

BỘ NÔNG NGHIỆP VÀ PHÁT TRIỂN NÔNG THÔN
TRƯỜNG CAO ĐẲNG CƠ GIỚI NINH BÌNH

- * - * - * -

GIÁO TRÌNH
MÔ ĐUN: KỸ THUẬT VI ĐIỀU KHIỂN
NGHỀ ĐIỆN DÂN DỤNG
Trình độ: CAO ĐẲNG NGHỀ

*(Ban hành kèm theo Quyết định số: /QĐ-CDNCGNB-ĐT, ngày tháng năm
của Hiệu trưởng trường Cao đẳng Cơ giới Ninh Bình)*

Ninh Bình, năm 2021

TUYÊN BỐ BẢN QUYỀN

Tài liệu này thuộc loại sách giáo trình nên các nguồn thông tin có thể được phép dùng nguyên bản hoặc trích dùng cho các mục đích về đào tạo và tham khảo.

Mọi mục đích khác mang tính lệch lạc hoặc sử dụng với mục đích kinh doanh thiếu lành mạnh sẽ bị nghiêm cấm.

Tailieu.vn

LỜI GIỚI THIỆU

Giáo trình Kỹ thuật vi điều khiển được biên soạn trên cơ sở chương trình giảng dạy theo chương trình khung của trường Cao Đẳng Cơ Giới Ninh Bình.

Khi biên soạn, nhóm biên soạn đã cố gắng cập nhật những kiến thức mới có liên quan đến nội dung chương trình đào tạo và phù hợp với mục tiêu đào tạo, nội dung lý thuyết và thực hành được biên soạn gắn với nhu cầu thực tế trong sản xuất đồng thời có tính thực tiễn cao.

Nội dung giáo trình được biên soạn với dung lượng thời gian đào tạo 60 giờ gồm có:

Bài 1: Tổng quan về vi xử lý - vi điều khiển.

Bài 2: Họ vi điều khiển 8051.

Bài 3: Ngôn ngữ lập trình

Bài 4: Hoạt động ngắt và hoạt động định thời

Bài 5: Lập trình ứng dụng vi điều khiển 8051

Trong quá trình sử dụng giáo trình, tùy theo yêu cầu cũng như khoa học và công nghệ phát triển có thể điều chỉnh thời gian và bổ sung những kiến thức mới cho phù hợp. Trong giáo trình, chúng tôi có đề ra nội dung thực tập của từng bài để người học cũng cố và áp dụng kiến thức phù hợp với kỹ năng. Tuy nhiên, tùy theo điều kiện cơ sở vật chất và trang thiết bị, các trường có thể sử dụng cho phù hợp.

Mặc dù đã cố gắng tổ chức biên soạn để đáp ứng được mục tiêu đào tạo nhưng không tránh được những khiếm khuyết. Rất mong nhận được đóng góp ý kiến của người sử dụng, người đọc để nhóm biên soạn sẽ hiện chỉnh hoàn thiện hơn sau thời gian sử dụng

Ninh Bình, ngày tháng năm

Nhóm tác giả

Tổng Thanh Bình

Mục Lục

TUYÊN BỐ BẢN QUYỀN.....	1
LỜI GIỚI THIỆU	2
Mục Lục.....	3
BÀI 1: TỔNG QUAN VỀ VI XỬ LÝ – VI ĐIỀU KHIỂN	8
1.1.Giới thiệu chung về Vi xử lý – Vi điều khiển.....	9
1.1.1 Một số khái niệm.....	9
1.1.2 Lịch sử phát triển của các bộ Vi xử lý – Vi điều khiển	11
1.1.3 Ứng dụng của Vi xử lý – vi điều khiển.....	12
1.2.Cấu trúc chung của hệ vi xử lý – vi điều khiển	13
1.2.1. Khối xử lý trung tâm (CPU).....	14
1.2.3. Cổng vào/ra song song.....	18
1.2.4. Cổng vào/ra nối tiếp.....	19
1.2.5. Bộ đếm/Bộ định thời.....	19
1.2.6. Hệ thống bus	20
1.3.Định dạng dữ liệu và biểu diễn thông tin trong hệ vi xử lý – vi điều khiển	20
1.4. Các hệ đếm.....	20
BÀI 2: HỌ VI ĐIỀU KHIỂN 8051	25
2.1. Giới thiệu chung.....	26
2.1.1. Những thành cơ bản của vi điều khiển.....	27
2.1.2. Hoạt động của vi điều khiển.....	27
2.2. Kiến trúc vi điều khiển 8051.....	33
2.2.1. Chuẩn 8051	33
2.2.2. Data sheet vi điều khiển 8051	35
2.2.4. Bộ đếm và bộ định thời.....	45
2.2.5. Truyền thông không đồng bộ (UART).....	46
2.2.6. Ngắt vi điều khiển 8051	46
BÀI 3: NGÔN NGỮ LẬP TRÌNH.....	47
3.1. Biến (variables).....	48
3.2. Hằng (Constant).....	48
3.3. Mảng (Array)	48
3.4. Các toán tử	48
3.5. Các kiểu dữ liệu.....	50
3.6. Cấu trúc của một chương trình.....	51

3.7.	Một số lệnh cơ bản	52
3.8.	Hướng dẫn sử dụng phần mềm KeilC	57
BÀI 4: HOẠT ĐỘNG NGẮT VÀ HOẠT ĐỘNG ĐỊNH THỜI		65
4.1.	Các bộ định thời (Timer).....	66
4.1.1.	Timer là gì?	66
4.1.2.	Các thanh ghi timer trong 8051.....	66
4.1.3.	Các chế độ hoạt động của timer	69
4.1.4.	Cách tính giá trị để nạp vào TH và TL.....	72
4.1.5.	Lập trình cho timer	72
4.2.	Các Ngắt (INTERRUPT)	74
4.2.1.	Ngắt là gì?	74
4.2.2.	Các ngắt của 8051.....	75
4.2.3.	Các thanh ghi của “ NGẮT”	75
BÀI 5: LẬP TRÌNH ỨNG DỤNG VI ĐIỀU KHIỂN 8051		81
5.1.	Giao tiếp các cổng vào/ra (I/O) với LED.....	82
5.2.	Giao tiếp vào/ra (I/O) với LED 7 thanh.....	86
5.3.	Giao tiếp vào/ra với LED Matrix.....	94
5.4.	Giao tiếp vào/ra (I/O) với keypad.....	100
5.5.	Giao tiếp vào/ra (I/O) với LCD.....	108
TÀI LIỆU THAM KHẢO.....		115

CHƯƠNG TRÌNH MÔ ĐUN

Tên mô đun : Kỹ thuật vi điều khiển

Mã mô đun: MĐ 26

Thời gian thực hiện mô đun: 60 giờ; (Lý thuyết: 17 giờ; Thực hành, thí nghiệm, thảo luận, bài tập: 40 giờ; Kiểm tra: 03 giờ.)

I. Vị trí tính chất của mô đun:

- Vị trí: Mô đun được bố trí dạy sau khi học xong môn học mô đun: Kỹ thuật xung số; Kỹ thuật điện tử; PLC cơ bản.

- Tính chất: Là mô đun chuyên môn nghề.

II. Mục tiêu mô đun:

Sau khi học xong mô đun, người học có khả năng:

- Về kiến thức:

+ Hiểu được các kiến thức cơ bản về Họ vi điều khiển 8051.

+ Giải thích được nguyên lý hoạt động của hệ dùng vi điều khiển.

- Về kỹ năng:

+ Kiểm tra và viết được các chương trình ứng dụng dùng vi điều khiển.

+ Vận hành được các thiết bị và dây chuyền sản xuất dùng vi điều khiển.

- Về năng lực tự chủ và trách nhiệm: Phát huy tính tích cực, chủ động, sáng tạo và tư duy khoa học trong công việc

III. Nội dung mô đun:

1. Nội dung tổng quát và phân bổ thời gian:

Số TT	Tên các bài trong mô đun	Thời gian			
		Tổng số	Lý thuyết	Thực hành, thí nghiệm, thảo luận, Bài tập	Kiểm tra*
1	Bài 1: Tổng quan về Vi xử lý – Vi điều khiển	2	2		
	1. Giới thiệu chung về Vi xử lý – Vi điều khiển		0,5		
	2. Cấu trúc chung của hệ vi xử lý – vi điều khiển		1		
	3. Định dạng dữ liệu và biểu diễn thông tin trong hệ Vi xử lý – Vi điều khiển		0,5		
2	Bài 2: Họ vi điều khiển 8051	2	2		
	1. Giới thiệu chung		1		
	2. Kiến trúc vi điều khiển 8051		1		
3	Bài 3: Ngôn ngữ lập trình	8	4	2	1
	1. Biến (variables)		0,25		
	2. Hằng (Constant)		0,25		
	3. Mảng (Array)		0,25		
	4. Các toán tử		0,25		
	5. Các kiểu dữ liệu		0,5		
	6. Cấu trúc của một chương trình		0,5		
	7. Một số lệnh cơ bản		1	1	

Số TT	Tên các bài trong mô đun	Thời gian			
		Tổng số	Lý thuyết	Thực hành, thí nghiệm, thảo luận, Bài tập	Kiểm tra*
	8. Hướng dẫn sử dụng phần mềm KeilC		1	1	
	9. Hướng dẫn sử dụng phần mềm proteus		1	3	
4	Bài 4: Hoạt động ngắt và hoạt động định thời	4	3	1	
	1. Các bộ định thời (Timer)		2	0,5	
	2. Các Ngắt (INTERRUPT)		1	0,5	
5	Bài 5: Lập trình ứng dụng vi điều khiển 8051	44	8	34	2
	1. Giao tiếp các cổng vào/ra (I/O) với LED		2	6	
	2. Giao tiếp vào/ra (I/O) với LED 7 thanh		2	6	
	3. Giao tiếp vào/ra với LED Matrix		2	6	
	4. Giao tiếp vào/ra (I/O) với Keypad		1	7	
	5. Giao tiếp vào/ra (I/O) với LCD		1	9	
	Cộng	60	17	40	3

BÀI 1: TỔNG QUAN VỀ VI XỬ LÝ – VI ĐIỀU KHIỂN

1. Mục tiêu của bài:

- Hiểu lịch sử phát triển của vi điều khiển;
- Hiểu được cấu trúc chung của vi điều khiển;
- Biết được các lĩnh vực ứng dụng và hướng phát triển trong tương lai của vi điều khiển.

2. Nội dung bài:

2.1. Giới thiệu chung về Vi xử lý – Vi điều khiển

2.1.1 Một số khái niệm

2.1.2 Lịch sử phát triển của các bộ Vi xử lý – Vi điều khiển

2.1.3 Ứng dụng của Vi xử lý – vi điều khiển

2.2. Cấu trúc chung của hệ vi xử lý – vi điều khiển

2.2.1. Khối xử lý trung tâm (CPU)

2.2.2. Bộ nhớ

2.2.3. Cổng vào/ra song song

2.2.4. Cổng vào/ra nối tiếp

2.2.5. Bộ đếm/Bộ định thời

2.2.6. Hệ thống bus

2.3. Định dạng dữ liệu và biểu diễn thông tin trong hệ Vi xử lý – Vi điều khiển

2.3.1 Các hệ đếm

2.3.2 Mã ký tự - Alphanumeric CODE (ASCII, EBCDIC)

2.3.3 Các phép toán số học trên hệ đếm nhị phân

1.1. Giới thiệu chung về Vi xử lý – Vi điều khiển

1.1.1 Một số khái niệm

Khái niệm “vi xử lý” (microprocessor) và “vi điều khiển” (microcontroller). Về cơ bản hai khái niệm này không khác nhau nhiều, “vi xử lý” là thuật ngữ chung dùng để đề cập đến kỹ thuật ứng dụng các công nghệ vi điện tử, công nghệ tích hợp và khả năng xử lý theo chương trình vào các lĩnh vực khác nhau. Vào những giai đoạn đầu trong quá trình phát triển của công nghệ vi xử lý, các chip (hay các vi xử lý) được chế tạo chỉ tích hợp những phần cứng thiết yếu như CPU cùng các mạch giao tiếp giữa CPU và các phần cứng khác. Trong giai đoạn này, các phần cứng (kể cả bộ nhớ) thường không được tích hợp trên chip mà phải ghép nối thêm bên ngoài. Các phần cứng này được gọi là các ngoại vi (Peripherals).

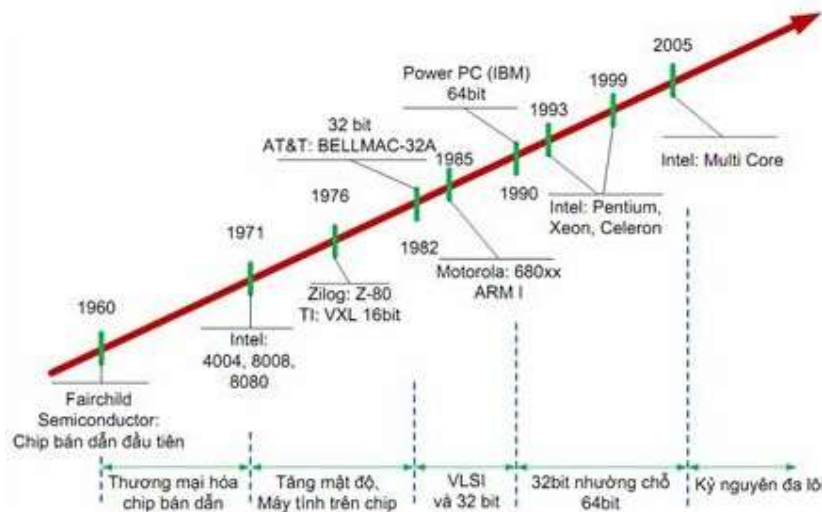
Về sau, nhờ sự phát triển vượt bậc của công nghệ tích hợp, các ngoại vi cũng được tích hợp vào bên trong IC và người ta gọi các vi xử lý đã được tích hợp thêm các ngoại vi là các “vi điều khiển”. Vi xử lý có các khối chức năng cần thiết để lấy dữ liệu, xử lý dữ liệu và xuất dữ liệu ra ngoài sau khi đã xử lý. Và chức năng chính của Vi xử lý chính là xử lý dữ liệu, chẳng hạn như cộng, trừ, nhân, chia, so sánh.v.v... Vi xử lý không có khả năng giao tiếp trực tiếp với các thiết bị ngoại vi, nó chỉ có khả năng nhận và xử lý dữ liệu. Để vi xử lý hoạt động cần có chương trình kèm theo, các chương trình này điều khiển các mạch logic và từ đó vi xử lý xử lý các dữ liệu cần thiết theo yêu cầu.

Chương trình là tập hợp các lệnh để xử lý dữ liệu thực hiện từng lệnh được lưu trữ trong bộ nhớ, công việc thực hành lệnh bao gồm: nhận lệnh từ bộ nhớ, giải mã lệnh và thực hiện lệnh sau khi đã giải mã. Để thực hiện các công việc với các thiết bị cuối cùng, chẳng hạn điều khiển động cơ, hiển thị kí tự trên màn hình

đòi hỏi phải kết hợp vi xử lý với các mạch điện giao tiếp với bên ngoài được gọi là các thiết bị I/O (nhập/xuất) hay còn gọi là các thiết bị ngoại vi. Bản thân các vi xử lý khi đứng một mình không có nhiều hiệu quả sử dụng, nhưng khi là một phần của một bộ vi điều khiển, thì hiệu quả ứng dụng của Vi xử lý là rất lớn. Vi xử lý kết hợp với các thiết bị khác được sử dụng trong các hệ thống lớn, phức tạp đòi hỏi phải xử lý một lượng lớn các phép tính phức tạp, có tốc độ nhanh. Chẳng hạn như các hệ thống sản xuất tự động trong công nghiệp, các tổng đài điện thoại, hoặc ở các robot có khả năng hoạt động phức tạp v.v... Bộ Vi xử lý có khả năng vượt bậc so với các hệ thống khác về khả năng tính toán, xử lý, và thay đổi chương trình linh hoạt theo mục đích người dùng, đặc biệt hiệu quả đối với các bài toán và hệ thống lớn. Tuy nhiên đối với các ứng dụng nhỏ, tầm tính toán không đòi hỏi khả năng tính toán lớn thì việc ứng dụng vi xử lý cần cân nhắc. Bởi vì hệ thống dù lớn hay nhỏ, nếu dùng vi xử lý thì cũng đòi hỏi

các khối mạch điện giao tiếp phức tạp như nhau. Các khối này bao gồm bộ nhớ để chứa dữ liệu và chương trình thực hiện, các mạch điện giao tiếp ngoại vi để xuất nhập và điều khiển trở lại, các khối này cùng liên kết với vi xử lý thì mới thực hiện được công việc. Để kết nối các khối này đòi hỏi người thiết kế phải hiểu biết tinh tường về các thành phần vi xử lý, bộ nhớ, các thiết bị ngoại vi. Hệ thống được tạo ra khá phức tạp, chiếm nhiều không gian, mạch in phức tạp và vấn đề chính là trình độ người thiết kế. Kết quả là giá thành sản phẩm cuối cùng rất cao, không phù hợp để áp dụng cho các hệ thống nhỏ. Vì một số nhược điểm trên nên các nhà chế tạo tích hợp một ít bộ nhớ và một số mạch giao tiếp ngoại vi cùng với vi xử lý vào một IC duy nhất được gọi là Microcontroller-Vi điều khiển, vi điều khiển có khả năng tương tự như khả năng của vi xử lý, nhưng cấu trúc phần cứng đơn giản hơn nhiều. Vi điều khiển ra đời mang lại sự tiện lợi đối với người dùng, họ không cần nắm vững một khối lượng kiến thức quá lớn như khi làm việc với vi xử lý, việc thiết kế mạch điện cũng trở nên đơn giản hơn nhiều và có khả năng giao tiếp trực tiếp với các thiết bị bên ngoài. Vi điều khiển tuy được xây dựng với phần cứng đơn giản hơn, nhưng thay vào ưu điểm này là khả năng xử lý bị giới hạn (tốc độ xử lý chậm hơn và khả năng tính toán ít hơn, dung lượng chương trình bị giới hạn). Vì vậy, vi điều khiển có giá thành rẻ hơn nhiều so với vi xử lý, việc sử dụng đơn giản, nên nó được ứng dụng vào nhiều, không đòi hỏi tính toán phức tạp. Vi điều khiển được ứng dụng trong các dây chuyền tự động loại nhỏ, các robot có chức năng đơn giản, trong máy giặt, ô tô v.v... Năm 1976 Intel giới thiệu bộ vi điều khiển (microcontroller) 8748, một chip tương tự như các bộ vi xử lý và là chip đầu tiên trong họ MCS-48. Độ phức tạp, kích thước và khả năng của Vi điều khiển tăng thêm một bậc quan trọng vào năm 1980 khi intel tung ra chip 8051, bộ Vi điều khiển đầu tiên của họ MCS-51 và là chuẩn công nghệ cho nhiều họ Vi điều khiển được sản xuất sau này. Sau đó rất nhiều họ Vi điều khiển của nhiều nhà chế tạo khác nhau lần lượt được đưa ra thị trường với tính năng được cải tiến ngày càng mạnh. Trong tài liệu này, ranh giới giữa hai khái niệm “vi xử lý” và “vi điều khiển” thực sự không cần phải phân biệt rõ ràng. Ta dùng thuật ngữ “vi xử lý” khi đề cập đến các khái niệm cơ bản của kỹ thuật vi xử lý nói chung và sẽ dùng thuật ngữ “vi điều khiển” khi đi sâu nghiên cứu một họ chip cụ thể.

1.1.2 Lịch sử phát triển của các bộ Vi xử lý – Vi điều khiển



Hình 1.1. Lịch sử phát triển của Vi xử lý – Vi điều khiển

- Thế hệ 1 (1971 - 1973): vi xử lý 4 bit, đại diện là 4004, 4040, 8080 (Intel) hay IPM-16 (National Semiconductor).
- + Độ dài word thường là 4 bit (có thể lớn hơn).
- + Tốc độ 10 - 60 μ s / lệnh với tần số xung nhịp 0.1 - 0.8 MHz. + Tập lệnh đơn giản và phải cần nhiều vi mạch phụ trợ.
- Thế hệ 2 (1974 - 1977): vi xử lý 8 bit, đại diện là 8080, 8085 (Intel) hay Z80
- + Tập lệnh phong phú hơn.
- + Địa chỉ có thể đến 64 KB. Một số bộ vi xử lý có thể phân biệt 256 địa chỉ cho thiết bị ngoại vi.
- + Sử dụng công nghệ NMOS hay CMOS.
- + Tốc độ 1 - 8 μ s / lệnh với tần số xung nhịp 1 - 5 MHz
- Thế hệ 3 (1978 - 1982): vi xử lý 16 bit, đại diện là 68000/68010 (Motorola) hay 8086/ 80286/ 80386 (Intel)
- + Tập lệnh đa dạng với các lệnh nhân, chia và xử lý chuỗi.
- + Địa chỉ bộ nhớ có thể từ 1 - 16 MB và có thể phân biệt tới 64KB địa chỉ cho ngoại vi
- + Sử dụng công nghệ HMOS.
- + Tốc độ 0.1 - 1 μ s / lệnh với tần số xung nhịp 5 - 10 MHz.
- Thế hệ 4: vi xử lý 32 bit 68020/68030/68040/68060 (Motorola) hay 80386/80486 (Intel) và vi xử lý 32 bit Pentium (Intel)
- + Bus địa chỉ 32 bit, phân biệt 4 GB bộ nhớ. + Có thể dùng thêm các bộ đồng xử lý (coprocessor). + Có khả năng làm việc với bộ nhớ ảo.

- + Có các cơ chế pipeline, bộ nhớ cache.
- + Sử dụng công nghệ HCMOS.
- Thế hệ 5: vi xử lý 64

1.1.3 Ứng dụng của Vi xử lý – vi điều khiển

Vi xử lý, chính là chip của các loại máy tính ngày nay. Ở đây, ta chỉ nói đến ứng dụng của vi điều khiển. Vi điều khiển có thể dùng trong thiết kế các loại hệ thống nhúng. Hệ thống nhúng có trong hầu hết các thiết bị tự động, thông minh ngày nay. Chúng ta có thể dùng vi điều khiển để thiết kế bộ điều khiển cho các sản phẩm như:

Trong các sản phẩm dân dụng:

- Nhà thông minh:
- Cửa tự động
- Khóa số
- Tự động điều tiết ánh sáng thông minh (bật/tắt đèn theo thời gian, theo cường độ ánh sáng,...)
- Điều khiển các thiết bị từ xa (qua điều khiển, qua tiếng vỗ tay,...)
- Điều tiết hơi ẩm, điều tiết nhiệt độ, điều tiết không khí, gió
- Hệ thống vệ sinh thông minh,...

Trong quảng cáo:

- Các loại biển quảng cáo nháy chữ
- Quảng cáo ma trận LED (một màu, 3 màu, đa màu)
- Điều khiển máy cuốn bạt quảng cáo,...

Các máy móc dân dụng:

- Máy điều tiết độ ẩm cho vườn cây
- Buồng ấp trứng gà/vịt
- Đồng hồ số, đồng hồ số có điều khiển theo thời gian

Các sản phẩm giải trí:

- Máy nghe nhạc
- Máy chơi game
- Đầu thu kỹ thuật số, đầu thu set-top-box,...

Trong các thiết bị y tế:

- Máy móc thiết bị hỗ trợ: máy đo nhịp tim, máy đo đường huyết, máy đo huyết áp, điện tim đồ, điện não đồ,...
- Máy cắt/mài kính

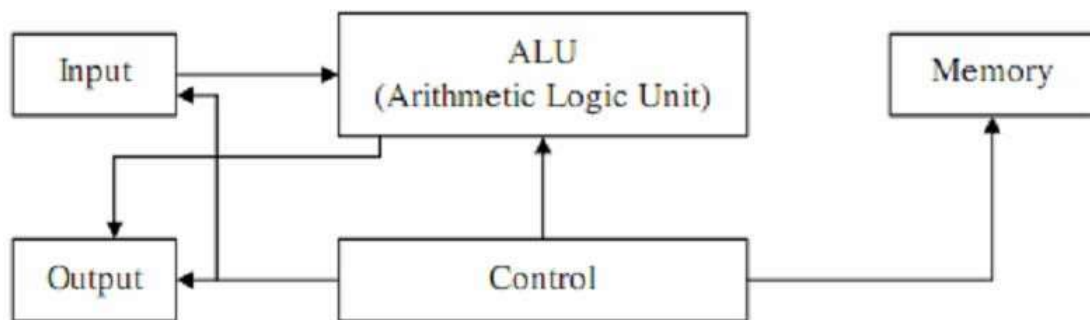
- Máy chụp chiếu (city, X-quang,...)

Các sản phẩm công nghiệp:

- Điều khiển động cơ
- Điều khiển số (PID, mờ,...)
- Đo lường (đo điện áp, đo dòng điện, áp suất, nhiệt độ,...)
- Cân băng tải, cân toa xe, cân ô tô,...
- Máy cán thép: điều khiển động cơ máy cán, điều khiển máy quấn thép,..
- Làm bộ điều khiển trung tâm cho RoBot
- Ổn định tốc độ động cơ
- Đếm sản phẩm của 1 nhà máy, xí nghiệp,...
- Máy vận hành tự động (dạng CNC)

1.2.Cấu trúc chung của hệ vi xử lý – vi điều khiển

Sơ đồ khối một máy tính cổ điển



Hình 1.2. Sơ đồ khối một máy tính cổ điển

- ALU (đơn vị logic số học): thực hiện các bài toán cho máy tính bao gồm: cộng (+), trừ (-), nhân (*), chia (/), phép toán logic, ...
- Control (điều khiển): điều khiển, kiểm soát các đường dữ liệu giữa các thành phần của máy tính.
- Memory (bộ nhớ): lưu trữ chương trình hay các kết quả trung gian.
- Input (nhập), Output (Xuất): xuất nhập dữ liệu (còn gọi là thiết bị ngoại vi).

Về cơ bản kiến trúc của một vi điều khiển gồm những phần cứng sau:

- Đơn vị xử lý trung tâm CPU (Central Processing Unit).
- Các bộ nhớ (Memories).
- Các cổng vào/ra (song song (Parallel I/O Ports), nối tiếp (Serial I/O Ports))
- Các bộ đếm/bộ định thời (Timers).
- Hệ thống BUS (Địa chỉ, dữ liệu, điều khiển)

Ngoài ra với mỗi loại vi điều khiển cụ thể còn có thể có thêm một số

phần cứng khác như bộ biến đổi tương tự - số ADC, bộ biến đổi số - tương tự DAC, các mạch điều chế dạng sóng WG, điều chế độ rộng xung PWM...Bộ não của mỗi vi điều khiển chính là CPU, các phần cứng khác chỉ là các cơ quan chấp hành dưới quyền của CPU. Mỗi cơ quan này đều có một cơ chế hoạt động nhất định mà CPU phải tuân theo khi giao tiếp với chúng.

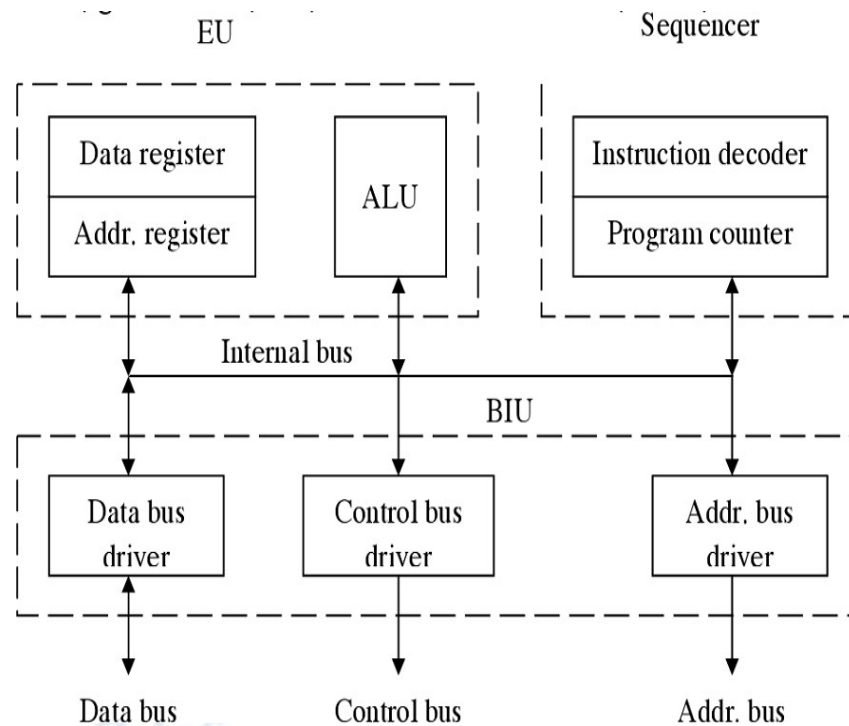


Hình 1.3. Sơ đồ khối hệ vi điều khiển

Để có thể giao tiếp và điều khiển các cơ quan chấp hành (các ngoại vi), CPU sử dụng 03 loại tín hiệu cơ bản là tín hiệu địa chỉ (Address), tín hiệu dữ liệu (Data) và tín hiệu điều khiển (Control). Về mặt vật lý thì các tín hiệu này là các đường dây dẫn điện nối từ CPU đến các ngoại vi hoặc thậm chí là giữa các ngoại vi với nhau. Tập hợp các đường tín hiệu có cùng chức năng gọi là các bus. Như vậy ta có các bus địa chỉ, bus dữ liệu và bus điều khiển.

1.2.1. Khối xử lý trung tâm (CPU)

CPU có cấu tạo gồm có đơn vị xử lý số học và logic (ALU), các thanh ghi, các khối logic và các mạch giao tiếp. Chức năng của CPU là tiến hành các thao tác tính toán xử lý, đưa ra các tín hiệu địa chỉ, dữ liệu và điều khiển nhằm thực hiện một nhiệm vụ nào đó do người lập trình đưa ra thông qua các lệnh (Instructions).

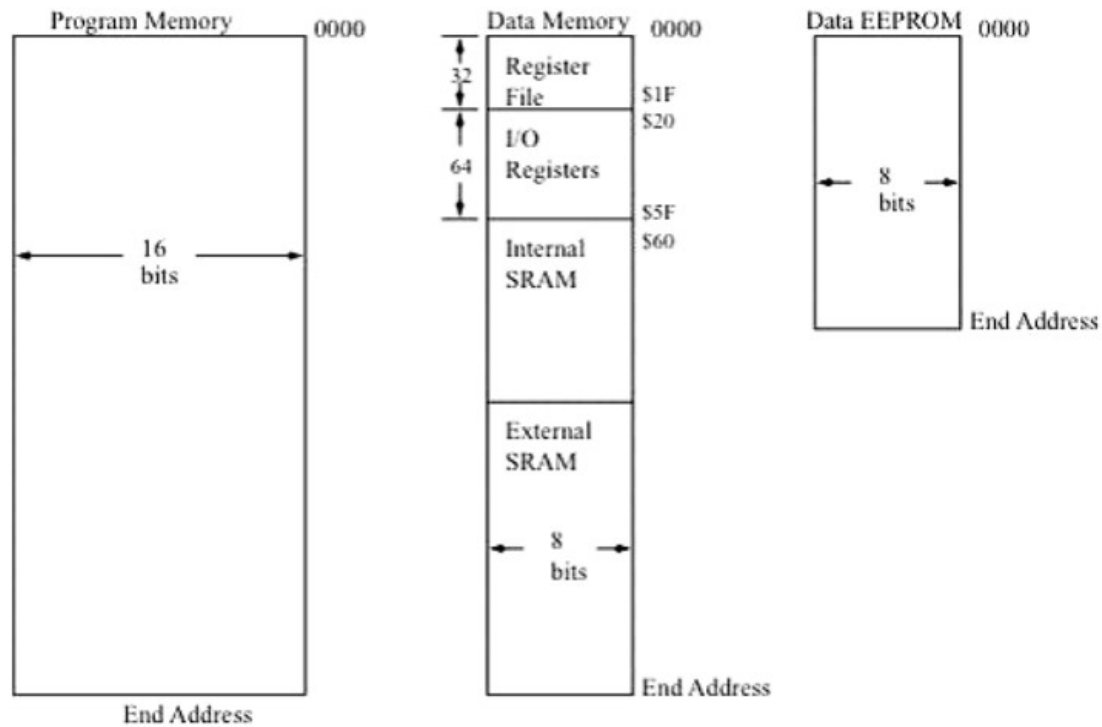


Hình 1.4. Khối xử lý trung tâm

1.2.2. Bộ nhớ

Nhìn chung có hai loại bộ nhớ là bộ nhớ chương trình và bộ nhớ dữ liệu. Bộ nhớ chương trình dùng để chứa mã chương trình hướng dẫn CPU thực hiện một nhiệm vụ nào đó. Thông thường thì bộ nhớ chương trình là các loại bộ nhớ “không bay hơi” (non-volatile), nghĩa là không bị mất nội dung chứa bên trong khi ngừng cung cấp nguồn nuôi. Có thể kể ra một số bộ nhớ thuộc loại này như: ROM, PROM, EPROM, EEPROM, Flash. Bộ nhớ dữ liệu là bộ nhớ dùng để chứa dữ liệu (bao gồm các tham số, các biến tạm thời...). Tùy thuộc loại dữ liệu mà bộ nhớ loại này có thể là loại “không bay hơi” hoặc “bay hơi” (mất dữ liệu khi cắt nguồn nuôi). Loại “bay hơi” thường thấy là các bộ nhớ SRAM.

Bộ nhớ chương trình (Program memory): Là bộ nhớ Flash lập trình được, trong các chip vi điều khiển cũ (như AT90S1200 hay AT(9)2313...) bộ nhớ chương trình chỉ gồm 1 phần là Application Flash Section nhưng trong các chip mới chúng ta có thêm phần Boot Flash section. Trong tài liệu này chúng ta hiểu bộ nhớ chương trình là Application section. Thực chất, application section bao gồm 2 phần: phần chứa các instruction (mã lệnh cho hoạt động của chip) và phần chứa các vector ngắt (interrupt vectors). Các vector ngắt nằm ở phần đầu của application section (từ địa chỉ 0x0000) và dài đến bao nhiêu tùy thuộc vào loại chip. Phần chứa instruction nằm liền sau đó, chương trình viết cho chip phải được load vào phần này.



Hình 1.5. Cấu trúc của bộ nhớ

Bộ nhớ dữ liệu (data memory): Đây là phần chứa các thanh ghi quan trọng nhất của chip, việc lập trình cho chip phần lớn là truy cập bộ nhớ này. Bộ nhớ dữ liệu trên các chip AVR có độ lớn khác nhau tùy theo mỗi chip, tuy nhiên về cơ bản phần bộ nhớ này được chia thành 5 phần:

Phần 1: là phần đầu tiên trong bộ nhớ dữ liệu, như mô tả trong hình 2, phần này bao gồm 32 thanh ghi có tên gọi là register file (RF), hay General Purpose Register

– GPR, hoặc đơn giản là các Thanh ghi. Tất cả các thanh ghi này đều là các

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
-------	-------	-------	-------	-------	-------	-------	-------

Hầu hết các chip đều bao gồm 32 thanh ghi Register File có địa chỉ tuyệt đối từ 0x0000 đến 0x001F. Mỗi thanh ghi có thể chứa giá trị dương từ 0 đến

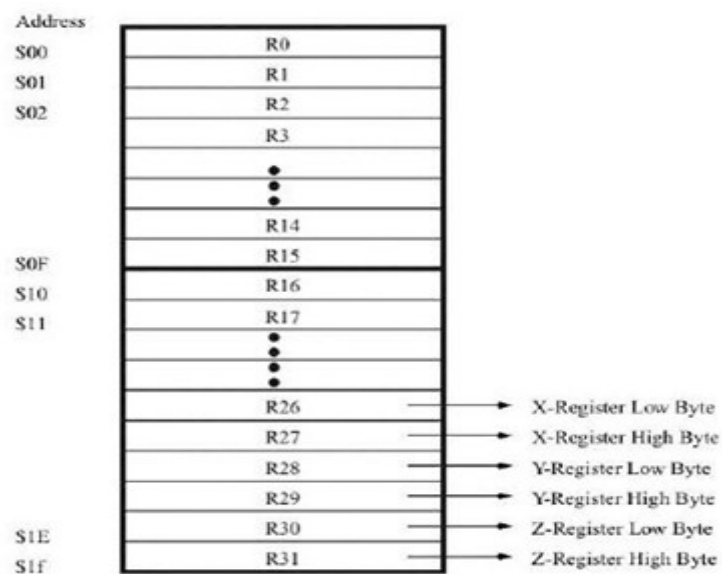
255 hoặc các giá trị có dấu từ -128 đến 127 hoặc mã ASCII của một ký tự nào đó... Các thanh ghi này được đặt tên theo thứ tự là R0 đến R31. Chúng được chia thành 2 phần, phần 1 bao gồm các thanh ghi từ R0 đến R15 và phần 2 là các thanh ghi R16 đến R31. Các thanh ghi này có các đặc điểm sau:

- Được truy cập trực tiếp trong các instruction .
- Các toán tử, phép toán thực hiện trên các thanh ghi này chỉ cần 1 chu kỳ xung clock.
- Register File được kết nối trực tiếp với bộ xử lý trung tâm – CPU của chip.
- Chúng là nguồn chứa các số hạng trong các phép toán và cũng là đích chứa kết quả trả lại của phép toán.

Tất cả các instruction sử dụng RF làm toán hạng đều có thể truy nhập tất cả các RF một cách trực tiếp trong 1 chu kỳ xung clock, ngoại trừ SBCI, SUBI, CPI, ANDI và LDI, các instruction này chỉ có thể truy nhập các thanh ghi từ R16 đến R31.

Thanh ghi R0 là thanh ghi duy nhất được sử dụng trong instruction LPM (Load Program Memory). Các thanh ghi R26, R27, R28, R29, R30 và R31 ngoài chức năng thông thường còn được sử dụng như các con trỏ (Pointer register) trong một số instruction truy xuất gián tiếp.

Tóm lại 32 RF của AVR được xem là 1 phần của CPU, vì thế chúng được CPU sử dụng trực tiếp và nhanh chóng, để gọi các thanh ghi này, chúng ta không cần gọi địa chỉ mà chỉ cần gọi trực tiếp tên của chúng. RF thường được sử dụng như các toán hạng (operand) của các phép toán trong lúc lập trình.



Hình 1.6. Mô tả các chức năng phụ của các RF.

Phần 2: là phần nằm ngay sau register file, phần này bao gồm 64 thanh

ghi được gọi là 64 thanh ghi nhập/xuất (64 I/O register) hay còn gọi là vùng nhớ I/O (I/O Memory). Vùng nhớ I/O là cửa ngõ giao tiếp giữa CPU và thiết bị ngoại vi. Tất cả các thanh ghi điều khiển, trạng thái... của thiết bị ngoại vi đều nằm ở đây. Xem lại ví dụ trong bài 1, trong đó tôi có đề cập về việc điều khiển các PORT của chip Vi điều khiển. Xa hơn, nếu muốn truy xuất các thiết bị ngoại vi khác như Timer, chuyển đổi Analog/Digital, giao tiếp USART... đều thực hiện thông qua việc điều khiển các thanh ghi trong vùng nhớ này. Vùng nhớ I/O có thể được truy cập như SRAM hay như các thanh ghi I/O. Nếu sử dụng instruction truy xuất SRAM để truy xuất vùng nhớ này thì địa chỉ của chúng được tính từ 0x0020 đến 0x005F. Nhưng nếu truy xuất như các thanh ghi I/O thì địa chỉ của chúng được tính từ 0x0000 đến 0x003F.

Để thống nhất cách sử dụng thuật ngữ, ta dùng khái niệm “địa chỉ I/O” cho các thanh ghi trong vùng nhớ I/O để nói đến địa chỉ không tính phần Register File, khái niệm “địa chỉ bộ nhớ” của thanh ghi là chỉ địa chỉ tuyệt đối của chúng trong SRAM.

Phần 3: RAM tĩnh, nội (internal SRAM), là vùng không gian cho chứa các biến (tạm thời hoặc toàn cục) trong lúc thực thi chương trình.

Phần 4: RAM ngoại (external SRAM), các chip vi xử lý đều cho phép người sử dụng gắn thêm các bộ nhớ ngoài để chứa biến, vùng này thực chất chỉ tồn tại khi nào người sử dụng gắn thêm bộ nhớ ngoài vào chip.

Phần 5: EEPROM (Electrically Erasable Programmable ROM) là một phần quan trọng của

các chip mới, vì là ROM nên bộ nhớ này không bị xóa ngay cả khi không cung cấp nguồn nuôi cho chip, rất thích hợp cho các ứng dụng lưu trữ dữ liệu. EEPROM được tách riêng và có địa chỉ tính từ 0x0000.

1.2.3. Cổng vào/ra song song

Đây là các đường tín hiệu được nối với một số chân của IC dùng để giao tiếp với thế giới bên ngoài IC. Giao tiếp ở đây là đưa điện áp ra hoặc đọc vào giá trị điện áp tại chân cổng. Các giá trị điện áp đưa ra hay đọc vào chỉ có thể được biểu diễn bởi một trong hai giá trị logic (0 hoặc 1). Trong kỹ thuật vi xử lý, người ta thường dùng quy ước logic dương: giá trị logic 0 ứng với mức điện áp thấp xấp xỉ 0VDC, giá trị logic 1 ứng với mức điện áp cao xấp xỉ +5VDC. Tùy loại vi xử lý mà “khoảng xấp xỉ” đó là khác nhau nhưng nhìn chung là tương thích với mức logic TTL. Mỗi cổng vào/ra song song thường gồm 08 đường vào/ra khác nhau và gọi là các cổng 08 bit. Các đường tín hiệu vào/ra của các cổng và thuộc cùng một cổng là độc lập với nhau. Điều đó có nghĩa là ta có thể đưa ra hay đọc vào các giá trị logic khác nhau đối với từng chân cổng (từng đường tín hiệu vào/ra). Một điều cần chú ý nữa đối với các cổng vào/ra đó là chúng có thể được tích hợp thêm (nói đúng hơn là kèm thêm) các chức năng đặc biệt liên quan đến các ngoại vi khác.

1.2.4. Cổng vào/ra nối tiếp

Khác với cổng song song, với cổng nối tiếp các bit dữ liệu được truyền lần lượt trên cùng một đường tín hiệu thay vì truyền cùng một lúc trên các đường tín hiệu khác nhau. Thông thường thì việc truyền dữ liệu bằng cổng nối tiếp

phải tuân theo một cơ chế, một giao thức hay một nguyên tắc nhất định. Có thể kể ra một số giao thức như SPI, I2C, SCI...

Cổng nối tiếp có 02 kiểu truyền dữ liệu chính:

Truyền đồng bộ (synchronous): thiết bị truyền và thiết bị nhận đều dùng chung một xung nhịp (clock). Truyền dị bộ (asynchronous): thiết bị truyền và thiết bị nhận sử dụng hai nguồn xung nhịp riêng. Tuy nhiên hai nguồn xung nhịp này không được khác nhau quá nhiều. Xung nhịp là yếu tố không thể thiếu trong truyền dữ liệu nối tiếp và nó có vai trò xác định giá trị của bit dữ liệu (hay nói đúng hơn là xác định thời điểm đọc mức logic trên đường truyền dữ liệu).

Cổng nối tiếp có thể có một trong các tính năng sau:

Đơn công: thiết bị chỉ có thể hoặc truyền hoặc nhận dữ liệu

Bán song công: thiết bị có thể truyền và nhận dữ liệu nhưng tại một thời điểm chỉ có thể làm một trong hai việc đó.

Song công: thiết bị có thể truyền và nhận dữ liệu đồng thời.

1.2.5. Bộ đếm/Bộ định thời

Đây là các ngoại vi được thiết kế để thực hiện một nhiệm vụ đơn giản: đếm các xung nhịp. Mỗi khi có thêm một xung nhịp tại đầu vào đếm thì giá trị của bộ đếm sẽ được tăng lên 01 đơn vị (trong chế độ đếm tiến/đếm lên) hay giảm đi 01 đơn vị (trong chế độ đếm lùi/đếm xuống).

Xung nhịp đưa vào đếm có thể là một trong hai loại:

Xung nhịp bên trong IC. Đó là xung nhịp được tạo ra nhờ kết hợp mạch dao động bên trong IC và các linh kiện phụ bên ngoài nối với IC. Ta có thể ví đó là “nhịp tim” để toàn bộ các phần cứng bên trong vi xử lý (bao gồm cả CPU và các ngoại vi) có thể hoạt động được. Trong trường hợp sử dụng xung nhịp loại này, người ta gọi là các bộ định thời (timers). Do xung nhịp bên loại này thường đều đặn nên ta có thể dùng để đếm thời gian một cách khá chính xác.

Xung nhịp bên ngoài IC. Đó là các tín hiệu logic thay đổi liên tục giữa 0 và 1 và không nhất thiết phải là đều đặn. Trong trường hợp này người ta gọi là các bộ đếm (counters). Ứng dụng phổ biến của các bộ đếm là đếm các sự kiện bên ngoài như đếm các sản phẩm chạy trên băng chuyền, đếm xe ra/vào kho bãi... Một khái niệm quan trọng cần phải nói đến là sự kiện “tràn” (overflow). Nó được hiểu là sự kiện bộ đếm đếm vượt quá giá trị tối đa mà nó có thể biểu diễn và quay trở về giá trị 0. Với bộ đếm 8 bit, giá trị tối đa là 255