

Chương IV

Các bài tập thực hiện trên Kit

Biên soạn : Lâm tăng Đức – Lê Tiến Dũng – Bộ môn TĐH

Chương IV

Các bài tập thực hiện trên Kit

Với mục đích là đào tạo nên tất cả chương trình DEMO của KIT đều được viết bằng hợp ngữ (Assembly), điều này giúp người học có thể hiểu rõ và nắm vững về cấu trúc phần cứng của họ vi điều khiển 8051. Ngoài ra còn làm quen được với cách lập trình không cấu trúc, khác với lập trình có cấu trúc bằng các ngôn ngữ bậc cao như C hay Basic, qua các bài thực hành. Cuối cùng người học có thể tự mình tối ưu được mã lệnh để tiết kiệm bộ nhớ, điều này là cần thiết khi dung lượng của ROM và RAM tích hợp sẵn trên chip của các vi điều khiển là không lớn.

4.1 Các ví dụ về điều khiển chip mở rộng I/O 8255 sử dụng trên KIT

Như đã tìm hiểu ở chương 2 về cấu trúc, các chế độ hoạt động của chip 8255, trong KIT này ta sử dụng 8255 làm chip mở rộng I/O hoạt động ở chế độ 0 – chế độ vào ra cơ bản. Khi 8255 làm việc ở chế độ này, các cổng A, B, CL, CU có thể được lập trình làm đầu vào hoặc đầu ra. Nên nhấn mạnh rằng, ở chế độ 0 thì tất cả các bit hoặc làm đầu vào, hoặc làm đầu ra mà không thể điều khiển riêng rẽ từng bit như các cổng P0-P3 của 8051.

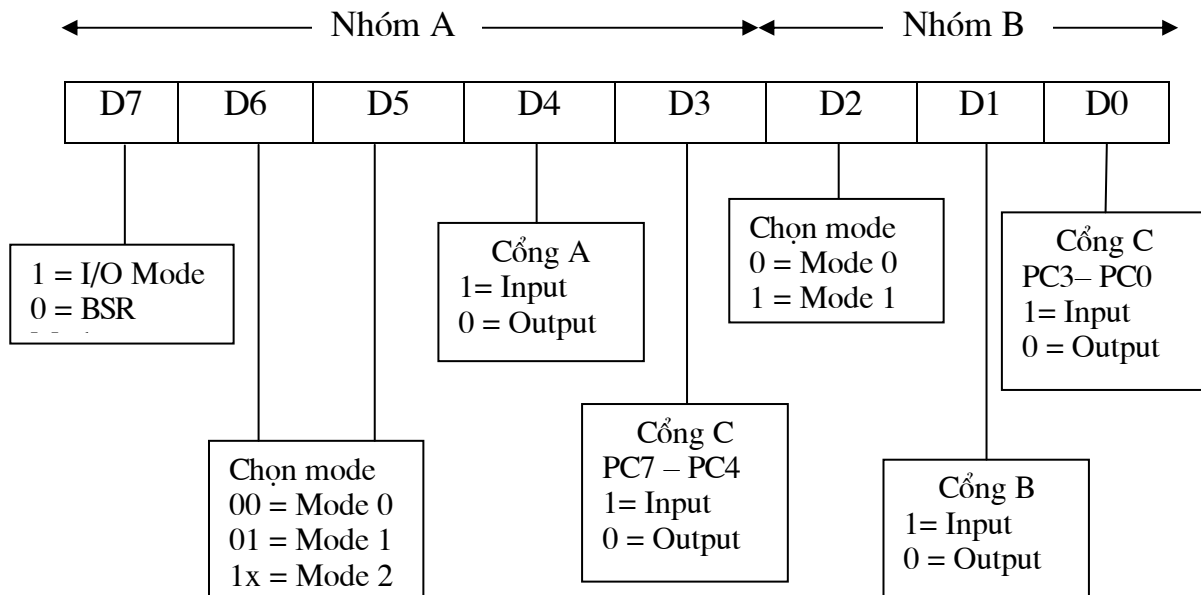
Cấu trúc từ điều khiển (CW) ở chế độ này có khuôn dạng như hình 2.15

4.1.1. Ví dụ 1: Xác định từ điều khiển của 8255 cho các cấu hình sau:

- a) Tất cả các cổng A,B và C đều là các cổng đầu ra (ở chế độ 0).
- b) PA làm đầu vào, PB - đầu ra, PCL - đầu vào và PCH - đầu ra

Giải: Từ hình 2.15 ta xác định được

- a) $10000000B = 80H$
- b) $10010001B = 91H$



Hình 2.15 Khuôn dạng từ điều khiển của 8255 (chế độ vào ra cơ sở)

4.1.2. Ví dụ 2: Giả sử như ta có một chip 8051 được nối với chip 8255 sao cho địa chỉ của các PORT và từ điều khiển của 8255 như sau:

8004H –PortA; 8005H – Port B; 8006H –PortC; 8007H –CW

- Tìm byte điều khiển để PA là cổng vào, PB, PC là cổng ra
- Viết một chương trình để lấy dữ liệu ở PA và gửi đến cổng B và C

Giải:

a) Dựa vào hình 2.15 ta có được từ điều khiển CW = 10010000B = 90H

b) Chương trình:

```
; #####
mov A, #90H ; PA là cổng vào, PB, PC là cổng ra
mov DPTR, #8007H ; CW
movx @DPTR, A ; Xuất từ điều khiển
```

Again:

```
mov  DPTR, #8004H    ; Địa chỉ PA
movx A, @DPTR        ; Nhận dữ liệu từ PA
inc  DPTR            ; Địa chỉ của cổng PB
movx @DPTR, A        ; Đưa dữ liệu ra cổng PB
inc  DPTR            ; Địa chỉ của cổng PC
movx @DPTR, A        ; Đưa dữ liệu ra PC

; #####
```

Sau khi đã hiểu rõ các thao tác làm việc với 8255 để giao tiếp với các thiết bị ngoại vi của KIT ta có tiến hành các bài thí nghiệm sau.

4.2 Bài toán 1 : Điều khiển 8 bit đầu ra số

Điều khiển 8 bit đầu ra số là một bài toán đơn giản nhằm giúp người học làm quen với việc điều khiển vi mạch 8255 để gửi ra cổng một tín hiệu số.

Bài toán thực hiện việc xuất ra một tín hiệu số 8 bit (1 byte) và kết quả này được hiển thị trên 8 đèn LED nhỏ tương ứng với 8 bit đã được bố trí trên bộ KIT, nếu 1 bit bằng 1 (High) thì đèn tương ứng sẽ tắt và ngược lại.

4.3 Bài toán 2 : Điều khiển nhận đầu vào 8 bit số và gửi đến 8 bit đầu ra

Bài toán thực hiện việc điều khiển vi mạch 8255 qui định một cổng của nó làm đầu vào nhận tín hiệu số và gửi tín hiệu đó đến một cổng ra khác.

4.4 Bài toán 3 : Điều khiển hiển thị tĩnh ma trận LED 8x8

Điều khiển hiển thị tĩnh ma trận LED là một bài toán điều khiển đơn giản ma trận LED hiển thị các điểm LED có màu sắc khác nhau theo từng cột xen kẽ nhau, bài toán này giúp người học hiểu rõ hơn cấu tạo của ma trận LED và làm quen với việc điều khiển nó.

4.5 Bài toán 4 : Điều khiển hiển thị động ma trận LED 8x8

Điều khiển hiển thị động ma trận LED là một bài toán phức tạp hơn bài toán hiển thị tĩnh ở trên. Nội dung của bài toán là thực hiện việc quét liên lục theo từng dòng của ma trận LED để hiển thị các chữ số trong hệ Hexa từ 0-9.. đến A-F lên trên bảng LED.

4.6 Bài toán 5 : Điều khiển ADC0809 và DAC0808

Với ADC0809: bài toán thực hiện việc chuyển đổi tín hiệu Analog ở một trong 8 kênh đầu vào thành tín hiệu số 8 bit ở đầu ra của ADC0809. Sau đó đọc tín hiệu số đã chuyển đổi này hiển thị lên LCD. Ở phần này trên KIT còn trang bị sẵn một cảm biến nhiệt độ, bài toán thí nghiệm để đo nhiệt độ được thực hiện khi ta bật nút thứ 4 của DIP SWITCH để nối tín hiệu đầu ra của LM35 đến kênh đầu vào 4 của ADC0809, nhiệt độ phòng cũng được hiển thị trên LCD.

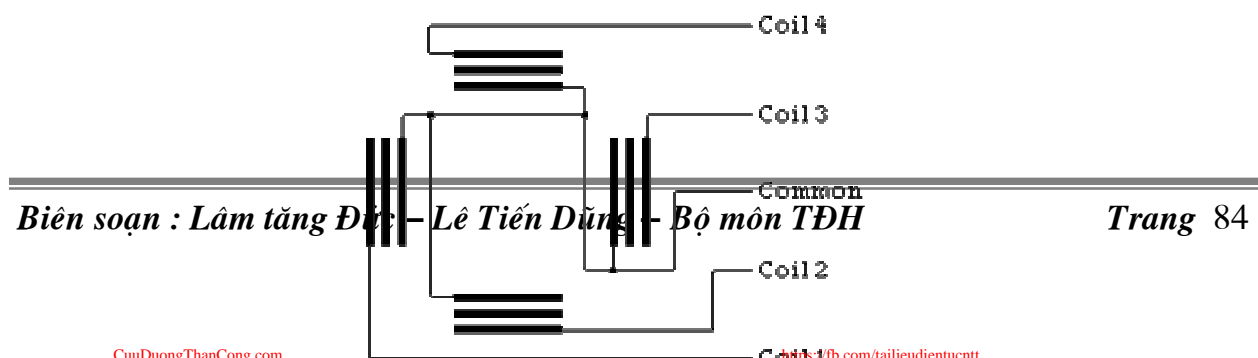
Với DAC0808: bài toán thực hiện chuyển đổi tín hiệu số 8 bit thành tín hiệu tương tự có dải 0 -10V ở đầu ra của DAC0808, tín hiệu này có thể đưa vào máy hiện sóng (Oscilloscope) để xem kết quả đối với chương trình tạo sóng (sin) hoặc có thể đo bằng đồng hồ bình thường nếu chỉ tạo ra ở đầu ra 1 giá trị điện áp không đổi.

4.7 Bài toán 6 : Điều khiển động cơ bước và động cơ 1 chiều

Bài toán thực hiện việc điều khiển động cơ bước quay, thay đổi tốc độ, đảo chiều, dừng động cơ. Chương trình sử dụng 4 đầu vào xung để làm các phím làm thay đổi các trạng thái trên của động cơ bước.

* Thường các cuộn dây của động cơ bước được xác định theo màu của các dây, tuy nhiên đối với một động cơ bước bất kỳ ta có thể dùng đồng hồ để xác định các cuộn dây như hình vẽ, ở đây trình bày cách xác định cho các động cơ có 5, 6 dây

1. **Dùng đồng hồ để xác định đầu Common:** dùng đồng hồ để ở thang đo trở, đo trở giữa các cặp dây, đầu Common là đầu có trở giữa nó các đầu khác bằng 1/2 điện trở của các đầu khác với nhau.



2. **Xác định thứ tự các cuộn dây** bằng cách cấp nguồn vào đầu Common và nối đất 1 trong 4 dây còn lại, đồng thời nối đất 1 trong 3 dây còn lại xem phản ứng (chiều quay) của động cơ và đánh dấu thứ tự cho nó như trong bảng sau:

Nối mass một đầu, đánh dấu nó là đầu 4	
Giữ nguyên nó, nối mass tiếp lần lượt một trong 3 dây còn lại	
Nếu động cơ quay theo chiều kim đồng hồ, đó là cuộn 3	
Nếu động cơ quay ngược chiều kim đồng hồ, đó là cuộn 1	
Dây không có phản ứng gì cả là cuộn 2	

Khi biết được thứ tự các cuộn dây, ta kích xung theo thứ tự đó động cơ sẽ chạy

Ví dụ một đoạn chương trình như sau, giả sử 4 đầu của động cơ được 4 bit P1.0 – P1.3 của 8051 điều khiển:

```
; #####
```

```
ORG 0000H
```

```
MOV R3, #00000011B
```

```
MOV A, R3
BACK: MOV P1,A
      RL A           ; Quay thanh ghi A
      ACALL DELAY
      SJMP BACK
; Tốc độ quay có thể thay đổi bằng thay đổi thời gian Delay
DELAY:
      MOV R1,#50
H1:   MOV R2 ,#255
H2:   DJNZ R2,H2
      DJNZ R1,H1
      RET
      END
; #####
```

4.8 Bài toán 7 : Giao tiếp với máy tính bằng giao thức RS232 sử dụng cổng nối tiếp. Chương trình DEMO chạy trên PC được viết bằng Delphi 7, có sử dụng thư viện CPort 3.0 hỗ trợ giao tiếp.

Các hướng dẫn sử dụng có thể xem trực tiếp phần trợ giúp của chương trình.

cuu duong than cong . com