

# KHẢ NĂNG CHẾ TẠO BỘ ĐIỀU KHIỂN LOGIC KHẢ TRÌNH PLC HỌ ĐƠN GIẢN

**Nguyễn Văn Sơn**

*Trường Đại học Thủ Dầu Một*

## TÓM TẮT

*Bộ điều khiển logic khả trình PLC (Programmable Logic Controller) là một khí cụ điện tử quan trọng không thể thiếu trong một dây chuyền tự động hóa trong công nghiệp. PLC được sản xuất theo chuẩn công nghiệp về mức logic điện lối vào (mức logic “1” là 24V, mức logic “0” là 0V), các lối ra là trạng thái on/off có thể là relay, triac hoặc transistor, và theo chuẩn về lắp đặt cơ khí. Họ PLC đơn giản như CPM1 của hãng Omron hoặc series FX0 của Mitsubishi có tổng số đầu vào và ra không quá 50, có tập lệnh căn bản. Họ PLC đơn giản như đã nêu đã được tác giả chế tạo thử phần cứng và phần mềm, cho thấy đáp ứng được các chỉ tiêu kỹ thuật của PLC và được giới thiệu trong bài báo này. PLC có cấu trúc phần cứng khá đơn giản, chế tạo PLC quan trọng là ở phần mềm lập trình cho PLC và các công cụ biên dịch. PLC, hiện nay chưa có đơn vị nào trong nước sản xuất. PLC nhập ngoại khá đắt, nên đặt vấn đề sản xuất PLC có ý nghĩa về mặt kinh tế và nâng cao trình độ sản xuất khí cụ điện tử trong nước.*

**Từ khóa:** PLC, chế tạo PLC, thiết kế PLC.

\*

## 1. Giới thiệu

PLC là khí cụ điện tử tự động hóa giúp kỹ thuật viên dễ lập trình và ứng dụng. Trước khi vi xử lý, vi điều khiển, PLC ra đời, hệ thống tự động hóa sử dụng các linh kiện là relay, bộ định thời (timer), bộ đếm (counter), với các tiếp điểm thường đóng và các tiếp điểm thường mở được lắp đặt trong tủ điện với 2 thanh cái đặt ở bên trái và bên phải, đây chính là hình ảnh của ngôn ngữ lập trình ladder sử dụng cho PLC ra đời sau này. Sử dụng vi xử lý, vi điều khiển cho hệ thống tự động hóa là quá phức tạp cho kỹ thuật viên để thiết kế phần cứng và lập trình bằng ngôn ngữ assembler hoặc C. Sự ra đời của PLC giúp các kỹ thuật viên dễ tiếp cận, dễ lập trình và ứng dụng, nhất là các kỹ thuật viên đã có kinh nghiệm thiết kế hệ

thống tự động hóa cơ điện sử dụng relay, bộ định thời, bộ đếm như đã nói đến ở trên. Ngôn ngữ lập trình cho PLC là ngôn ngữ ladder, ngôn ngữ câu lệnh (instruction program). Ngôn ngữ ladder sử dụng phổ biến hơn là ngôn ngữ trực quan để lập trình, các phần tử của ngôn ngữ gồm: tiếp điểm thường đóng, tiếp điểm thường mở, bộ định thời, bộ đếm. Đây cũng chính là các phần tử tự động hóa thế hệ đầu mà các kỹ thuật viên tự động hóa đã quá quen thuộc.

Chế tạo PLC gồm 2 phần: phần cứng và phần mềm. Cấu trúc phần cứng: trung tâm là vi điều khiển hoặc vi xử lý, bộ nhớ, mạch vào, mạch ra, phần giao tiếp với máy tính để lập trình; phần nguồn. Phần mềm gồm: xây dựng một giao diện đồ họa để soạn thảo lập trình cho PLC bằng ngôn ngữ ladder hoặc

soạn thảo bằng ngôn ngữ câu lệnh; xây dựng một công cụ chuyển đổi từ ngôn ngữ ladder sang ngôn ngữ câu lệnh; xây dựng một công cụ chuyển đổi từ ngôn ngữ instruction sang dạng hợp ngữ (Assembly program); viết trình biên dịch để biên dịch từ hợp ngữ sang file nhị phân để có thể nạp vào bộ nhớ chương trình cho vi điều khiển; xây dựng một công cụ cho phép nạp file nhị phân hoặc file hex từ máy tính vào PLC.

**2. Nội dung**

*2.1. Cấu trúc phần cứng của PLC*

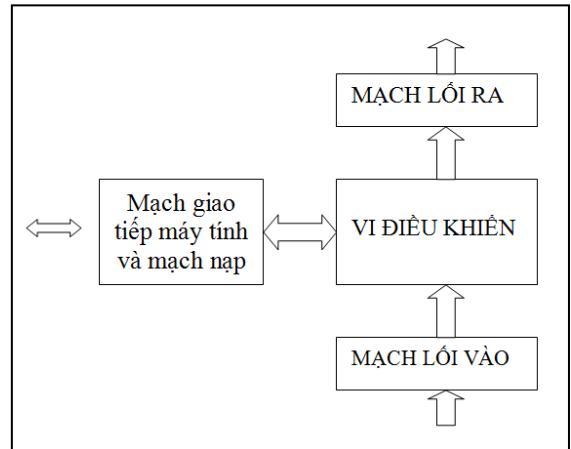
Trung tâm bộ PLC là vi điều khiển (MCU: Micro-Controller Unit), các lối vào và mạch lối ra PLC liên kết với các cổng vào – ra của vi điều khiển thông qua các linh kiện ghép quang, để cách ly về phương diện điện, tăng khả năng chống nhiễu. Ngoài ra dùng linh kiện ghép quang còn cho phép tạo ra nhiều đường dây chung (common) cho mạch ra hoặc mạch vào, tiện lợi cho việc sử dụng. Mức logic điện lối vào là 0V và 24V (mức logic “1” là 24V, mức logic “0” là 0V). Lối ra các bộ PLC trên thị trường có 3 loại:

- Lối ra là tiếp điểm relay, cho phép dòng điện tải lối ra là một chiều hoặc xoay chiều, dòng qua tiếp điểm cho phép không quá 500 mA, đây là loại lối ra phổ biến nhất.
- Lối ra là transistor, chỉ cho phép dòng điện tải lối ra là một chiều, dòng điện

không quá 400 mA, điện áp không quá 40VDC, đây là loại lối ra ít được phổ biến, nhưng lại có ưu điểm về thời gian chuyển mạch.

- Lối ra triac, cho phép dòng điện qua là xoay chiều, dòng điện tối đa cho phép không quá 1A, điện áp không 400VAC, đây cũng là loại lối ra phổ biến, thích hợp để điều khiển cho các relay xoay chiều hoặc các khởi động từ.

Nguồn điện cấp cho PLC là 24VDC hoặc 220VAC


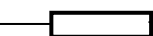
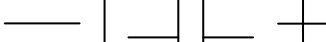


**Hình1.** Cấu trúc phần cứng bộ lập trình PLC

*2.2. Các phần tử ngôn ngữ ladder*

Ngôn ngữ ladder là ngôn ngữ trực quan, các phần tử của ngôn ngữ là: tiếp điểm thường mở (NO: Normal Open), tiếp điểm thường đóng (NC: Normal Close), bộ định thời (T: Timer), bộ đếm (C: Counter), hộp lệnh (CB: Command Box) và dây nối.

Loại phần tử	Ký hiệu	Tên phần tử
Tiếp điểm thường mở		X0, X1,... tiếp điểm lối vào Y0, Y1,... tiếp điểm của lối ra T0, C1,... tiếp điểm của timer, counter, hoặc M0, M1,... tiếp điểm của relay trung gian
Tiếp điểm thường đóng		X0, X1, ... Y0, Y1, ... T0, C1, ... M0, M1, ...
Bộ định thời		T0 K100 <sup>+</sup> , T1 K500 <sup>+</sup> ...
Bộ đếm		C0 K100 <sup>+</sup> , C1 K500 <sup>+</sup> ...

Các ngõ ra hoặc relay trung gian		Y0, Y1, ... M0, M1, ...
Hộp lệnh		RST ( Reset ), SET END
Các loại dây nối		(các dây nối không có tên)

**Bảng 1.** Các phần tử của PLC, ký hiệu và cách đặt tên  
(\* các hằng số K chỉ là ví dụ)

Bản chất bộ đếm và bộ định thời là như nhau, bộ định thời đếm xung thời gian, bộ đếm đếm sự kiện lỗi vào. Tên bộ định thời: T0, T1,... ; tên bộ đếm: C0, C1,... hằng số K của bộ định thời, ví dụ K100, có nghĩa thời gian định thời là 100 đơn vị thời gian, đơn vị thời gian có thể là 1 ms, 10 ms hoặc 100 ms tương ứng với các nhóm tên của bộ định thời. Các hằng số K của bộ đếm là số xung định trước để bộ đếm đếm tới số xung này sẽ tác động các tiếp điểm của nó.

Số tiếp điểm (thường đóng hay thường mở) của một phần tử là không hạn chế, có nghĩa là có thể sử dụng bao nhiêu lần cũng được, đây là điểm khác nhau cơ bản giữa ảo và thực, với một relay thực số cặp tiếp điểm tối đa là 4. Tùy thuộc vào hãng chế tạo mà tên các phần tử trên chương trình câu lệnh có khác nhau, cũng tùy thuộc vào họ bộ lập trình PLC cũng có thể có thêm các phần tử khác.

**2.3. Cấu hình PLC thiết kế thử nghiệm**

Đây là cấu hình thử nghiệm, các bộ định thời có đơn vị thời gian là 1ms, hằng số K tối đa là 65535, có nghĩa thời gian định thời bé nhất là 1ms và cao nhất là 65535 ms = 65 s = 1 phút + 5 s. Đối với bộ đếm: số đếm tối đa là 65535.

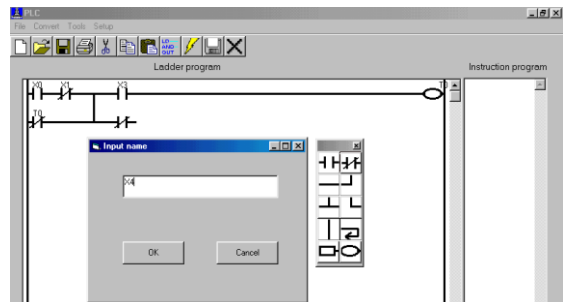
Các phần tử	Số lượng	Ký hiệu
Đầu vào	08	X0 ...X7
Đầu ra	08	Y0 ... Y7

Relay trung gian	16	M0 ... M15
Bộ định thời ( Timer )	08 ( 16 bit )	T0 ... T7
Bộ đếm ( Counter)	08 ( 16 bit )	C0 ... C7

**Bảng 2.** Cấu hình bộ PLC chế tạo thử

**2.4. Xây dựng giao diện đồ họa soạn thảo chương trình ladder**

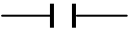
Để xây dựng giao diện đồ họa soạn thảo chương trình ladder chúng tôi sử dụng component flexgrid của ngôn ngữ lập trình Visual Basics, hình ảnh trên ô flexgrid là hình ảnh các phần tử của ngôn ngữ ladder như trong hình 2.



**Hình 2.** Giao diện đồ họa của phần mềm

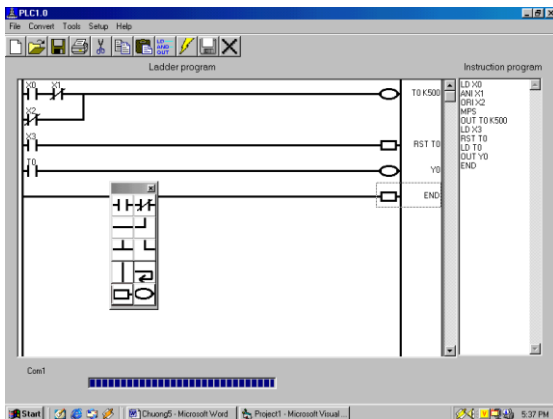
**2.5. Xây dựng công cụ chuyển đổi từ chương trình ladder sang chương trình câu lệnh**

Để chuyển chương trình ladder sang chương trình instruction, chương trình ladder được quét từ trái sang phải và từ trên xuống dưới, tại mỗi vị trí quét, xét vị trí của phần tử đang chuyển đổi sơ đồ ladder theo bảng dưới đây

Ladder Program	Instruction program Trường hợp 1	Instruction program Trường hợp 2	Instruction program Trường hợp 3
X0 	LD X0 - ở vị trí đầu thanh cái - ở đầu một block	AND X0 - nối tiếp sau một phần tử khác	OR X0 - nối song song dưới một phần tử khác
	LDI X0	ANI X0	ORI X0

	- ở vị trí đầu thanh cái - ở đầu một block	- nối tiếp sau một phần tử khác	- nối song song dưới một phần tử khác
	ANB Khi 2 block mắc nối tiếp nhau	(Không có)	(Không có)
	ORB Khi 2 block mắc song song nhau	(Không có)	(Không có)
	OUT Y0 OUT T...K... OUT C... K...	(Không có)	(Không có)
	END RST T... RST C...	(Không có)	(Không có)

**Bảng 3.** Tóm tắt các quy luật chuyển đổi từ ladder chương trình sang chương trình câu lệnh

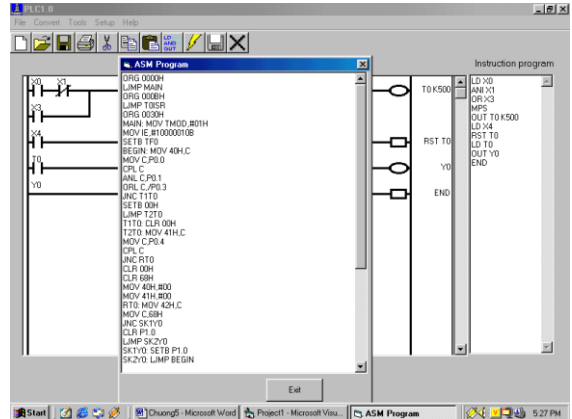


**Hình 3.** Chương trình câu lệnh được chuyển đổi từ chương trình ladder và đặt vào cửa sổ instruction program.

**2.6. Công cụ chuyển đổi từ chương trình câu lệnh sang chương trình hợp ngữ**

Cấu trúc một chương trình hợp ngữ được chuyển đổi gồm 3 đoạn chương trình:

1. Đoạn chương trình khởi động cho vi điều khiển, đoạn chương trình này là như nhau cho các chương trình hợp ngữ soạn thảo cho PLC.
2. Đoạn chương trình chuyển đổi tương ứng instruction program - Assembly program.
3. Đoạn chương trình con ngắt cho các timer khi có khai báo sử dụng.



**Hình 4.** Chương trình hợp ngữ được tạo ra từ chương trình câu lệnh và đặt trên form ASM program

**2.7. Xây dựng trình biên dịch để dịch từ chương trình hợp ngữ sang file bin.**

Chương trình biên dịch được thực hiện bằng 4 bước

**Bước 1:** Đọc từng dòng chương trình assembly, xử lý sơ bộ chuỗi nhằm xóa bỏ các dòng trống, xóa bỏ các đoạn chú thích, xóa bỏ các dấu cách, xóa bỏ các dấu Tab, chia một dòng lệnh assembly thành các bộ phận chứa vào một record có các trường là các bộ phận đó của một file. Ví dụ dòng lệnh sau:

LAB: CJNE @R0,#100,LAB1

List.label	List.part 1	List.part 2	List.part 3	List.part 4
LAB	CJNE	@R0	#100	LAB1

Dòng lệnh nào thiếu các bộ phận thì để trống.

Bước 1 được thực hiện cho đến hết chương trình assembly, tức là đến lệnh END

*Bước 2:* Đọc từng record của file temp.dat đã thực hiện ở bước 1, phân nhóm các trường List.part1, List.part2, List.part3, List.part4 bằng các private function Class1, Class2, Class3, Class4 .

t1 = Class1(list.part1)

t2 = Class2(list.part2)

t3 = Class3(list.part3)

t4 = Class4(list.part4)

*Bước 3:* Tra mảng MN(t1,t2,t3,t4) để tìm số thứ tự của mảng, số thứ tự này được coi là số thứ tự record và đọc số record này trong file lenh.dat (lenh.dat chứa các record mang thông tin mã lệnh, số byte của lệnh và loại lệnh phân loại theo quy ước, đã được tạo ra từ trước) để lấy được các thông tin mã lệnh, số byte, loại lệnh chứa trong các trường list.malenh, list.sobyte list.loailenh, sử dụng private sub CreateCode tạo ra 3 trường list.obj1, list.obj2 và list.obj3. chứa 3 trường này cùng với 4 trường List.part1, List.part2, List.part3, List.part4 vào một record của file FileCode.dat.

*Bước 4:* Đọc record 1 đến hết của FileCode.dat, ba trường list.obj1, list.obj2 và list.obj3 là các mã đối tượng ở dạng hexa, được chuyển thành dạng nhị phân và lưu vào file nhị phân, tức đã tạo thành file nhị phân.

## 2.8. Xây dựng phần mềm nạp file bin vào vi điều khiển của PLC

Sử dụng cổng nối tiếp của máy tính để giao tiếp với bộ logic lập trình PLC cho

quá trình nạp chương trình vào bộ nhớ chương trình của vi điều khiển. Visual Basic hỗ trợ cho việc truy cập lên cổng nối tiếp của máy tính thông qua đối tượng Microsoft comm.

Cổng truyền thông là COM1:

MSComm1.CommPort = 1

Baud rate: 600 b/s, n: không kiểm tra chẵn lẻ, 8:1 byte nối tiếp 8 bit, 1:1 bit stop:

MSComm1.Settings = "600,n,8,1"

Mở cổng truyền thông:

MSComm1.PortOpen = True

Truyền byte nhị phân ra cổng truyền thông:

MSComm1.Output = Chr\$(Bytengahphan)

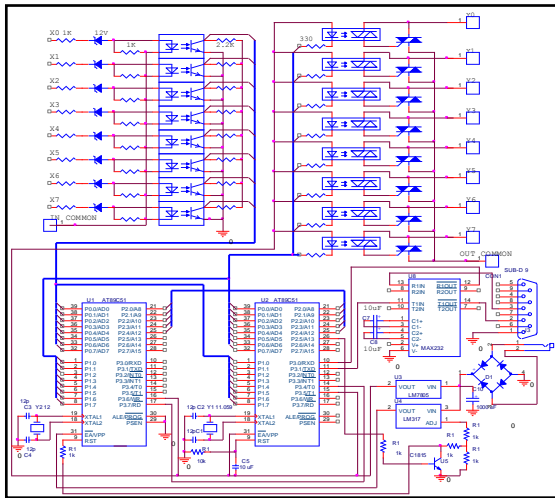
Sau quá trình nạp cho vi điều khiển là quá trình đọc ngược từ vi điều khiển vào máy tính để kiểm tra từng byte, nếu đúng hết sẽ phát thông báo nạp thành công, ngược lại phát thông báo không thành công. Sử dụng cổng truyền thông xong, phải đóng cổng truyền thông: MSComm1.PortOpen = False.

## 2.9. Xây dựng phần cứng bộ PLC

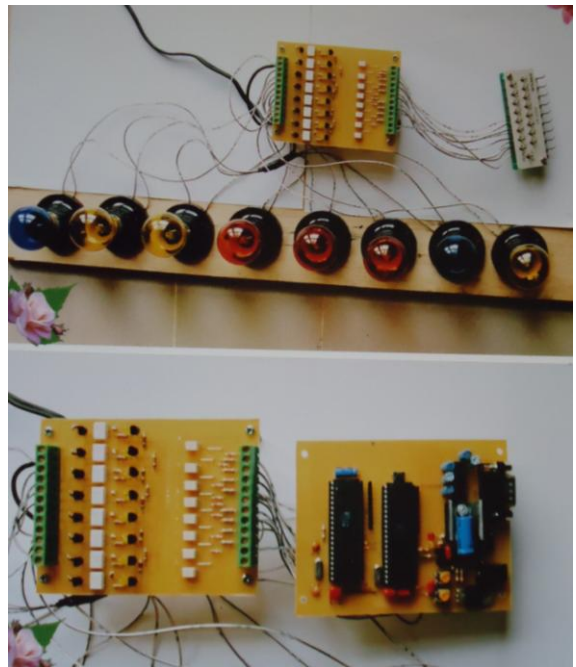
Sơ đồ phần cứng gồm 2 MCU AT89C51 U1 và U2, U1 có chức năng của PLC, U2 có chức năng của bộ nạp cho U1. Khi đang ở chế độ lập trình (nạp), P2.5 của U2 ở mức logic 0, transistor U5 ngưng, LM317 cấp Vpp = 12V cho U1, đồng thời cũng tác động mức reset = 5V cho U1. Còn đang ở chế độ hoạt động PLC, P2.5 của U2 ở mức logic 1, transistor U5 dẫn, LM317 cấp Vpp = 5V cho U1 và không tác động reset cho U1, để U1 hoạt động ở chế độ vi điều khiển. IC U8: MAX232 giao tiếp với máy tính và có nhiệm vụ chuyển mức RS232 - TTL, giao tiếp với máy tính ở đây được thiết kế là giao tiếp nối tiếp không bắt tay nên chỉ sử dụng 3 dây: Rx, D,

TxD, và dây đất. Mạch vào và mạch ra được cách ly với mạch trung tâm bằng Opto - Transistor và Opto - Triac, mạch vào có mức logic 1 = 24V và mức logic 0 = 0V, mức ngưỡng được thiết kế là 12V, nhờ

diode zener 12V. Việc sử dụng opto transistor ở mạch vào khiến mạch vào chỉ nhạy về dòng điện, tránh nhiễu điện áp do cảm ứng điện từ. Mạch ra được thiết kế opto – triac kích cho triac.



**Hình 5** (bên trái): Sơ đồ phân cứng PLC thiết kế.  
**Hình 6** (bên phải): Hình ảnh bo mạch PLC chế tạo và hình ảnh hoạt động mô phỏng thử nghiệm.



### 3. Các kết quả và kết luận

Đã tạo ra được một phần mềm tích hợp các công cụ sau:

- Công cụ soạn thảo chương trình ladder.
- Công cụ chuyển đổi chương trình ladder sang chương trình instruction.
- Công cụ chuyển đổi chương trình instruction sang chương trình hợp ngữ.
- Công cụ chuyển đổi chương trình hợp ngữ sang file nhị phân.
- Công cụ trợ giúp phần mềm.

- Phần mềm điều khiển quá trình nạp file nhị phân vào bộ điều khiển lập trình PLC.

- Chế tạo bộ PLC (phần cứng).

Bộ điều khiển lập trình PLC đã thiết kế và chế tạo cho thấy đã hoạt động đúng các chức năng căn bản của bộ điều khiển lập trình PLC. Từ kết quả chế tạo thử nghiệm, cho thấy bước đầu chúng ta hoàn toàn có khả năng chế tạo họ PLC đơn giản có tính năng tương đương với họ PLC của OMRON là CPM1 hoặc series FX0 của MITSUBISHI.

\*

## ABILITY TO MAKE SIMPLE PROGRAMMABLE LOGIC CONTROLLER (PLC)

Nguyen Van Son

Thu Dau Mot University

### ABSTRACT

*A programmable logic controller PLC is an important indispensable electronic instrument in an industrial automated production line. PLCs are manufactured under*

*industry standards for entry logic power (logic level "1" is 24V; logic level "0" is 0V). The outputs are on/off, or can be the relay, triac or transistor according to the standards for mechanical installation. Simple PLCs such as CPM1 of OMRON or FX0 series of MITSUBISHI have total number of inputs and outputs not exceeding 50 with basic scripts. The mentioned simple PLC, which was fabricated the hardware and software, meets the technical criteria of PLCs and is introduced in this article. PLCs have quite simple hardware. The important part of making PLCs is their programming software and recompiled tools. Currently, there is no local business producing PLCs. For imported PLCs are quite expensive, locally producing PLCs is economically significant and can enhance production levels of electronic instrument of the country.*

### **TÀI LIỆU THAM KHẢO**

- [1] MITSUBISHI, *FX Series Programming Manual*, February, 1999.
- [2] MITSUBISHI, *FX – PCS/Win – E Software Manual*, February, 1999.
- [3] MITSUBISHI, *FX0s Series Programmable Controllers – Hardware Manual*, Dec 1997.
- [4] MITSUBISHI, *Introductory Guide To Programmable Controllers*, February, 1999.
- [5] OMRON, *CPM1A Manual*, February 1999.
- [6] Các tài liệu tại trang web [www.atmel.com](http://www.atmel.com).
- [7] OMRON, *ES100P – Digital Controller – User’s Manual*, 1994.
- [8] Gary Dunning, *Introduction to Programmable Logic Controllers*, Thomson Learning, 1998, ISBN 0-8273-7866-1.
- [9] Ian G. Warnock, *Programmable Controllers – Operation and Application*, Prentice Hall, 1988.