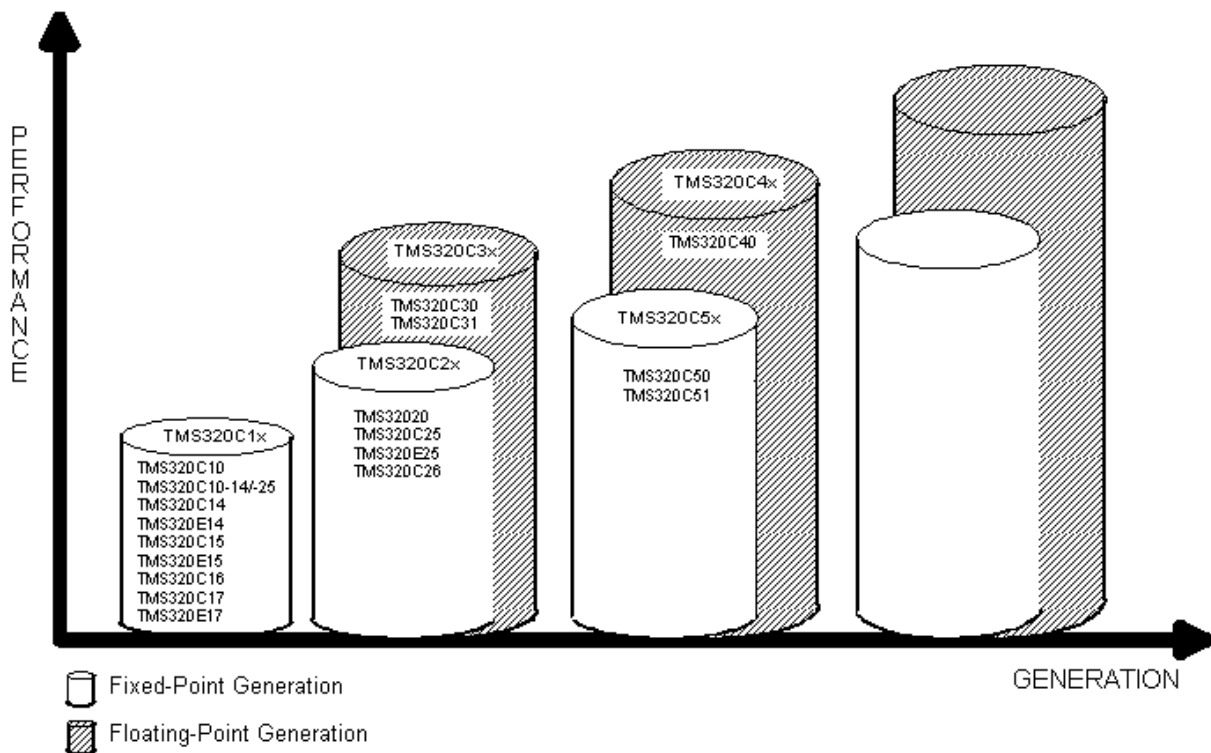


圖2.5 TI公司DSP系列的發展趨勢

3. 以DSP為基礎的數位控制系統

一個典型的閉路控制系統如圖3.1所示，基本上由三個單元所組成：控制器(controller)、受控體(plant)與感測器(sensor)。控制器根據所下達的參考命令與由感測器所測得的回授信號比較，所產生的誤差再經由控制定律計算出適當的修正信號送至受控體，因此控制系統的目的即在於使系統的輸出能根據所期望之輸出而有最快速與精確的反應。控制器之主要目的即在於根據控制命令與回授信號產生適當的修正信號而能使系統有最佳的反應，而這個過程主要是完成控制定律(control law)的執行，因此可以用類比(analog)、數位(digital)或混合(hybrid)的方式完成。

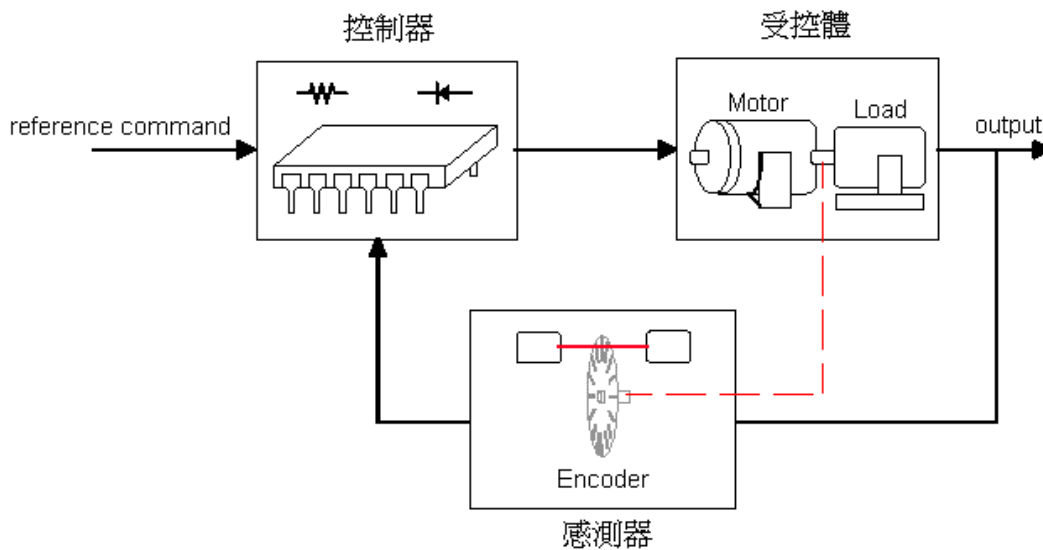


圖3.1 控制系統的方塊圖

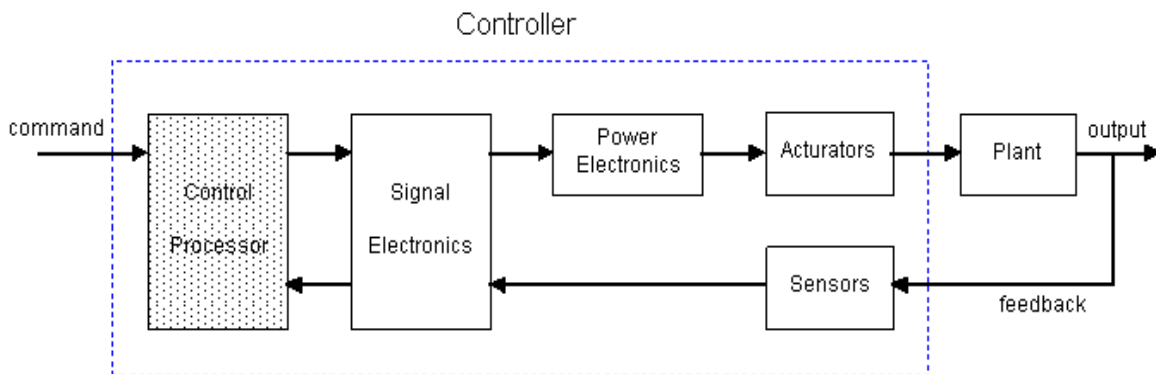


圖3.2 控制器的細部方塊圖

在觀念上一個控制系統亦可說是由兩個主要的方塊所構成：控制器(controller)與受控體(plant) (如圖3.2所示)。受控體即為所控制的對象，它可能是一個馬達、熱水器、或是一枚飛彈，而控制器則由多個子系統所組成，其核心即為控制處理器(control processor)。控制處理器的主要功能即

在於實現控制定律，控制定律由於採取不同的控制方法而具有不同的型式，例如採用PIO方法設計，則其控制定律為：

$$u(t) = k_p(e(t) + \frac{1}{T_j} \int_0^t e(\tau) d\tau + T_d \frac{de(t)}{dt})$$

其中 $e(t)$ 為誤差信號 (error signal)， $u(t)$ 為致動信號 (actuating signal)。如採用可變結構控制 (variable structure control) 設計方法，則其控制定律具有類似下列方程式之型式：

$$\begin{aligned} Z &= cx_1 + x_2 \\ \text{if } x_1 \cdot Z > 0 &\text{ then } \omega = \alpha \\ \text{if } x_2 \cdot Z \leq 0 &\text{ then } \omega = \beta \\ u &= \omega \cdot x_1 \end{aligned}$$

其中 x_1, x_2 為狀態變數 (state variable)， u 為致動信號。控制工程師可根據實際問題之需要而採取不同控制器的設計方法，近代控制理論所發展之各種控制方法如：適應控制、強韌控制、模糊控制 (fuzzy control) 與類神經控制 (neural control) 等等均可用來進行控制定律的設計。控制定律五花八門，但基本上均由數學方程式所組成，再加上一些控制流程指令如 if...then、go to... 等，而查表 (table look-up) 有時亦是必須的，所以控制處理器即在於如何運用軟硬體技術實現控制定律。

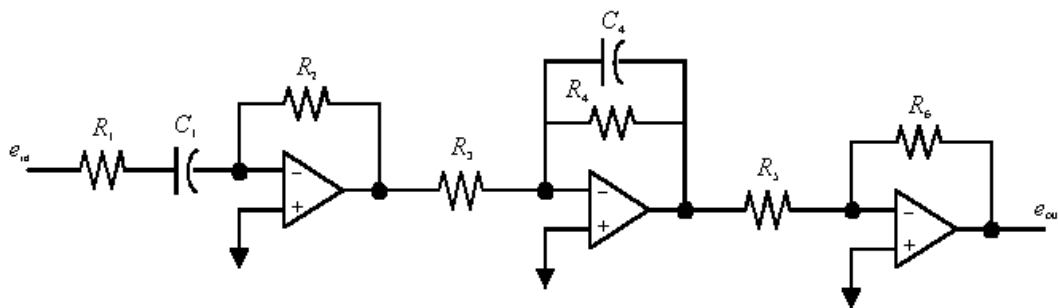


圖3.3 類比控制器

控制定律如以類比電路實現，如圖3.3所示即為一類比控制器，主要藉由運算放大器 (operational amplifier) 來實現控制法則，但如控制定律較為複雜則難以類比電路實現。控制處理器如以數位方式實現則稱之為數位控制器，一般均以微處理器或數位信號處理器實現。類比或數位控制的優劣點可參考表3.1，採用微處理器或DSP的數位控制方式是目前設計控制系統的發展趨勢，主要的優點如下：

1. 能夠以軟體實現複雜的控制法則而不需要藉由複雜的類比路設計。
2. 可經由軟體程式修改而實現不同的控制方法，無需硬體電的更改。

3. 可降低體積、重量與消耗之功率，同時價格也較為便宜。
4. 具有較高的可靠度且易於維修與測試。
5. 對雜訊與干擾有較強的免疫力。
6. 可大幅減少元件的數目而易於製造並提高可靠度。

表3.1 類比與數位控制器之優劣比較

	Analog Controller	Digital Controller
Advantages	High bandwidth High resolution Ease of design	Programmable solution Insensitive to environment Shows precise behavior Implements advanced algorithms Capable of additional functions
Disadvantages	Component aging Temperature drift Hard-wired design Good only for simpler design	Creates numerical problems Must use high-performance processor Difficult to design

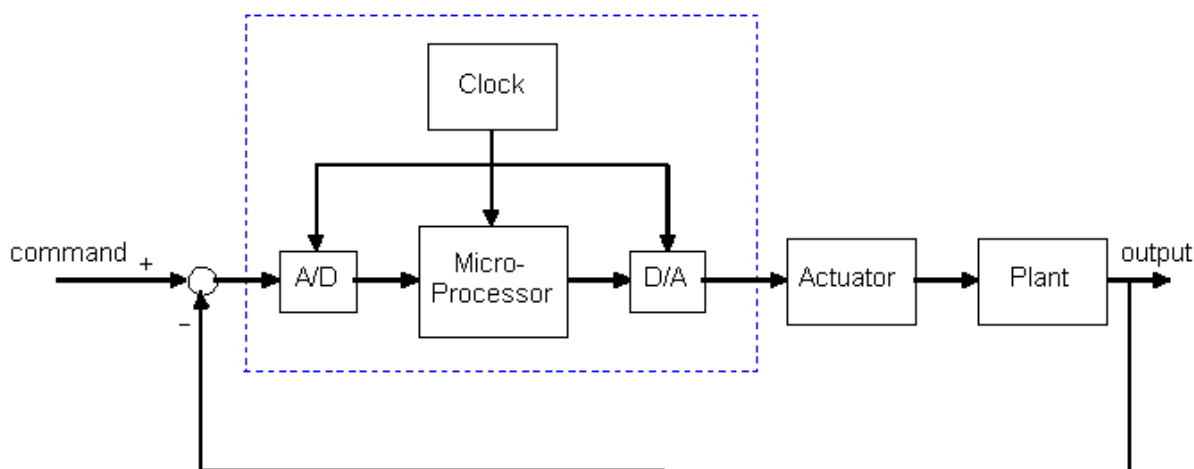


圖3.4 一個典型的微電腦控制系統方塊圖

由上述之優點可知數位控制已是必然的趨勢，至於採用微處理器或DSP則取決於價格、性能之要求、發展工具的齊備與支援能力等因素。一個典型的微電腦控制系統如圖3.4所示，藉由微處理器來實現控制處理器，如果將微處理器以DSP取代，則成為一個以DSP為基礎的圖3.5以DSP為基礎的數位控制系統數位控制系統如圖3.5所示。設計一個以微處理器或DSP為基礎的數位控制系統，通常須具備下列相關知識：

- 數位控制理論
- 微電腦 / DSP的軟硬體設計
- 感測器的選擇與介面電路設計
- 功率放大器與介面電路設計

- 受控體的相關知識

因此設計一個數位控制系統時，通常需要由不同知識領域的工作人員在知識、技術與經驗上整合，在設計團隊的組成與領導是相當重要的。

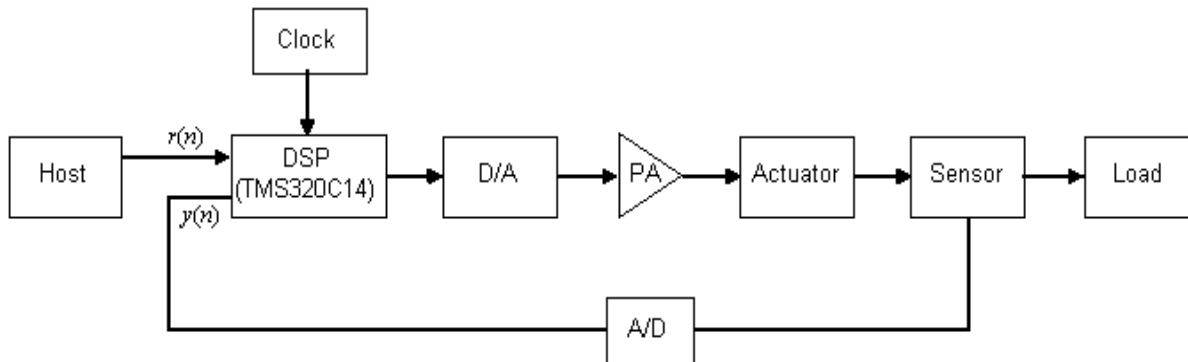


圖3.5 以DSP為基礎的數位控制系統

一個數位控制系統的設計過程如圖3.6所示，分成下列幾個主要步驟：

1. 定義規格。
2. 數位控制系統的理論層次設計。
3. 決定硬體架構。
4. 軟/硬體設計。
5. 系統整合測試與確認。

微電腦 / DSP—數位控制的優點與限制

以微電腦或數位信號處理器為基礎所設計的數位控制系統是未來的發展趨勢，其優點與限制可摘要如下[2]-[5]：

較低的硬體成本

採用微處理器或DSP主要之優點就是能夠以軟體程式取代複雜的特定硬體設計，由於大型積體電路(VLSI)的快速發展，廉價的晶片、更高計算邏輯功能的CPU與更大容量的RAM與ROM，均能以更快的速度實現更複雜的控制法則，因此實現控制功能之硬體價格亦相對降低。同時由於ASIC (Application Specified IC)與顧客委託設計均能將更多的功能由較少的晶片完成，不僅提高系統的可靠度，也降低了生產與維修測試的成本。

更高的可靠度

複雜的控制系統應用於工業自動化如CNC或機器人，應用於國防工業如飛彈與導航等，均需要更高的可靠度。系統的可靠度或MTBF(mean time between failure)均可因採用微電腦或DSP而提高，因為VLSI晶片的可靠度遠大於一些採用個別元件所組成的等功能電路，在生產製造上亦較為簡單。根據過去的經驗顯示，一個以微處理器或DSP控制的電力電子系統或馬達驅動系統，微電腦或DSP晶片的可靠度都是較高的。

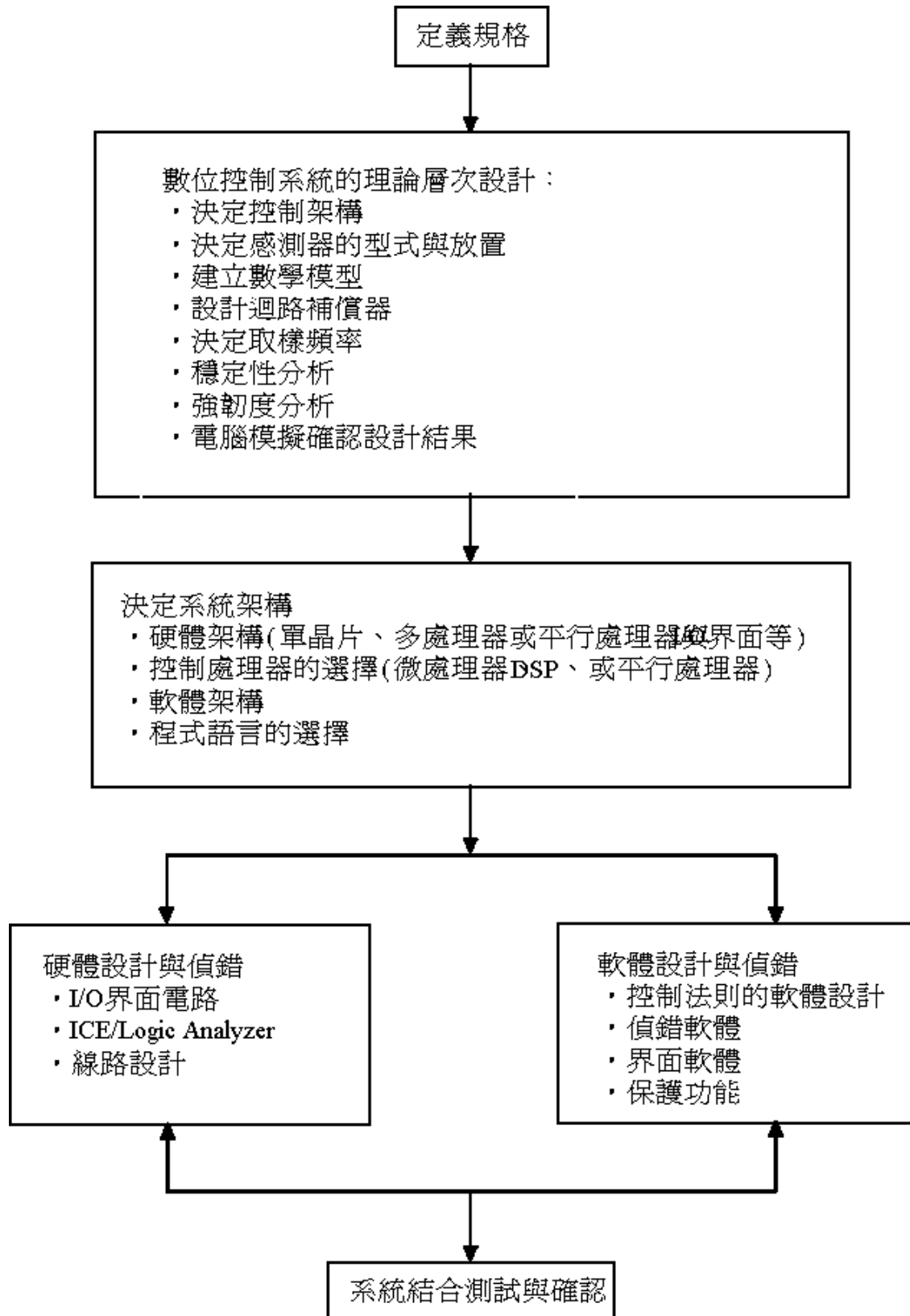


圖3.6 微電腦數位控制系統設計流程

電磁干擾(EMI)問題較少

VLSI的高密度電路積體化，降低了在電力電子系統中高電壓與高電流變化對電路所形成干擾的影響，控制晶片的屏蔽(shielding)較為簡單，經由電力線(power line)或信號線所傳導之雜訊亦較易濾除。

不會有漂移與參數變化

在類比電路設計所遭遇的一些問題如因溫升或環境變化而導致電路的工作點有漂移的現象，這種現象也會因為元件老化而發生，但在數位電路則可消除這些問題，以微處理器軟體實現的數值計算不僅不會有參數變化，計算亦百分之百正確，而截位(truncation)與溢位(overflow)等問題亦可由適當的尺規劃(scaling)而解決[35]-[38]。

具有與階層式數位控制相容的能力

採用微電腦的低階區域性控制器可與上層的高階控制系統藉由數位通訊而整合，例如在自動化工廠的應用，區域控制器與主控制器可藉由雙向通訊介面互通訊息，對於整廠之監控與檢修均更易於達成目的。

通用的軟硬體架構

採用微處理器或DSP實現數位控制系統，可使系列產品具有相同或類似的硬體架構，此不僅易於維修，也簡化了生產製造的過程，同時對軟體設計而言，也因為有了共同的硬體標準而較易規劃與設計。以變頻器為例，針對不同的應用需求即可加入不同的軟體功能模組，這些規劃在複雜的控制系統設計上是非常重要的。

自動偵錯能力

採用微電腦控制可藉由軟體進行系統硬體的自動偵錯，不僅不確保系統在操作上的正常與安全，並且由於具有偵錯的能力也簡化了維修。例如在變頻器設計上可藉此檢測系統是否接線正確、功率級是否有短路現象等。

計算能力的限制

雖然由於微電子技術的快速進展，微處理器與DSP的執行速度已大幅提高，但在執行速度上仍無法與以特定類比或數位電路所完成的硬體控制所抗衡，因為特定硬體在信號處理上是同時以平行方式完成的，而在微處理器為基礎的數位控制則是以串序(serial)方式執行指令而完成，如果微處理器有數個工作(task)要做，則亦須將其分散於數個時間槽(time slot)依序完成，因而造成了取樣延遲(sampling delay)，因此訊號無法連續處理，在取樣(sampling)之間的訊號就無從得知，這樣的限制使得系統的頻寬也就受到了限制，這個限制雖然可藉由多重微處理器與平行處理器的技術而降低，但先天上取樣效應所造成的限制則是無法去除的。

量化誤差

物理系統的信號先天上就是類比的，因此以微電腦或DSP為基礎的數位控制系統就必須藉由

A/D與D/A轉換器才能與外界的類比信號溝通。由於這些類比/數位轉換器均有解析度上的限制，通常為8-位元或12-位元，因此在取樣信號上也就造成了量化誤差(quantization error)。由於量化(quantization)在信號的大小上所造成的不連續與所導致的誤差以及因為取樣(sampling)在時間上所造成的不連續均對信號造成了變形，而這也就是數位控制系統的主要限制。這種因為量化與取樣對信號所造成的誤差是極為明顯的，如果微電腦藉由A/D轉換器將一個正弦信號讀入再經由D/A轉換器送出，比較輸入與輸出信號即可觀察到所造成的誤差，這種現象當正弦波的頻率逐漸增高時，則更為明顯，因此經由數位取樣後即無法回復為原來的信號。由取樣定理(sampling theorem)得知如取樣頻率高於信號頻寬的兩倍即可完全回復原先之信號，但實際上由於理想的取樣不可得，再加上雜訊與量化所造成的誤差，一般而言取樣頻率通常高於信號頻寬的10倍以上。

軟體訊號不易量測

採用微處理器進行數位控制，將以軟體程式實現控制法則，因此其內部變數與控制器的參數均不易觀測與調整，不像類比控制系統藉由示波器即可觀測任何一點的訊號。對於控制器參數的調整雖可藉由程式修改，重新燒寫EPROM而達此目的，但在控制系統測試時，往往需要嘗試性的調機過程，這些均需要額外的軟硬體設計。

軟體發展昂貴

由於微處理器與DSP功能的大幅提升與RAM、EPROM容量的增加，採用"軟體控制已是必然的趨勢，但是發展這些特殊應用與控制的軟體其代價卻是相當昂貴的。在設計過程中不僅需要良好的發展工具，亦需對整個系統有相當深入的瞭解，並能將控制方法以軟體方式實現。在大多數的系統中，亦需要良好的人機界面，如此均增加了軟體的複雜度。當軟體程式的程式碼增加時，在文件說明(documentation)所增加的工作量將更為巨大，如果不能夠符合軟體工程(software engineering)所提出的一些發展原則與規劃管理方法，則整個程式將難以擴充與維護，這樣的問題發生在較複雜的電腦控制系統則尤為明顯，如CNC與機器人控制系統。在系統層次所發生的一些設計問題如精度、穩定度與反應時間的改善亦無法藉一般的微電腦系統發展工具如邏輯分析儀(logic analyzer)或模擬器(in-circuit emulator, ICE)來解決。因此在發展微電腦控制系統時，在軟體開發上所投入的資金與人力往往是遠超過當初所預期的。

4. 微處理器 / 數位信號處理器的控制功能

微電腦/DSP數位控制在工業控制上有廣泛的應用，表4.1列出一些在電力電子與馬達驅動系統的應用，在這些應用中微電腦或DSP所提供之功能可摘要如表4.2所列，以下說明一些如何以微處理器或DSP來實現這些功能以及在設計上的一些考慮因素。

表4.1 微電腦數位控制在電力電子與馬達驅動的應用

- Four-quadrant phase-controlled dual-converter dc drives
- Four-quadrant PWM chopper controlled dc drives
- High voltage dc (HVDC) converter control with cell fault monitoring.
- Uninterruptible power supply (UPS) systems
- PWM inverter drive for induction and synchronous motors
- Four-quadrant current-fed inverter drives for induction and synchronous motors
- Four-quadrant cycloconverter drives for induction and synchronous motor
- Static VAR generators
- Static induction motor starters and phase-controlled drives
- Heating and light dimming controls
- Variable speed constant frequency (VSCF) power generation by induction and synchronous machines
- Synchronous machine static excitation control
- Control of photovoltaic power conversion

表4.2 微處理器 / DSP電力電子系統的控制功能

- Control of feedback loops
- Gate firing control of phase-controlled converters
- PWM or square-wave signal generation of inverters
- Estimation of feedback signals
- Optimal and adaptive control
- General sequencing control
- Protection and fault overriding control
- Signals monitoring and warning
- Data acquisition
- Diagnostics
- Miscellaneous computation and control

回授控制

電力電子系統根據其應用而具有一個或多個控制迴路(control loop)，一個數位控制系統本質上是一個離散系統，經由取樣與量化而在時間與數值上造成不連續的現象，它可能是線性的或非線性的。週期性的取樣不僅為了完成數位控制，也產生功率級所需要的開關信號。如果取樣週期(sampling period)與開關時段(switching interval)遠小於系統的反應時間，則經由取樣所造成的離散時間效應(discrete time effect)通常可忽略不計。在這樣的情況下，以拉氏轉換理論(Laplace transform theory)為基礎的線性系統分析方法可有效的分析系統的一些特性，如果再加上傳輸延遲(transport lag, e^{-Ts} , T=取樣週期)則可得到更精確的結果。反之如果這種情況不成立，也就是不能忽略取樣所造成離散時間效應，則必須採用Z轉換進行系統的分析或者採用狀態變數方法進行系統的分析設計。Z-轉換僅限用於線性離散系統，狀態變數法(state variable method)則可適用於線

性與非線性系統、SISO與MIMO系統，並已發展了完整的理論體系。採用Z轉換則可應用古典控制理論的一些分析設計方法，亦為一般控制系統設計工程師所熟悉。

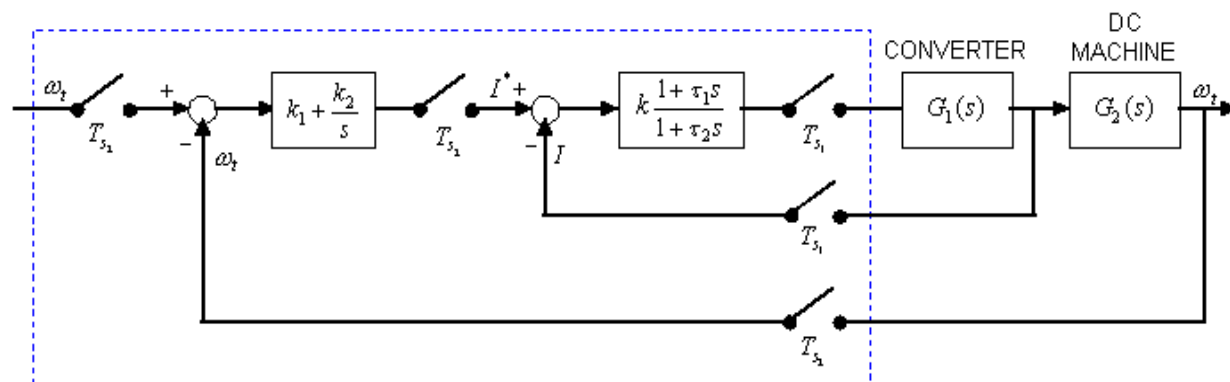


圖4.1 直流馬達速度控制系統

一個典型的多迴路(multiloop)多取樣頻率(multirate)直流馬達驅動控制系統如圖4.1所示，虛線部份由微電腦數位控制完成，此系統有一個採用lead/lag控制的電流內迴路與一個採用PI控制的速度外迴路，控制器、轉換器與馬達的轉換函數均列於圖上。此系統也是一個多取樣頻率系統，電流內迴路的取樣頻率 $1/T_{s1}$ 高於轉速外迴路的取樣頻率 $1/T_{s2}$ 。採用Z-轉換去分析一個多取樣頻率系統是相當複雜的，尤其當比值既不是整數亦不是分數時，因此採用電腦模擬是較可行的。對於電流內迴路的分析可採用Z-轉換，但對轉速外迴路的分析，一般則可忽略內迴路所造成的取樣效應。

補償器的軟體實現

一個回授控制迴路的補償器(compensator)可能是一個單純的增益(gain)，或者是比例積分微分PID控制器、相位領先落後(lead-lag)補償器，或者以其它型式呈現，但最廣為採用的仍是PID或Lead-lag或數位濾波器型式的高階補償器。以下說明如何將類比的PI與lead-lag補償器轉換為數位的PI與lead-lag補償器。

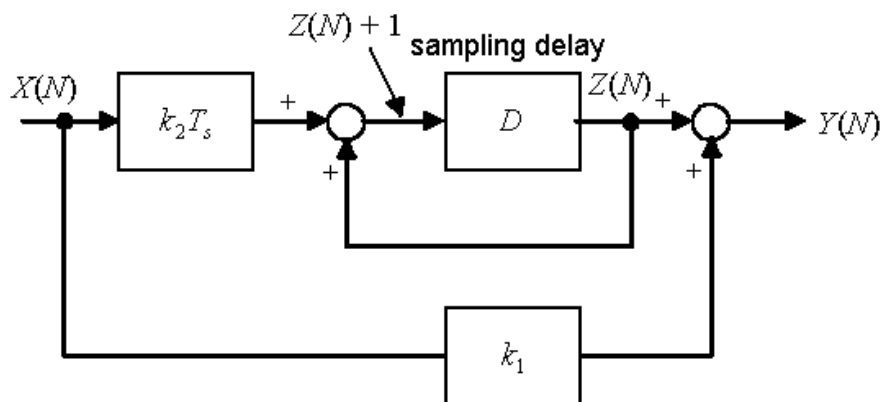


圖4.2 數位PI補償器的實現

一個PI補償器其拉氏轉換函數為

$$\frac{Y(s)}{X(s)} = k_1 + \frac{k_2}{s}$$

上式可改寫為

$$sY(s) = k_2x(s) + k_1s$$

如果取樣週期非常短，則微分可以差分之方式表示，因此上式可改寫為

$$\frac{y(k+1) - y(k)}{T_s} = k_2x(k) + k_1\left[\frac{x(k+1) - x(k)}{T_s}\right]$$

其中 T_s 為取樣週期， $k, k+1$ 則為連續的取樣時刻，上式可以標準的差分方程式(difference equation)表示

$$y(k+1) - y(k) = k_2T_sx(k) + k_1x(k+1) - k_1x(k)$$

若表示為標準的狀態變數動態方程式

$$z(k+1) = Az(k) + Bx(k)$$

$$y(k) = cz(k) + Dx(k)$$

其中 $z(k)$ 為狀態變數， $A = 1$ ， $B = k_2T_s$ ， $C = 1$ 以及 $D = k_1$ 。上述方程式以數位實現之方塊圖如圖4.2所示。控制處理器(control processor)依序計算其中之方塊，並且每隔 T_s 的時段更新 $y(k)$ ，如果 T_s 縮短則其響應將更接近類比的轉換函數。

相位領先一落後(lead-lag)補償器之轉換函數為

$$\frac{Y(s)}{X(s)} = k_3 \frac{1 + \tau_1s}{1 + \tau_2s}$$

上式可改寫為

$$Y(s) + \tau_2sY(s) = k_3X(s) + k_3\tau_1sX(s)$$

將微分項以差分型式表示並將之簡化，其差分方程式可表示為

$$y(k+1) - \left(1 + \frac{T_s}{\tau_2}\right)y(k) = k_3 \frac{\tau_1}{\tau_2}x(k+1) + k_3\left[\frac{T_s}{\tau_2} - \frac{\tau_1}{\tau_2}\right]x(k)$$

狀態方程為

$$z(k+1) = \left(1 - \frac{T_s}{\tau_2}\right)z(k) + \frac{k_3 T_s}{\tau_2} x(k)$$

$$y(k) = \left(1 - \frac{\tau_1}{\tau_2}\right)z(k) + \frac{k_3 \tau_1}{\tau_2} x(k)$$

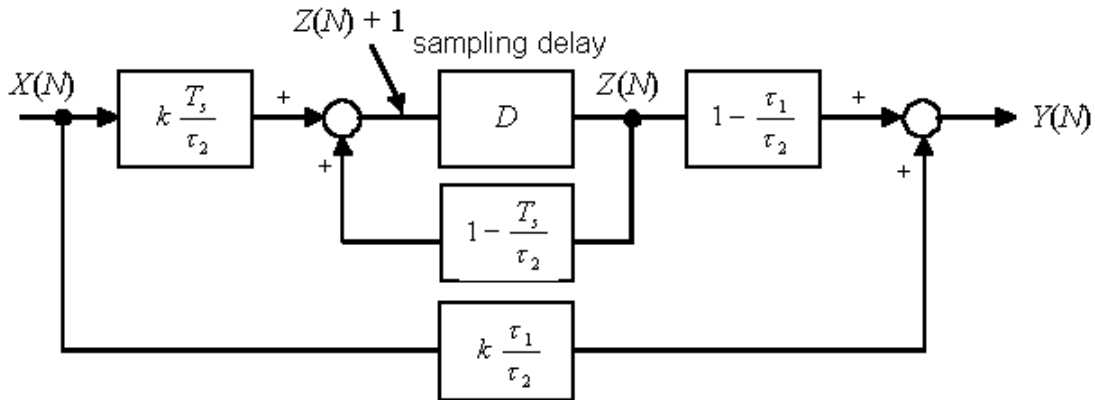


圖4.3 數位lead-lag補償器的實現

其方塊圖如圖4.3所示。有關如何以微電腦或DSP來實現PID與lead-lag補償器可參考[23],[32]。

功率元件觸發信號控制

開流體(tyristor)通常應用於相位控制的電力轉換器，其觸發信號通常為一控制迴路的輸出，由於輸出控制信號與觸發角(firing angle)具有 \cos^{-1} 的關係，因此可用查表法進行此一非線性轉換，所得之數值即可做為計數器的時間延遲量，而使觸發有效的脈寬波則可由線電壓所得之邏輯信號產生。而 $\alpha=0$ 時之同步信號則可由線電壓之零點超越偵測電路所產生之脈衝波做為中斷信號，由於線電壓的頻率經常會波動，因此通常需要一個相位鎖相迴路(phase-locked loop)產生計時器所需之時鐘脈波。時鐘的頻率即決定了觸發角的解析度，如果觸發角之命令發生於觸發之前，則計數器之值可即時更正，否則藉由查表之判斷可決定其位於那一個 60° 的區段內，在這種情況最大的反應延遲為 60° 。

限制器

在一個回授控制系統，迴路的控制輸出信號必須限制在實際系統物理變數能接受的範圍之內，譬如在一個馬達變速控制系統，輸出的電流、電壓、磁通與扭矩等均會受到限制，這些限制也會根據操作情況而改變。

非線性函數的產生

一維或多維的非線性函數可藉由查表、線性插值、或曲線函數計算而產生，這些非線性的函

數關係在微電腦控制是常需要的，例如在馬達控制時為了效率的改善而必須調整磁通與扭矩的關係，或是在感應馬達驅動系統為了維持氣隙在固定磁通下所需要的滑差—電流非線性補償。在變頻器的應用亦必須提供非線性的電壓頻率補償等。在某些非線性系統，可藉由非線性反函數而消除原系統之非線性特性，如相位控制功率轉換器觸發角非線性函數的線性化補償或是相位控制轉換器在不連續導通模式工作狀態時增益變化的補償。

回授信號之估計

在一個數位控制系統，某些信號如電壓、電流等可直接量測再經由A/D轉換器讀入微處理器，馬達光電解碼器的脈衝信號可由一個計數器累加，再將之轉換為一個數位字元讀入微處理器，但有些物理量則不易量測，如磁通(flux)、扭矩(torque)、功率、功率因數等，這些物理量可由其他之回授信號與系統的參數計算出。在運動控制，如何藉由增量的脈衝位置信號來估計轉速也是相當重要的。

可程式化的時間延遲

微處理器利用軟體與可程式計時器可以很容易的產生有延遲的脈衝波，這些脈衝波可用來控制相位觸發角或產生協調性中斷處理信號而控制其他週邊元件。

脈寬調變(Pulse-Width Modulation)

微電腦可產生PWM波以應用於截波器(chopper)或變頻器(inverter)。在截波器控制，可由一個數位字元經由軟體計數或硬體計數器方式產生脈寬調變波。在變頻器的應用裡，有幾種方法可用來產生脈寬調變波，如將基本波之振幅所相對的波寬信號儲於記憶體，而後以查表方式將字元讀出以控制計數器產生需要的脈寬信號。PWM表則可根據諧波消去法(selected harmonics elimination)產生或最小漣波電流之最佳PWM方式產生，這些PWM脈寬查詢表可先經由離線方式(off-line)計算產生，再將之燒錄在EPROM以節省on-line的計算時間。如果微處理器或DSP的速度夠快，則可根據一些數位式脈寬調變方法如Regular PWM或空間向量(space vector)PWM法，直接計算產生脈寬調變波。

可程式化命令產生

主控微電腦可產生命令輪廓(command profile)以命令與其連結之微電腦控制系統，此命令輪廓可用於測試或實際系統之操作命令。有些應用如單軸或多軸運動控制系統，需要藉由鍵盤或面板操作以產生伺服機械系統的移動命令，或者如線性插直器(linear interpolator)或圓弧插值器(circular interpolator)均可藉由微電腦計算產生直線或圓弧運動的移動命令。

數位濾波

功能類似於類比濾波器的數位濾波器可藉由微處理器以軟體計算方式實現，而數位信號處理器尤其適合應用於此一工作。經由數位濾波器之實現，不僅可免除類比電路設計所增加之電路也因為採軟體方式實現，而具有易於修改的優點。

監視與預警

微電腦控制系統經其類比與數位I/O界面，可以很容易的對一些重要的物理量進行監視而防止意外狀況的發生，當信號超出安全工作區時，系統可自動產生警告信號，避免危險之發生。

保護與避開錯誤控制

系統的安全警戒與失效保護可由微電腦來執行，如果保護動作在時間反應上的要求是極為快速的，如功率晶體的短路保護，則必須藉由特殊設計的硬體保護來完成，其他如過載保護、過熱保護等，不需要那麼快的反應時間，則可由軟體保護完成。複雜的保護功能是難以藉硬體來實現的，而保護功能在電力電子系統是至為重要的。系統的避開錯誤控制(fault overriding control)主要是藉由詳細的系統運作分析才得以完成，例如在高壓直流轉換(HVDC)系統，某些閘流體的失效或誤觸發可藉由其他閘流體選擇性的觸發而使系統繼續運作。在三相馬達驅動系統，如果一個相腳(phase leg)的開關元件損壞，系統仍可在單相模式在工作，祇是其輸出扭矩必須降低，系統的性能也降低，但仍能處於運轉狀態，不致於使系統停擺，這在某些應用是非常重要的。

資料擷取與記錄

利用微電腦的功能在工業控制應用中進行資料擷取是相當重要的，資料擷取的目的如下：

- 記錄資料以進行系統性能的評估。

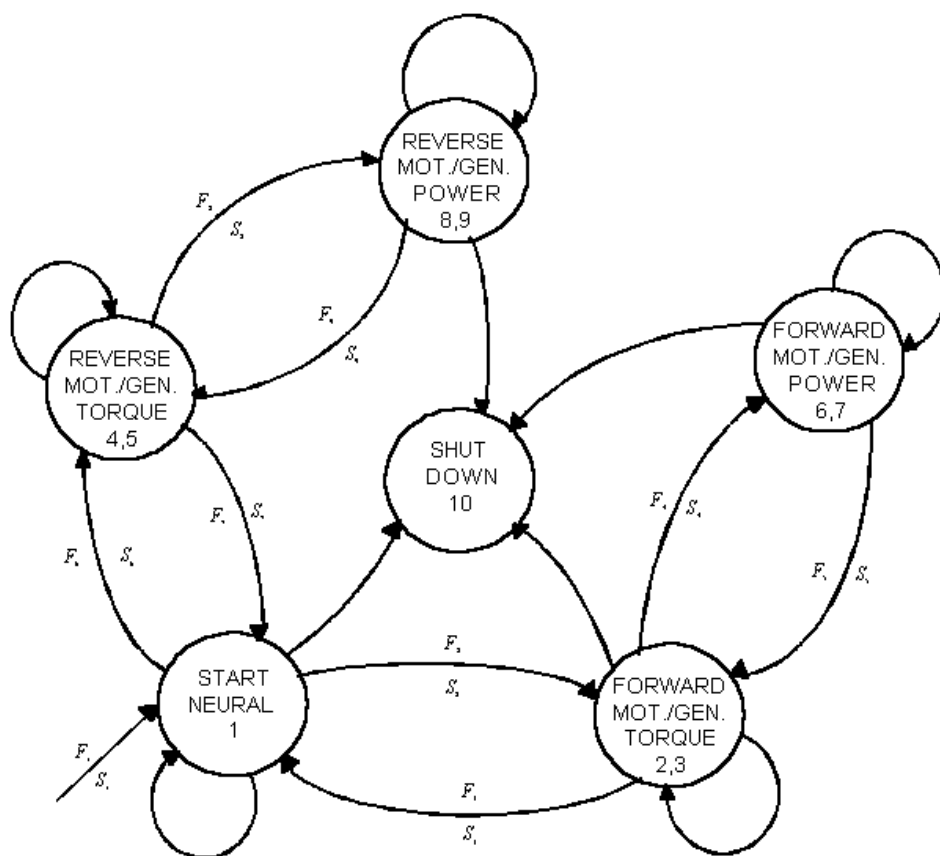


圖4.4 一個簡單的序向控制狀態圖

- 將記錄資料儲存歸檔，以便未來進行性能評定。
- 記錄資料以便系統偵錯。
- 作為系統監視與控制調整之用

序向控制(sequencing control)

微電腦可藉由序向控制而使系統在不同的操作模式能平滑的運轉，例如一個交流馬達驅動系統可能有下列的操作模式：

- 中性啟動(starting neutral)。
- 在定扭矩區正轉/回生(motoring/regeneration)。
- 在定扭矩區反轉/回生。
- 在定功率區正轉/回生。
- 在定功率區反轉/回生。
- 停止運轉。

一個簡單的交流馬達驅動系統的序向控制圖如圖4.4所示，每一個圓圈代表一個操作模式，有箭頭的弧線則代表操作模式間的轉移。操作模式間的轉移，則是因為操作狀態滿足某一組軟體所設定的模式參數，例如由定扭矩轉移至定功率區操作時，將發生在馬達電壓已飽和至方波且磁通控制已失效時。轉移狀況發生的取決標準可由布林函數Fn(Boolean function)來定義，如果外向箭頭(outward arrow)轉移路徑的布林函數無法符合條件，則仍然處於相同的工作模式，以一個環狀箭頭路徑來表示。一旦某個布林函數的條件相符，則控制器執行一組"動作程序"(action routine) Sn 以產生所需要的平滑轉移。系統的停機模式則可由任一個操作模式抵達，這也就意味著系統在任何一個操作模式均有可能轉移至停機模式。如果系統的操作模式不太多，則由一般的流程圖即可掌握其操作狀況，但如果一個龐大的系統有著複雜的序向控制則必須藉由有限狀態機器(finite state machine, FSM) 理論與可程式控制器(programmable controller)來完成分析設計與實現。

測試與偵錯

微電腦在電力電子系統可用來進行系統測試與偵錯的工作，由資料擷取系統所記錄之資料可用於計算一些性能曲線，如扭矩、效率、功率因數等等，感應馬達的無載測試與轉子鎖定測試或是馬達參數均可藉由微電腦系統完成。某些線上測試如轉移函數之判別、系統參數之估測等，均可藉由微電腦控制系統完成。在一個非線性系統不同的操作點，亦可注入小信號擾動，觀測其響應而決定系統的結構、增益、極點與零點的位置等等。微電腦系統亦可藉由其I/O界面、特定設計的硬體與軟體而完成功能強大的自動偵錯系統，使其工作於即時(real-time)模式，在操作時監視整個系統的運作狀況。系統偵錯可以由線上(on-line)或離線(off-line)的方式完成，例如在開機前即可經由自我診斷程式確保系統處於正常狀況，或者將不正常的狀況顯示出來以便於故障的排除。

5. 數位控制系統設計

設計一個控制系統可分為下列幾個階段

- 定義系統的規格
- 建立受控體的數學模型與特性分析
- 控制器的分析與設計
- 電腦模擬
- 軟 / 硬體設計
- 系統整合、除錯與性能測試

圖5.1為控制系統設計的流程圖，以下將逐一說明。

5.1 定義系統的規格

如何定義一個控制系統的規格在控制系統設計過程中是一個非常重要卻常受到忽略的項目。定義適當的規格不僅牽涉到功能、價格、市場區分等問題，也與所採用的設計方法、實現的可能性、製造的難易等有著密切的關係。在定義一個系統的規格時，也往往因為考慮這些因素而難以決定，其時程往往佔掉整個研發三分之一的時間。

一個控制系統的規格可分為三類：性能規格(performance specifications)、強韌度規格(robustness specifications)與控制律規格(control law specifications)。

性能規格

性能規格描述一個閉路控制系統所應有的響應，如時域響應的上升時間、超越(overshoot)百分比等，或是頻域響應的頻寬、增益邊界與相位邊界等。一個控制系統的性能規格期望具有下列特性：

- (1) 對干擾具有良好的抵禦能力。
- (2) 對輸入命令能有預期的反應。
- (3) 關鍵訊號不致於太大。

強韌度規格

強韌度是指當受控體改變或是系統之模型化誤差而導致閉路控制系統之性能所受影響的程度。受控體所受到之擾動包括：

- 受控體的某些特性會改變，如元件值的漂移、老化或溫度係數的改變等。例如在化學工業程序控制裡所使用的幫浦，因為使用年限的增長而可能降低至原有效率的70%。
- 受控體可能不精確的予以模型化或是判別，導致此結果的原因可能是無意造成的或無法避免的，如參數的誤差、量測的誤差等等，也可能是有意的，因為系統變化的幅度大，而無法以單一精確的模型化。例如在飛機的動態模型裡，某些結構模式或非線性即予以忽略，以簡化系統的模型，但也因此造成了模型化的誤差。

強韌度規格的定義可能有幾種型式：

- 低微分靈敏度：相對於系統參數的某些迴路的微分量很小。例如電源供應器因元件值改變而造成的輸出電壓變化應很小。
- 保證性能：控制系統即使在受到干擾波動時仍應保持有一定程度的性能表現。例如一個伺服系統即使遭到外力干擾或負載變化其反應時間仍應小於30 msec。

控制律規格

在設計一個控制系統時，有時所採取的控制器設計方法也會受到某些限制，這些限制通常與控制器的實現方法有關，例如：

- 控制器必須具備某些特定型式，如PID。
- 控制器必須是線性且非時變的(LTI)，如利用運算放大器來實現控制器。
- 在某些控制系統可能需要很多個感測器與制動器，每個制動器只需要一個回授信號彼此間有獨立性，但又需要藉由通訊傳遞訊息，而形成一個分權式的控制系統。
- 控制器也許需要用特定的處理器來實現而限制了其複雜度。

在設計一個控制系統時，由於這些設計規格與限制，因此就必須做出妥善的考慮與折衷。

5.2 模型化與分析

在設計一個控制系統的過程中，對於受控體的瞭解是相當重要的。建立受控體的模型通常藉由兩種途徑達成，一是經由物理定律推導控制量與輸出量的關係，建立動態方程式；另一種方法則是經由實際量測各物理量，再經由系統判別(system identification)的方法，建立數學模型。在一般情況下，由於對控制的對象已有相當的認識，因此大多採用前者，對某些複雜的系統，如化工程序控制，由於系統的階數高、非線性、參數量測不易等因素，因此不易由物理關係建立系統的數學模型，在這樣的情況通常採用系統判別的方法，且由於系統的階數高，不易掌握系統的特性與設計控制器，因此常採用模型簡化(model reduction)的方式得到一個足以代表系統重要特性的簡化模型，再進行設計。

在電力電子與馬達驅動的控制領域裡，建立系統的動態數學模型是相當重要的一環。例如在PWM DC-DC 轉換器的模型化方法，就有狀態空間平均化法(state-space averaging method)及等效電路模型等方法，藉此可建立轉換器的小信號模型，以進行時域與頻域響應的分析與設計。

根據所建立的模型，瞭解受控體的階數、極點與零點的個數，是否有右半平面的極點與零點？都是非常重要的。而參數變化對系統穩定度或是極點變化的影響，可藉由靈敏度分析有著進一步的認識。在電力電子轉換系統由於電感電流是否處於連續導通狀態，常會造成系統階數的變化，這在設計控制器時也是值得注意的。

控制器的分析與設計

在建立受控體的模型之後，將根據其特性與系統的規格要求，決定控制系統的架構(configuration)與控制器設計的方法。在古典控制方法裡所採用的根軌跡分析、頻率響應分析等是常用的方法。在一個實際的控制系統，控制器所執行的功能不只是迴路補償控制，其他如回授信號的讀取與估測、訊號監視、序向控制、保護功能、自我檢測與偵錯、操作界面等功能均是控制器所須

完成的工作。如何在系統規劃設計時，即瞭解這些功能需要的優先權與處理速度是相當重要的，根據這些需要才能適當的決定硬體與軟體的架構？是否需要即時多工操作系統的配合？選用何種操作系統 (real-time multi-tasking operating system)？採用何種程式語言？...等等，均需要由系統分析提供設計上考慮的資料。複雜的控制系統設計常需要藉由電腦模擬輔助完成，以下將談談電腦輔助設計與模擬在控制系統設計所扮演的角色。

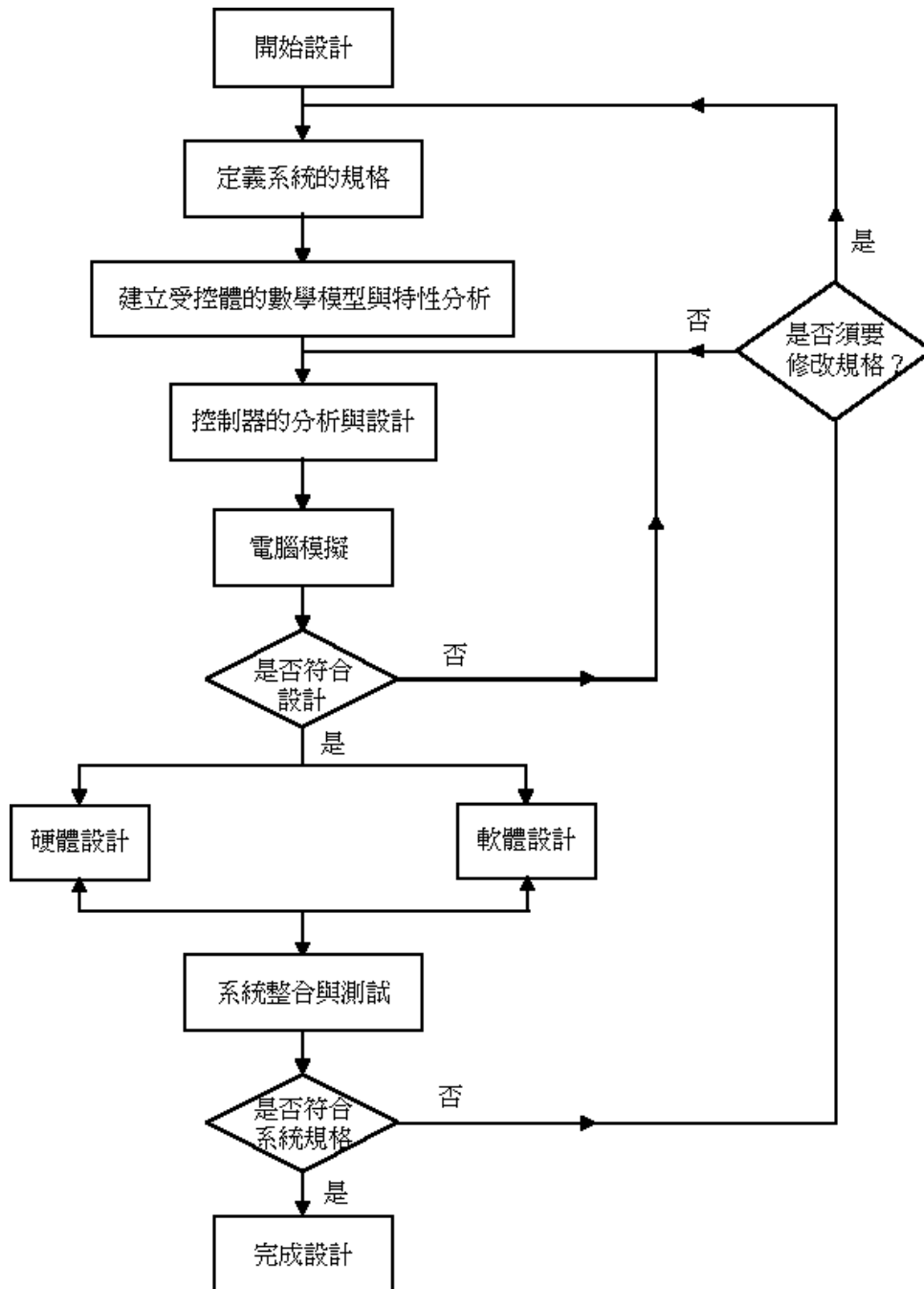


圖5.1 控制系統設計的流程圖

電腦輔助設計與模擬 [22]

電腦輔助設計(CAD)在控制系統的設計過程裡扮演了重要的角色。不論類比或數位控制系統的設計，通常均需藉由電腦模擬完成。模擬的定義，簡單來說即是以模型做為實驗的對象(experimentation with models)。每個模擬程式通常包含了兩個主要部分：一個是模擬對象的數學模型，另一則是實驗的本身。早期著名的控制系統模擬程式即是在1960年代由IBM所發展的CSMP(Continuous System Modeling Program)。早期的控制系統模擬多藉由類比計算機完成，隨著數位計算機的發展，而產生了混合式(hybrid)類比—數位計算機模擬，今日由於數位計算機的長足進步，計算機模擬已大都由數位計算方式完成。控制系統模擬軟體如SIMNON(Lund Institute of Technology, Sweden), ACSL(System Control Technology), MATLAB(New Mexico University), CTRL-C(Systems Control Technology)等等均可在個人電腦上完成複雜動態系統的模擬。

一個電力電子系統的模擬則更為複雜，通常可將其分為三個層次：

- (一) 元件層次(device level)：例如模擬功率元件的開關行為、溫度係數、電流增益等元件等性與其元件設計參數間的關係，這類模擬通常為半導體製程、元件設計工程師用來評估與改良半導體元件的特性。
- (二) 電路層次(circuit level)：模擬電路的動態響應行為，例如功率元件驅動電路與緩振電路(snubber circuit)的設計，這類軟體如SPICE可幫助工程師瞭解電路的一些暫態與穩態行為。
- (三) 系統層次(system level)：模擬電路在控制系統層次，通常相對於開關頻率(switching frequency)為所謂的低頻響應 (low frequency)動態行為，而這也是回授控制系統所欲改良的，電路層次與系統層次不僅在模擬所需要的時間與動態行為上有著顯著的差異，在所採用的模擬模型上也大不相同。

圖5.2是DC-DC轉換器輸出電壓的模擬結果。

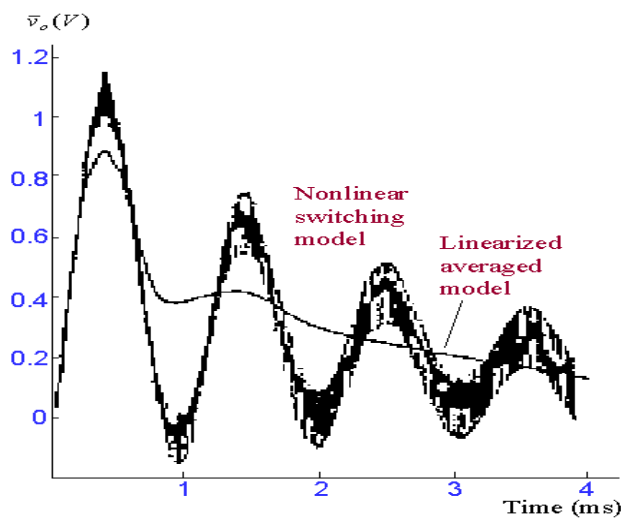


圖5.2 高頻非線性開關模型與線性化的低頻模型所得到的模擬結果

在電力電子回授控制系統的電腦模擬，通常將開關電路部份以一個等效的線性電路取代，如此可將系統的高頻非線性部份去除而得到一個低頻的、平均化的線性模型，再由SPICE等電路分析模擬軟體分析其低頻特性如頻率響應等。

近年來控制系統的設計方法也有了長足的進展，以近代控制理論為基礎的 H_{∞} 理論、 Q 參數設計法，對MIMO線性控制系統都發展出了紮實的理論基礎與系統化的設計方法。在MATLAB與MATRIX系統均建立了相關的工具箱(toolbox)程式庫，以便設計者能結合這些程式庫而進行設計。一些新近發展並已有相當成功應用實例的智慧型控制方法如模糊控制(fuzzy control)、神經網路控制(neural-net control)與專家控制(expert control)等均需藉由計算機完成，而電腦輔助設計與模擬在這設計這類型的控制系統中更是不可或缺的。

硬體設計

以微處理器為基礎的數位控制器硬體設計通常包含兩個部份：一為以微處理器為主的相關電路設計，如中斷界面、記憶體與I/O的定址等；另一為特定的類比/數位電路設計，如信號調整(signal conditioning)、類比/數位信號轉換、功率級觸發電路、保護電路等。這些特定電路設計的範圍主要決定於所採用微處理器的計算速度與功能，某些具有時間緊迫性且不易以軟體完成之控制功能也只好經由特殊設計的硬體完成。經由控制系統的功能需求分析後，即可決定適當的微處理器(DSP可視為一具有特殊功能的微處理器)，選用微處理器考慮的準則包括：

- 字元的長度(8-bit CPU 或16-bit CPU)
- 執行速度(instruction cycle time)
- 乘/除法的計算能力
- 整數或浮點計算
- 功能整合的能力(記憶體與I/O界面等)
- 中斷處理的能力
- 可程式化計時器的個數與功能
- 是否提供工業級 / 軍事級的包裝
- 發展系統的支援能力
- 電路模擬(in-circuit emulation)的能力
- 是否具有多重供貨的來源
- 個人電腦上軟體發展工具的支援能力

在設計微處理器系統時，記憶體與I/O的配置(memory and I/O map)是需要審慎考慮的，ROM與RAM的大小需要根據程式與資料的大小而決定。在一般的工業控制應用中，以組合語言所設計的程式大約數k位元就足夠了，一般通用型的微控器(microcontroller)就足夠用了。但如以C語言設計控制程式，則就需要進行更為審慎而保守的估計，因為程式較易撰寫，而具有較佳的擴充能力，可能導致未來功能擴充時記憶體不足的限制。在大多數的應用中，多以EPROM取代ROM，雖然價格可能昂貴些，但有易於修改的優點，同時量少時價格更為便宜。I/O界面的定址在系統規劃

時也是相當重要的，I/O的分類與定址亦與硬體 / 軟體設計有關，適當的I/O規劃與驅動程式設計，將可使系統的軟體設計更具有模組化設計的特色。

計時器與中斷處理在一個以即時(real-time)多工(multitasking)為前題的數位控制系統是相當重要的。可程式化計時器可由微處理器本身提供，亦可由其週邊裝置提供。在選擇微處理器時其可程式化計時器的個數與操作模式所提供的功能是相當重要的考慮因素。可程式計時器可用來產生定時的軟體中斷(software interrupt) 以便控制程式執行特定的功能如擷取信號或執行PID計算等。

中斷信號的個數與優先權也是必須事先訂定的，軟體中斷與硬體中斷須妥為劃分，可遮蔽式與不可遮蔽式中斷(non-maskable interrupt)的處理在系統設計時即須妥善安排，因為在即時多工的控制系統中其相關的軟體設計是系統能否正常運作的核心部分。

對於一些高速的I/O信號，如計數器、PWM觸發信號等需考慮由特定之硬體完成。為了計算能力的考量，在衡量多處理技術時"主—主"或"主—從"的多處理結構也是需要根據實際的需要而決定。在多處理器系統中，處理器之間的通訊方式是極為重要的，由其耦合緊密的程度，亦即需要彼此交換訊息的多寡而決定軟硬體的通訊方式，在多處理器系統有下列通訊技術：

- 串聯界面(如RS-232)
- 平行I/O界面與握手協定(如Centronics)
- DMA(Direct Memory Access)傳輸
- FIFO(First In First Out)傳輸
- 雙埠記憶體
- 多匯流排配合仲裁邏輯電路

由於積體電路的快速進展與電腦輔助設計的配合發展，使得由早期的觀念以軟體取代硬體逐漸演變成將軟體以硬體方式實現，因而發展出高階硬體描述語言(High-level Hardware Description Language)，這種情況發展的極致，將可產生可程式化的硬體，再經由大型積體電路的製程技術產生具有特定功能的硬體，這種發展趨勢，在發展高性能智慧型控制系統時是值得注意的。

軟體設計

當明定一個控制系統的控制功能之後，實現這些功能的控制法則(control algorithm) 就必須詳細描述出來，再藉由程式語言予以實現。這些控制法則的描述主要藉由流程圖來表示，再配合相關的時程圖 (timing diagram)與中斷機制(interrupt mechanism)。控制法則(algorithm)的描述主要包含三個要項：

- 流程圖(flowchart)
- 時程圖(timing diagram)
- 中斷機制(interrupt mechanism)

微電腦控制軟體由一些"程式模組"(program module)所組成，再以即時的方式依序執行，所謂"依序"即依照中斷所排定的順序，依其優先權與發生的先後而決定。由於必須以"即時"的方式行，因此相關的程式模組必須要在設定的時程內執行完畢，所以執行時間與時程圖分析是相當重

要的。以下逐步說明控制軟體的設計步驟：

1. 選擇程式語言

在電力電子與驅動系統的控制中，迴路控制的計算速度是相當重要的，其中包括回授信號的處理、控制定律的計算與觸發信號的產生。由於在計算上有時間的迫切性，因此通常以組合語言完成。採用組合語言具有執行快速、程式較為精簡、易於發揮硬體的功能、程式撰寫較不受限制等優點，但是卻有程式難以撰寫費時耗力、不易偵錯、難以閱讀、不易說明、擴充能力不佳等缺點，尤其在撰寫大型控制程式時，難以掌握而使軟體發展失控。

高階語言如C、PLM、PASCAL則可與低階組合語言產生互補的效果。在時間緊迫的部份如中斷服務程式可由組合語言撰寫，在時間較為寬鬆的部份如人機界面程式則可由高階語言撰寫。在程式撰寫時模組化的設計是非常重要的，根據經驗在設計一個微電腦控制系統時，在軟體開發所投資的人力與時間通常遠較預估為大，也高於硬體開發的成本，由於工業生產設備的生命週期較長，軟體維修與更新的需求性也就愈大，做好"software documentation"在整個開發過程中是最重要的。

2. 程式撰寫原則

在微電腦控制的研發過程中，發展出"可靠的"控制軟體是第一個要建立的觀念，要如何發展出可靠的軟體有些原則可做為參考：

- 採用模組化的設計方式。
- 訂定流程圖、時程圖與中斷機制的
- 標準描述格式，並以此為依據進行系統設計，並且規定在此項工作未完成時，不得進行細部程式之撰寫。
- 定義系統的資料結構與參數、變數名稱，在名稱的選用上必須定義出一個依循的原則共用變數(global variable)的定義必須明確。
- 訂定程式撰寫的格式，如程式名稱、作者、日期、說明、輸出入變數、呼叫方式、如何加註解等均有依循的原則與範例可供參考。
- 在程式發展之初，即應有另一組工作人員進程式測試之工作，確定測試項目與方法軟體的可靠度測試是必須詳加規劃的，而非僅由實際之操作完成。

3. 選擇字元的長度

在數位信號處理與控制裡，信號解析度的要求決定了基本字元的長度，而這也就直接影響到微處理器的選擇。目前一般的通用型微處理器多為8-位元或16-位元。在數位控制裡，被控制的變數在 ± 1 個位元內跳動，即所謂的量化雜訊(quantization noise)，其所相對的實際物理量變化是決定字元長度主要的依據。例如在一個 $\pm 6000\text{rpm}$ 的迴路速控系統中，16位元的解析度將造成 $6000/(2^{16}-1) = 0.183\text{rpm}$ 的轉速量化誤差，且其幅度的大小可能超過1位元，依據整個迴路中最差的解析度而定。同理，在多迴路控制系統中，外迴路的量化抖動會被內迴路放大，因此當確定外迴路的解析度之後，內迴路的解析度亦應與之相當，以免有放大效應產生。

4. 決定取樣頻率

微電腦控制系統根據時序去執行所程式的功能，因此取樣頻率必須設定。在一個數位控制系統，取樣頻率的選擇不僅影響到控制系統的性能，也會對系統的穩定度造成影響。一般而言取樣頻率愈高，則在單位時間內系統經由回授修正的次數也就愈多，因而能得到較佳的控制品質。但取樣頻率愈高，則在一個取樣週期內所能執行的控制功能也就愈少，同時也會受到微處理器計算能力的限制，因此如何折衷的選擇適當的取樣頻率在設計數位控制系統時是個重要的關鍵。從工程的觀點而言，如何在符合設計規格的要求下，以最低的取樣頻率完成控制系統的設計，是最佳的選擇。

一個數位控制系統如果取樣時間很短且解析度又很高則將趨近於其等效的類比系統，解析度(字元長度)對取樣時間的比值可視為一個優劣的指標。

在一個微電腦多工(multi-tasking)控制系統因為有數個task依據所設定的頻率與優先權先後執行，因此優先權最高且最頻繁的內迴路控制將佔去大部份的計算時間，在設計時必須考慮留時間以供外迴路計算之用。選用較長的取樣時間，則有較多的時間餘域進行計算，亦可考慮用高階語言進程式設計，但亦會造成延遲與訊號的失真，甚至造成系統的不穩定。取樣時間的選擇通常由迴路的反應時間(response time)而定，通常為1/10或者更小。一般而言內迴路較外迴路有更寬的頻寬，也就是說其反應時間更短，因此取樣時間也就更短。

在有些電力電子轉換系統，取樣時間的選擇有時會受到系統的限制，例如在一個3相相位控制(phase control)AC-DC轉換器，因為開關頻率為360 Hz，因此電流或電壓控制迴路的取樣時間也就因此而定為2.77 msec (1/360sec)。

5. 尺規劃(scaling)[8],[36]

在目前一般通用型的微處理器都沒有浮點運算處理器，因此如採用浮點計算就必須以軟體方式完成，但如此一來計算時間將大幅增加因而不可行。因此控制法則的實現在大多數的應用情況均必須以整數計算方式完成。為了防止整數計算時產生的溢位(overflow)就必須對所有的控制變數予以適當的比例調整，也就是尺規劃(scaling)。數位控制系統中整數計算的尺規劃就有如類比控制系統一般，其目的在於保持信號的解析度而防止over-flow與underflow。在數位控制系統中是絕不能讓overflow或underflow發生的，因為會造成計算的錯亂而導致系統的不穩定。以下以一個簡單的例子來說明數位控制系統的尺規劃方法[3]。

在一個直流馬達控制系統，馬達所產生的扭矩與電樞電壓、電流與轉子轉速間的關係可定義為：

$$T_e = \eta \frac{V_a I_a}{\omega}$$

其中 T_e 為馬達產生之扭矩(N*m)， V_a 為電樞電壓(volts)， I_a 為電樞電流(amps)， ω 為轉子的旋轉角頻率(rad/sec)， η 為效率(%)。上式可表示為

$$T_e = \eta \frac{V_a I_a}{N} \cdot \frac{60}{2\pi}$$

其中N為rpm。這些變數與程式中的軟體變數的關係可定義為

$$\begin{aligned} V_a &= \frac{V - V_a}{k_1} & I_a &= \frac{V - I_a}{k_2} \\ N &= \frac{V - N}{k_3} & T_e &= \frac{V - T_e}{k_4} \end{aligned}$$

其中V、Va, V、Ia, V、N,與V、Te 均為軟體變數(software variable), 而k1,k2, k3,k4則為相對的尺規因子(scaling factor)。將這些軟體變數代入原先之扭矩方程式則得

$$V - T_e = 9.557 \frac{k_3 k_4}{k_1 k_2} \cdot \frac{V - V_a \cdot V - I_a}{V - N}$$

假設Va, Ia與N為雙極性信號(有正負值)且由10位元的A/D轉換器取樣, 並假設Va(max) = 200V, Ia(max) = 500A, N(max) = 500rpm, 則

$$k_1 = (2^9 - 1) / 200 = 2.555 \text{ bits/volt}$$

$$k_2 = (2^9 - 1) / 500 = 1.022 \text{ bits/amp}$$

$$k_3 = (2^9 - 1) / 5000 = 0.102 \text{ bits/rpm}$$

假設η=0.85, Te(max) = 162.42Nm, 選擇k4 = 8 bits/Nm (也就是0.125 Nm/bit), 則V、Te (max) = 1299 bits, 亦即12位元的雙極性信號, 將這些尺規因子代入, 則得

$$V - T_e = k \frac{V - V_a \cdot V - I_a}{V - N}$$

其中k=2.9847, 上式在計算時, k為一常數。由上述之說明可知這種尺規劃方法, 主要是考慮各變數之最大值, 再根據所採用之位元數, 找到一個尺規因子, 使這些軟體變數在乘除之間不致於overflow或underflow。

軟體結構與時程

在軟體程式設計時, 必須先確定各軟體功能模組以及其與實際物理系統間的關係, 各功能模組與I/O之間的關係可參考圖5.3。各功能模組均有其執行時優先權高低之分, 其取樣時間亦可決定, 因此這些軟體功能模組可歸類成幾個"工作模組"(task module), 簡稱工作(task)。所謂"工作"可定義為一個完成特定任務的執行程式, 因此所謂的"多工系統"(multi-tasking system)就是指有數個可同時執行的"工作"在運作的系統。這裡所謂的同時是觀念上的, 或者可以說是感覺上的, 實際上這在單處理器系統是分時完成的, 祇不過由於時間非常短暫, 在感覺上無法區分罷了。在控制系統或操作系統(operating system)常有所謂的"即時系統"(real-time system), "即時"可定義為在

所設定的時間內完成交付的任務，簡單來說就是來的及。一個實際的控制系統均可稱之為"即時控制系統"(real-time control system)，因為在實際控制過程中，如果系統要能夠正常運作，控制系統就必須根據在真實時間下所取得的命令與回授信號，再經過控制法則的計算而"即時的"送出修正的命令。

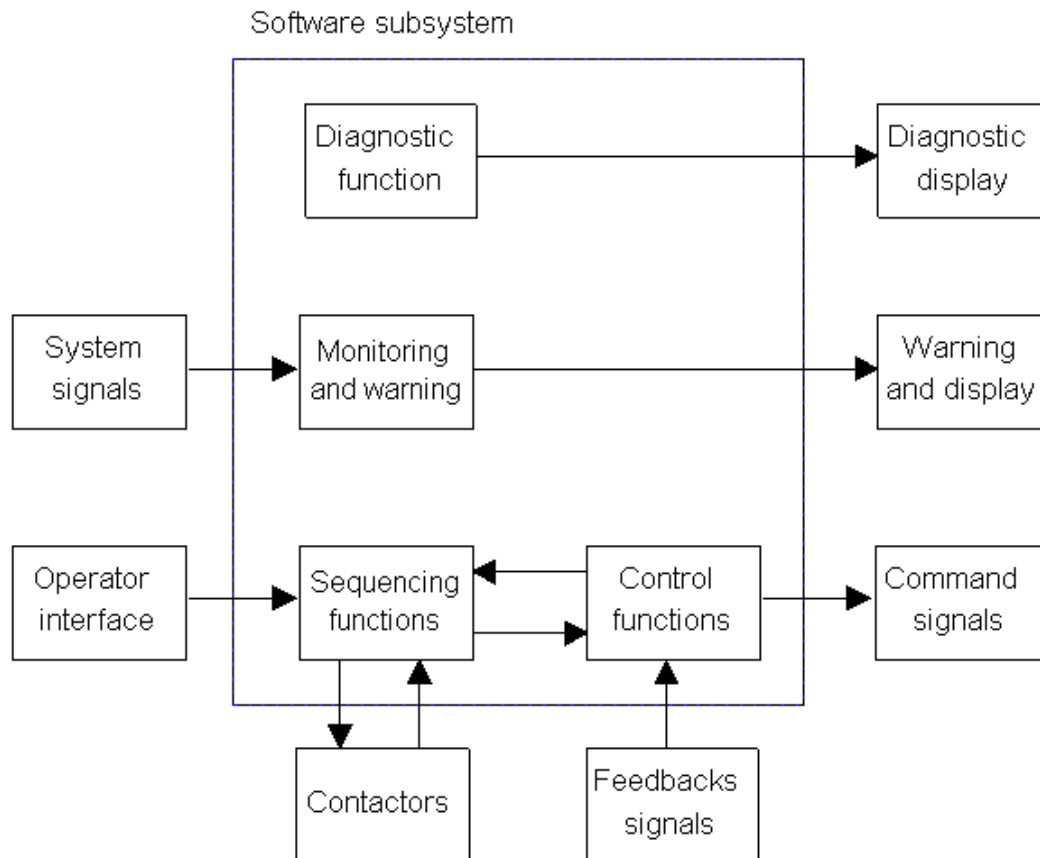


圖5.3 微電腦控制系統軟體功能模組間之關係

圖5.4是一個控制系統工作模組的結構圖，各工作模組的取樣時間亦顯示在圖上。一般而言，功能相似且具有同等級優先權的功能模組均將之編列在同一個工作模組，因此一般在定義上，不同的工作模組通常其取樣時間與優先權均不相同。這些工作模組均在一個稱之為"即時排程"(real-time scheduler, RTS)的控制之下，其為一執行操作系統軟體，它根據各"工作"之優先權排定執行的順序。RTS由硬體時鐘T1所驅動，其週期為優先權最高的Task-1的取樣週期。由圖5.4可看出系統的開機重置(reset)、關機(shutdown)與閒置時間等工作亦排定在工作結構(task structure)圖上。各工作的啟動均由中斷來驅動，這些中斷信號可能是由內部的時鐘所產生，也可能由外部的硬體產生。因此整個系統可以稱之為是一個以中斷為驅動信號的即時多工控制系統。各工作的時程圖(timing diagram)如圖5.5所示。其中Task-1必須在T1的時段內執行完成，如果一個優先權較低的工作已在進行，則到了T1的時刻，它會被暫停(suspend)，先執行Task-1，完成Task-1後，再將原先暫停的工作繼續(resume)執行。從圖5.5可看出Task-2曾被暫停過一次，如標記所示，Task-4被暫停3次

，如標記、所示，當工作Task-4完成後，系統完成了一個循環。在一個循環中，各工作所執行的軟體功能亦列於工作時程圖上，這在系統分析設計時是非常重要的。

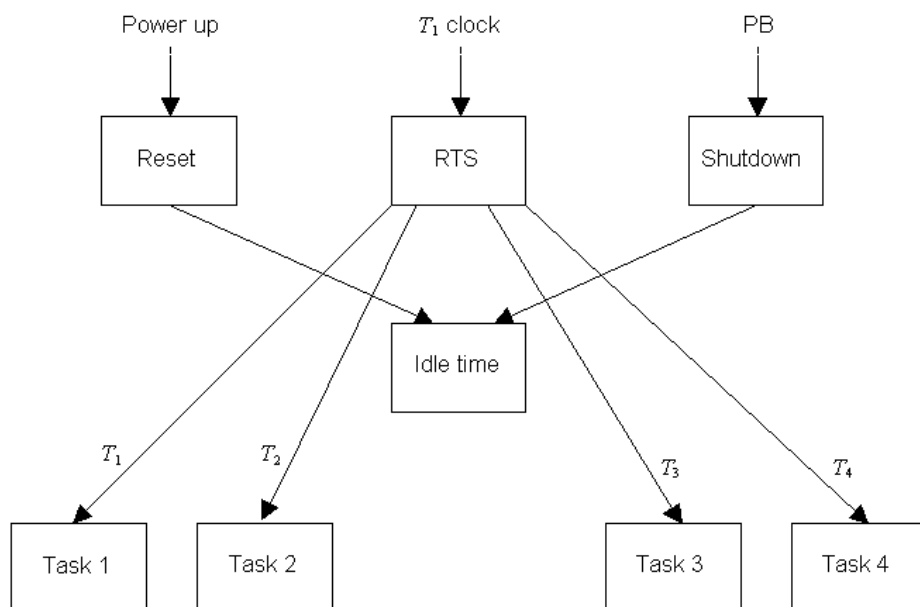


圖5.4 工作模組的結構圖

計算負載因數(computation loading factor)定義為CPU忙碌時段與最長的取樣時段的比值，以圖5.5而言：

$$\text{計算負載因數} = \Sigma \text{計算時間} / T_4$$

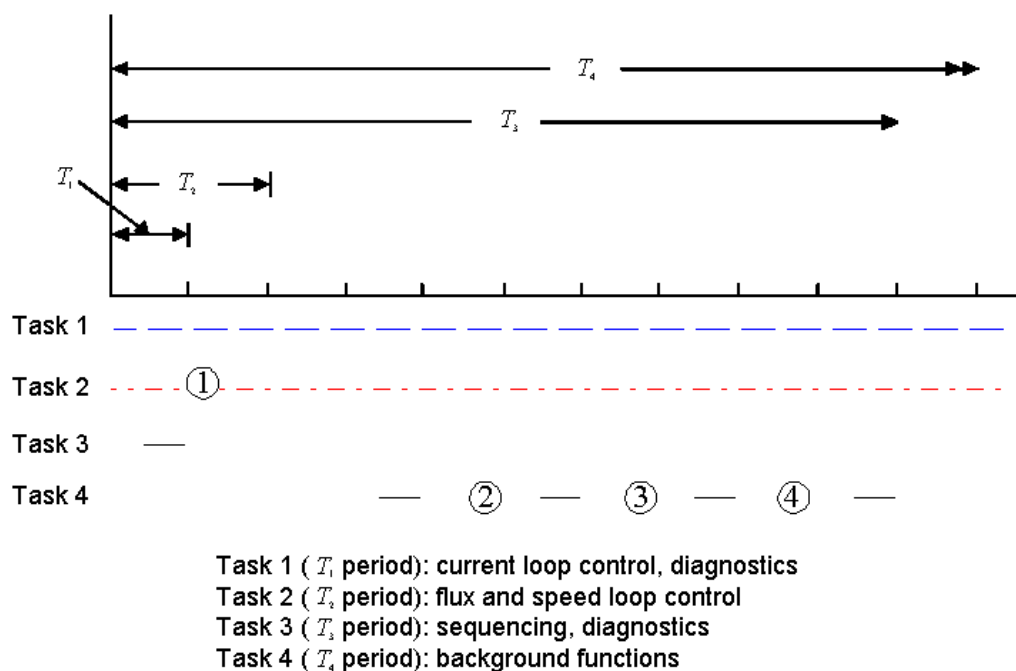


圖5.5 工作時程圖

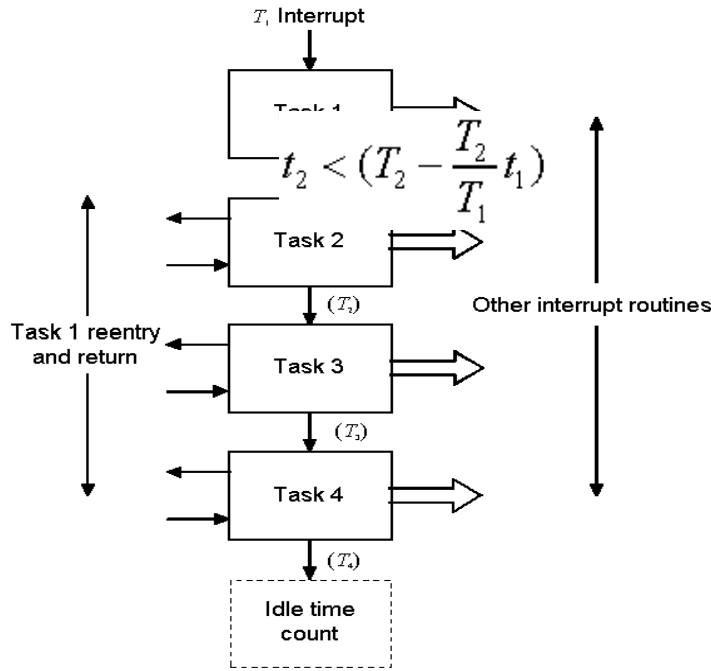


圖5.6 RTS的控制流程圖

各工作模組的取樣時間通常定義為整數倍，因為易於分析與實現，例如

$$n_1 = \frac{T_2}{T_1} \quad n_2 = \frac{T_3}{T_2} \quad n_3 = \frac{T_4}{T_3}$$

其中 n_j 為整數，如果這四個工作的執行時間分別為 t_1, t_2, t_3 與 t_4 則

$$\begin{aligned} &= \frac{1}{T_4} \left[\frac{T_4}{T_1} t_1 + \frac{T_4}{T_2} t_2 + \frac{T_4}{T_3} t_3 + t_4 \right] \\ &= \frac{t_1}{T_1} + \frac{t_2}{T_2} + \frac{t_3}{T_3} + \frac{t_4}{T_4} \end{aligned}$$

若 μ_a = 計算負載因數, 則上式可改寫為

$$\begin{aligned} \mu_a &= \frac{1}{T_1} \left[t_1 + \frac{T_1}{T_2} t_2 + \frac{T_1}{T_3} t_3 + \frac{T_1}{T_4} t_4 \right] \\ &= \frac{1}{T_1} \left[t_1 + \frac{t_2}{n_1} + \frac{t_3}{n_1 n_2} + \frac{t_4}{n_1 n_2 n_3} \right] \end{aligned}$$

μ_a 應小於1，也就是說

$$t_1 + \frac{t_2}{n_1} + \frac{t_3}{n_1 n_2} + \frac{t_4}{n_1 n_2 n_3} < T_1$$

各工作模組執行時間的限制為

$$\begin{aligned}
 t_1 &< T_1 \\
 t_2 &< (T_2 - \frac{T_2}{T_1}t_1) \\
 t_3 &< (T_3 - \frac{T_3}{T_1}t_1 - \frac{T_3}{T_2}t_2) \\
 t_4 &< (T_4 - \frac{T_4}{T_1}t_1 - \frac{T_4}{T_2}t_2 - \frac{T_4}{T_3}t_3)
 \end{aligned}$$

圖5.6為RTS控制各工作模組時程的流程圖，時段信號 T2,T 3與T4由時鐘信號T1，經由計時器產生。Task-1由時鐘信號T1所驅動，其他各工作模組則由相關之驅動時鐘信號或其他中斷信號所驅動，再由事先所設定之優先權決定執行的先後順序。某些不是經常發生的中斷，可能具有更高的執行優先權，如緊急停機或保護開關被觸發時。

6. 程式發展與模擬測試

發展系統

在開發微電腦控制系統時，建立其相關的發展工具是不可或缺的，其配置如圖5.7所示，通常包含了：

- 微電腦系統(通常為PC)
- EPROM 燒錄器
- ICE模擬器或 ROMEmulator
- 印表機
- 軟體支援
- 邏輯分析儀

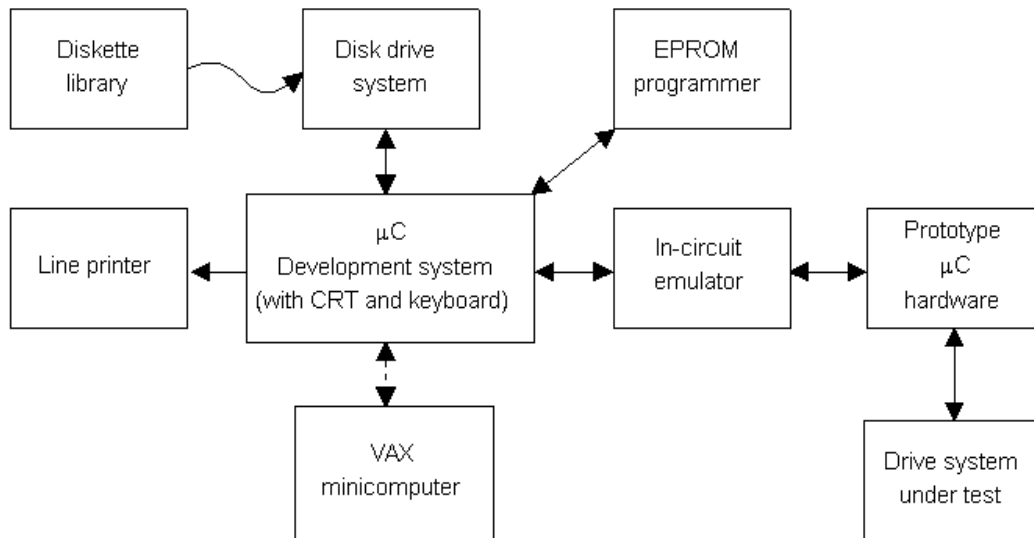


圖5.7 微電腦發展系統與週邊設備

程式發展流程

程式發展流程如圖5.8所示。先由程式編輯器(Editor)產生程式原始檔案(program source file)，再經過assembler或compiler產生目標碼(object code)，再經過linker與locator將程式與程式庫連結後再定止，產生absolute object module，之後即可燒錄在EPROM或download到ICE上進行測試與偵錯。

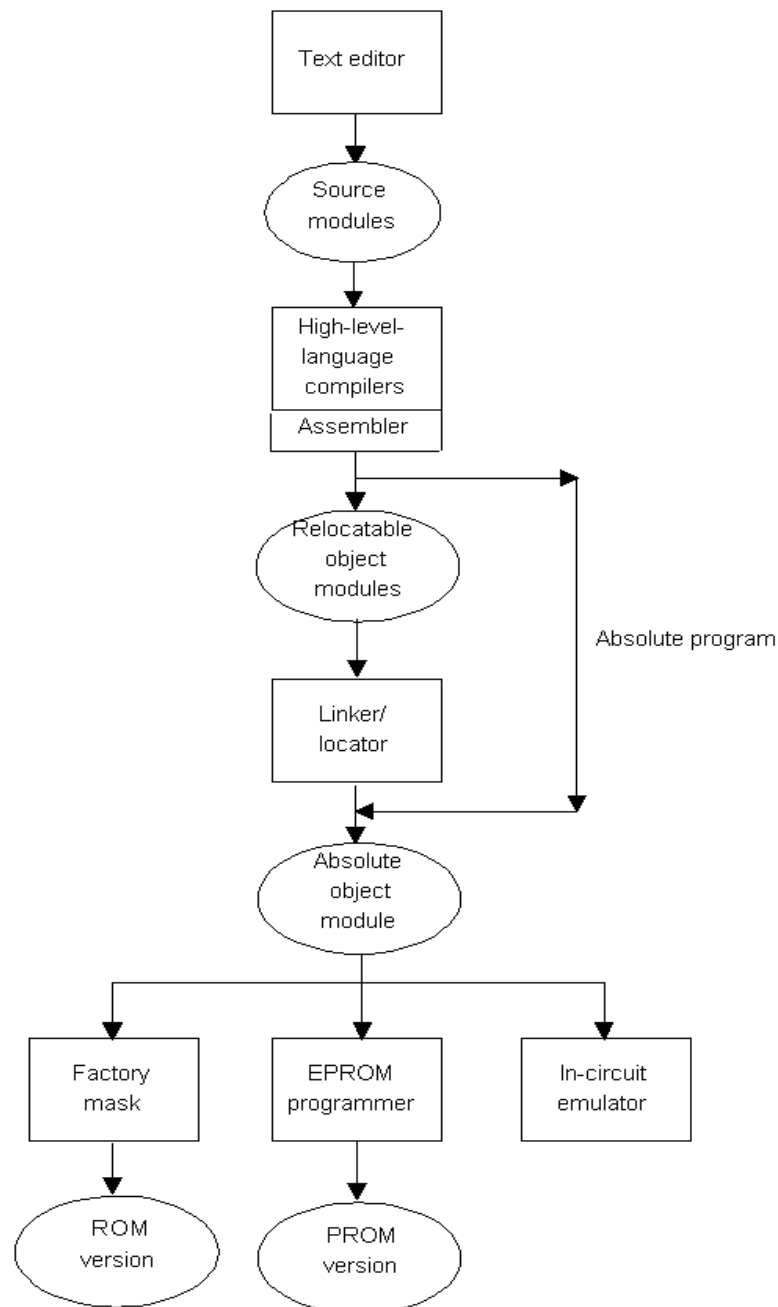


圖5.8 程式發展的流程圖

參考文獻

Introduction to Microcomputer/DSP-Based Digital Control

- [1] K. J. Astrom and B. Wittenmark, Computer Controlled Systems: Theory and Design, Prentice-Hall, Inc., 1984.
- [2] R. J. Bibbero, Microprocessors in Instruments and Control, New York, NY: Wiley, 1977.
- [3] Bose, B. K., "Introduction to microcomputer control," from Microcomputer Control of Power Electronics and Drives, IEEE Press, pp. 3-22, 1987.
- [4] B. K. Bose, Editor, Microcomputer Control of Power Electronics and Drives, IEEE Press, 1987.
- [5] F. F. Driscou, Microprocessor-Microcomputer Technology, Van Nostrand, New York, NY, 1983.
- [6] DC Motors, Speed Controls, Servo Systems, including Optical Encoders, An Engineering Handbook by Electro-Craft Corporation, Hopkins, MN, Fifth Edition, 1980.
- [7] G. F. Franklin, J. D. Powell, and M. L. Workman, Digital Control of Dynamic Systems, Addison-Wesley Publishing Company, 1990.
- [8] P. Katz, Digital Control Using Microprocessors, Prentice-Hall, Inc., 1981.
- [9] L. A. Leventhal, Introduction to Microprocessors: Software, Hardware, Programming, Prentice-Hall, Inc., 1978.
- [10] N. Nakano and S. Hara, Microprocessor-based Control Systems, D. Reidel Publishing Company, Amsterdam, The Netherlands, 1986.
- [11] Digital Control Applications with the TMS320 Family: Selected Application Notes, Texas Instruments, 1991.

DSP Hardware/Software and Development Support

- [12] A. Schwarte and H. Hanselmann, "The programming language DSPL," PCIM, June 25-28, 1990.
- [13] TMS320C1x User's Guide, Texas Instruments, 1989.
- [14] TMS320C14/E14 User's Guide, Texas Instruments, 1988.
- [15] TMS320C2x User's Guide, Texas Instruments, 1990.
- [16] TMS320C3x User's Guide, Texas Instruments, 1990.
- [17] TMS320C4x User's Guide, Texas Instruments, 1991.
- [18] TMS320C5x User's Guide, Texas Instruments, 1990.
- [19] Digital Signal Processing Applications with TMS 320 Family: Volume I, Texas Instruments, 1986.
- [20] Digital Signal Processing Applications with TMS 320 Family: Volume II, Texas Instruments, 1990.
- [21] Digital Signal Processing Applications with TMS 320 Family: Volume III, Texas Instruments, 1990.

Implementation of Digital Controller

- [22] D. M. Auslander and C. H. Tham, *Real-Time Software for Control: program examples in C*, Prentice Hall, 1990.
- [23] K. J. Astrom and H. Steingrimsson, "Implementation of a PID controller on a DSP," from: *Digital Control Applications with the TMS320 Family: Selected Application Notes*, Texas Instruments, 1991.
- [24] C. P. Diduch and R. Doraiswami, "Design of a microcomputer based servomechanism controller," *Comput. & Elect. Engng.*, vol. 13, no. 1, pp. 29-40, 1987.
- [25] G. F. Franklin, J. D. Powell, and M. L. Workman, *Sample Rate Selection*, Chap. 10 of *Digital Control of Dynamic Systems*, Addison-Wesley Publishing Company, 1990.
- [26] K. H. Gurubasavaraj, "Implementation of a self-tuning controller using digital signal processor chips," *IEEE Control Systems Magazine*, June 1989.
- [27] Hanselmann, H., "Implementation of digital controllers — a survey," *Automatica*, vol. 23, no. 1, pp. 7-32, 1987.
- [28] P. Katz, "Mechanization of control algorithms on microcontrollers," Chapter 5 from: *Digital Control Using Microprocessors*, Prentice-Hall, Inc., 1981.
- [29] H. T. Nagle and V. P. Nelson, "Digital filter implementation on 16-bit microcomputers," *IEEE Micro Magazine*, pp. 23-41, Feb. 1981.
- [30] C. Slivinsky and J. Borninski, "Control system compensation and implementation with the TMS32010," *Digital Signal Processing Application Report*, Texas Instruments, 1985.
- [31] I. A. Stevenlindquist, "Digital signal processors simplifying high-performance control," *Machine Design*, Sept. 10, 1987.
- [32] J. Stokes and G. R. L. Sohie, "Implementation of PID controllers on the Motorola DSP56000/DSP56001," Motorola Inc., Application Note, 1989.
- [33] T. Tsuchiya, "Improved direct digital control algorithm for microprocessor implementation," *IEEE Tran. Automat. Contr.*, vol. 27, no. 2, pp. 295-306, April 1982.

Quantization, Scaling, Roundoff, and Limit Cycles

- [34] T. A. C. M. Claasen, W. F. G. Mecklenbrauker, and J. B. H. Peek, "Effects of quantization and overflow in recursive digital filters," *IEEE Trans. on Acoustics, Speech, and Signal Processing*, vol. 24, no. 6, pp. 517-529, Dec. 1976.
- [35] M. E. Ahmed and P.R. Belanger, "Scaling and roundoff in fixed-point implementation of control algorithms," *IEEE Trans. on Ind. Electron.*, vol. 31, no. 3, pp. 228-234, August 1984.
- [36] M. E. Ahmed and P. R. Belanger, "Limit cycles in fixed-point implementation of control algorithms," *IEEE Trans. on Ind. Electron.*, vol. 31, no. 3, pp. 235-242, August 1984.

[37] V. C. Jaswa, "Quantization noise and limit cycling in digital closed-loop systems," IEEE Trans. on Ind. Electron., vol. 31, no. 2, pp. 149-151, May 1984.

Digital Control System Development Tools

[38] Control System Development Using Dynamic Signal Analyzers, HP Application Note 243-2, Hewlett-Packard, 1984.

[39] M. Jamshidi and C. J. Herget, Computer-Aided Control Systems Engineering, Elsevier Science Publishing Company, Inc., North-Holland, 1985.

[40] M. Jamshidi, M. Tarokh, and B. Shafai, Computer-Aided Analysis and Design of Linear Control Systems, Prentice-Hall, Inc., 1992.

[41] PC-MATLAB: IBM-PC Based Control System Design ToolBox, The MathWorks Inc., # 158, Woodland St. Sherborn, MA 01770, 1987.

[42] MATRIX User's Guide, Integrated System Inc., 2500 Mission College Blvd. Santa Clara, CA 95054, 1987.

[43] B. Shahian and M. Hassul, Control System Design Using MATRIXX, Prentice Hall, Inc., 1992.

[44] PC-SIMNON: IBM-PC Based Nonlinear Dynamical System Simulator: User's Guide, The MathWorks Inc., # 158, Woodland St. Sherborn, MA 01770 Tel (617) 653-1415, 1987.

[45] SYSL/M: Continuous System Simulation Language, User's Manual, E2 Consulting, P.O. Box 1182, Poway, CA 92064, Tel (619) 578-4057, 1987.

[46] Y. Y. Tzou, S. Y. Hwang, H. J. Wu, and I. K. Wang, "Design and implementation of an interactive digital controller development system," IEEE IECON Conf. Rec., pp. 785-791, Nov. 1989.

Design of Digitally Controlled DC Drive

[47] J. P. Jiang and R. S. Marleau, "Digitally controlled DC drive motors," IEEE Trans. on Ind. Appl., vol. 18, no. 6, pp. 728-735, Nov./Dec. 1982.

[48] W. Lord and J.H. Hwang, "DC servomotors: modeling and parameter determination," IEEE Trans. Ind. Appl., vol. 13, no. 3, pp. 234-243, May/June 1977.

[49] S. Meshkat, "A servo system parameter identification for optimum compensation design," Proceedings of MOTORCON, pp. 348-354, April 1985.

[50] G. A. Perdikaris and K. W. VanPatten, "Computer schemes for modeling, tuning, and control of dc motor drive systems," PCI Proceedings, vol. 3, pp. 83-96, March, 1982.

[51] R. A. Schultz, "A frequency response method for determining the parameters of high-performance dc motors," IEEE Trans. on Industrial Electronics, vol. 30, no. 1, pp.39-42, Feb.,1983.

[52] M. R. Stojic, "Design of the microprocessor-based digital system for DC motor speed control," IEEE Trans. on Ind. Electron., vol. 31, no. 3, pp. 243-248, August 1984.

DSP: Applications in Motion Control

[53] J. M. Corliss and R. Neubert, "DSP helps keep disk drives on track," Computer Design, June 15, 1988.

[54] Y. Dote, "Application of modern control techniques to motor control," IEEE Proc., vol. 76, no. 4, pp. 438-454, April 1988.

[55] Y. Dote, Servo Motor and Motion Control Using Digital Signal Processors, Prentice Hall and Texas Instruments Digital Signal Processing Series, 1990.

[56] S. Meshkat, "Application of Kalman filtering in motion control using TMS320C25," pp. 185-203, from Digital Control Applications with the TMS320 Family: Selected Application Notes, Texas Instruments, 1991.

[57] H. Steingrimsson and K. J. Astrom, "DSP implementation of a disk drive controller," pp. 239-253, from Digital Control Applications with the TMS320 Family: Selected Application Notes, Texas Instruments, 1991.

[58] R. Kruk and J. Scannell, "Motion controller employs DSP technology," PCIM, 1988.

DSP: Applications in PWM Inverters

[59] S. R. Bowes and M. J. Mount, "Microprocessor Control of PWM Inverters," IEE Proc. Electr. Power Appl., Pt. B, vol. 128, no. 6, pp. 293-305, 1981.

[60] R. J. Chance and J. A. Taufiq, "A TMS32010 based near optimized pulse width modulated waveform generator," Third International Conference on Power Electronics and Variable Speed Drives, Conference Publication Number 291, July 1988.

[61] S. Vadivel, G. Bhuvaneshwari, and G. Sridhara, "A unified approach to the real-time implementation of microprocessor-based PWM waveforms," IEEE Trans. on Power Electronics, vol. 6, no. 4, pp. 565-575, Oct. 1991.

[62] M. Varnovitsky, "A microcomputer-based control signal generator for a three-phase switching power inverter," IEEE Trans. Ind. Appl., vol. 19, no. 2, pp. 228-234, Mar./Apr. 1983.

DSP: Applications in AC Drives

[63] R. Dhaouadi, N. Mohan, and L. Norum, "Design and implementation of an extended Kalman filter for the state estimation of a permanent magnet synchronous motor," Proceedings of Power Electronic Specialists Conference, June 1990.

- [64] S. Meshkat and I. Ahmed, "Using DSPs in AC induction motor drives," *Control Engineering*, 1988.
- [65] R. Lessmeier, W. Schumacher, and W. Leonhard, "Microprocessor-controlled AC-servo drives with synchronous or induction motor: Which is preferable ?" *IEEE Transactions on Industry Applications*, vol. 22, no. 5, Sept./Oct. 1986.
- [66] B. K. Bose and P. M. Szczesny, "A microcomputer-based control and simulation of an advanced IPM synchronous machine drive system for electric vehicle propulsion," *IEEE Transactions on Industrial Electronics*, vol. 35, no. 4, Nov. 1988.
- [67] N. Matsui and H. Ohashi, "DSP-based adaptive control of a brushless motor," *IEEE IAS Annual Meeting, Conf. Rec.*, 1988.
- [68] N. Matsui, N. Akao, and T. Wakino, "High precision torque control of reluctance motors," *IEEE IAS Annual Meeting, Conf. Rec.*, 1989.
- [69] S. K. Sul, B. H. Kwon, J. K. Kang, K. Y. Lim, and M. H. Park, "Design of an optimal discrete current regulator," *IEEE IAS Annual Meeting, Conf. Rec.*, pp. 348-395, 1989.
- [70] I. Takahashi and M. Iwata, "High resolution position control under 1 sec. of an induction motor with full digitized methods," *IEEE IAS Annual Meeting, Conf. Rec.*, 1989.

[DSP: Applications in UPS](#)

- [71] K. P. Gokhale, "Microprocessor control of PWM inverters for sinusoidal output waveform synthesis," Ph.D Dissertation, Univ. of Missouri-Columbia, 1985.
- [72] A. Kawamura and K. Ishihara, "Real time digital feedback control of three-phase PWM inverters with quick transient response," *IEEE IAS Annual Meeting Conf. Rec.*, pp. 728-734, 1988.
- [73] I. Kubo, Y. Ozawa, R. Nakatsuka, A. Shimizu, "A fully digital controlled UPS using IGBT's," *IEEE IAS Annual Meeting Conf. Rec.*, pp. 1042-1046, 1991.
- H. I. Cha, S. S. Kim, M. G. Kang, and Y. H. Chung, "Real-time digital control of PWM inverter with PI compensator UPS," *IEEE IAS Annual Meeting Conf. Rec.*, pp. 1124-1128, 1990.