

BỘ CÔNG THƯƠNG
TRƯỜNG CAO ĐẲNG CÔNG NGHIỆP VÀ THƯƠNG MẠI

GIÁO TRÌNH

Tên mô đun: Kỹ thuật Vi điều khiển

NGHỀ: ĐIỆN TỬ CÔNG NGHIỆP

TRÌNH ĐỘ TRUNG CẤP/CAO ĐẲNG NGHỀ

*Ban hành kèm theo Quyết định số: /QĐ-CĐCNPY, ngày tháng năm 2018
của Hiệu trưởng trường Cao đẳng Công nghiệp và Thương mại*

Vĩnh Phúc, năm 2018

MỤC LỤC

ĐỀ MỤC	TRANG
MỤC LỤC	2
CHƯƠNG TRÌNH MÔ ĐUN	4
Bài 1. ĐIỀU KHIỂN PIC 18f4520:	8
ĐẶC TÍNH, CẤU TRÚC, CHỨC NĂNG CÁC PORT	8
1.1. Giới thiệu.....	8
1.1.1. Cấu tạo vi điều khiển.	8
1.1.2. Nguyên lý hoạt động của Vi điều khiển	8
1.1.3. Lĩnh vực và ứng dụng.....	11
1.1.4. Sản phẩm dân dụng.....	12
1.2. Khảo sát vi điều khiển Microchip	12
1.2.1. Cấu hình vi điều khiển PIC 18f4520.	12
1.2.2. Cấu trúc của vi điều khiển PIC 18f4520.	13
1.2.3. Khảo sát sơ đồ chân vi điều khiển PIC 18f4520.	15
Bài 2. TỔ CHỨC BỘ NHỚ THANH GHI	22
2.1. Giới thiệu.....	22
2.2. Kiến trúc bộ nhớ.....	22
2.3. Tổ chức bộ nhớ của vi điều khiển PIC 18f4520.....	23
2.3.1. Tổ chức bộ nhớ chương trình.....	23
2.3.2. Khởi tạo dao động.....	24
2.3.3. Khảo sát bộ nhớ dữ liệu và thanh ghi trạng thái.....	26
Bài 3: LẬP TRÌNH ĐIỀU KHIỂN VỚI VĐK PIC 18F4520	30
3.1. Các thành phần cơ bản của ngôn ngữ lập trình C.....	30
3.1.1. Các kiểu dữ liệu của biến.....	30
3.1.2. Các toán tử.....	31
3.1.3. Các lệnh C cơ bản	33
3.2. Trình biên dịch C.....	36
3.2.1. Các thành phần mở rộng của trình biên dịch C	36
3.2.2. Khai báo biến và hằng số.....	37
3.2.3. Khai báo các biến truy xuất bit	38
3.2.4. Khai báo con trỏ, mảng.....	39
3.2.5. Khai báo và sử dụng hàm.....	39
3.2.6. Khai báo chương trình con phục vụ ngắt.....	40
3.2.7. Cấu trúc của một chương trình C cho vi điều khiển PIC	56
3.2.8. File thư viện của vi điều khiển PIC	57
3.3. Phần mềm MPLAB_IDE_v8_70	60
Bài 4: PORT XUẤT NHẬP, TIMER – COUNTER	68
4.1. Cấu hình, thông số điện áp, dòng của các port.....	68
4.2. Các ứng dụng dùng port điều khiển của vi điều khiển 8 bit.	70

4.2.1. Các ứng dụng điều khiển led đơn.....	70
4.2.2. Các ứng dụng điều khiển led 7 đoạn trực tiếp.....	73
4.2.3. Các ứng dụng điều khiển led 7 đoạn quét.....	74
4.2.4. Giao tiếp vi điều khiển 8 bit với LCD	76
4.3. Timer/counter của vi điều khiển PIC 18f4520	79
4.3.1. Khảo sát các timer của vi điều khiển PIC 18f4520.....	79
4.3.2. Cách truy xuất các thành phần của timer trong ngôn ngữ lập trình C.....	91
4.3.3. Ứng dụng timer/counter của vi điều khiển PIC 18f4520.....	93
a. Giao tiếp DS1307	93
b. Ứng dụng 1.....	98
c. Ứng dụng 2.....	101

CHƯƠNG TRÌNH MÔ ĐUN

Tên mô đun: KỸ THUẬT VI ĐIỀU KHIỂN

Mã mô đun: MĐTC14020040

Thời gian thực hiện mô đun: 75 giờ (Lý thuyết: 15 giờ; Thực hành, thí nghiệm, thảo luận, bài tập 57 giờ; Kiểm tra : 3 giờ)

I. Vị trí, tính chất mô đun

- Vị trí: Đây là mô đun chuyên ngành cho học sinh ngành điện tử, viễn thông. Mô đun này phải học sau khi đã học xong các mô đun: Kỹ thuật vi xử lý.
- Tính chất: Bắt buộc

II. Mục tiêu mô đun

- Kiến thức:
 - + Mô tả được cấu trúc bên trong, chức năng các chân của họ vi điều khiển PIC 18F4520.
 - + Trình bày được các lệnh lập trình C của họ vi điều khiển PIC 18F4520
 - + Trình bày được một số ứng dụng của họ vi điều khiển
- Kỹ năng:
 - + Lập được lưu đồ thuật toán của một số ứng dụng cơ bản
 - + Lập trình trên ngôn ngữ lập trình C và nạp được vào trong chip một số bài tập ứng dụng của vi điều khiển PIC 18F4520
- Về năng lực tự chủ và trách nhiệm:
 - + Dự lớp đầy đủ theo quy định, rèn luyện tác phong công nghiệp, biết cách làm việc nhóm.

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 (giờ)			
		Tổng số	Lý thuyết	Thực hành, thí nghiệm, thảo luận, bài tập	Kiểm tra
1.	Bài 1. VI ĐIỀU KHIỂN PIC 18f4520: ĐẶC TÍNH, CẤU TRÚC, CHỨC NĂNG CÁC PORT 1.1. Giới thiệu 1.2. Khảo sát vi điều khiển Microchip	9	3	6	
2.	Bài 2. TỔ CHỨC BỘ NHỚ THANH GHI 2.1. Giới thiệu.	16	3	12	1

	2.2. Kiến trúc bộ nhớ. 2.3. Tổ chức bộ nhớ của vi điều khiển PIC 18f4520.				
3.	Bài 3. LẬP TRÌNH ĐIỀU KHIỂN VỚI VĐK PIC 18F4520, 3.1. Các thành phần cơ bản của ngôn ngữ lập trình C 3.2. Trình biên dịch C. 3.3. Phần mềm MPLAB_IDE_v8_70	20	3	16	1
4.	Bài 4. PORT XUẤT NHẬP, TIMER – COUNTER 4.1. Cấu hình, thông số điện áp, dòng của các port 4.2. Các ứng dụng dùng port điều khiển của vi điều khiển 8 bit. 4.3. Timer/counter của vi điều khiển PIC 18f4520	30	6	23	1
	Cộng	75	15	27+30	3

2. Nội dung chi tiết

Bài 1. ĐIỀU KHIỂN PIC 18f4520: ĐẶC TÍNH, CẤU TRÚC, CHỨC NĂNG CÁC PORT

(Thời gian: 9 giờ)

*** Mục tiêu của bài:** Sau khi học xong bài học này, sinh viên có thể

- Trình bày được về sơ đồ cấu trúc điều khiển và các ngoại vi tích hợp của vi điều khiển PIC 18f4520
- Phân tích được sơ đồ chân vi điều khiển PIC 18f4520.
- Cần thận, sáng tạo đảm bảo an toàn thiết bị và dụng cụ.
- Nghiêm túc, khoa học, tỉ mỉ.

*** Nội dung của bài:**

- 1.1. Giới thiệu
- 1.2. Khảo sát vi điều khiển Microchip
 - 1.2.1. Cấu hình vi điều khiển PIC 18f4520.
 - 1.2.2. Sơ đồ cấu trúc của vi điều khiển PIC 18f4520.
 - 1.2.3. Khảo sát sơ đồ chân vi điều khiển PIC 18f4520.

Bài 2. TỔ CHỨC BỘ NHỚ THANH GHI

(Thời gian: 16 giờ)

*** Mục tiêu của bài:** Sau khi học xong bài học này, sinh viên có thể:

- Trình bày được về kiến trúc bộ nhớ, tổ chức bộ nhớ vi điều khiển PIC 18f4520
- Vẽ được tổ chức bộ nhớ vi điều khiển PIC 18f4520.
- Khảo sát việc kết nối với các linh kiện cơ bản như led đơn, nút nhấn, led 7 đoạn, LCD

- Cẩn thận, sáng tạo đảm bảo an toàn thiết bị và dụng cụ.
- Nghiêm túc, khoa học, tỉ mỉ.

*** Nội dung của bài:**

- 2.1. Giới thiệu.
- 2.2. Kiến trúc bộ nhớ.
- 2.3. Tổ chức bộ nhớ của vi điều khiển PIC 18f4520.
 - 2.3.1. Tổ chức bộ nhớ chương trình.
 - 2.3.2. Mã lệnh 14 bit.
 - 2.3.3. Khảo sát bộ nhớ dữ liệu và thanh ghi trạng thái.
 - 2.3.4. Bộ nhớ dữ liệu EEPROM.

Bài 3: LẬP TRÌNH ĐIỀU KHIỂN VỚI VĐK PIC 18F4520 (Thời gian: 20 giờ)

*** Mục tiêu của bài:** Sau khi học xong bài học này, sinh viên có thể:

- Trình bày được lệnh cơ bản của vi điều khiển PIC 18f4520
- Lập trình điều khiển được khi dùng ngôn ngữ C cho các hệ thống điều khiển dùng vi điều khiển PIC 18f4520
- Phân tích được thư viện của vi điều khiển PIC 18f4520.
- Cẩn thận, sáng tạo đảm bảo an toàn thiết bị và dụng cụ.
- Nghiêm túc, khoa học, tỉ mỉ.

*** Nội dung của bài:**

- 3.1. Các thành phần cơ bản của ngôn ngữ lập trình C
 - 3.1.1. Các kiểu dữ liệu của biến.
 - 3.1.2. Các toán tử.
 - 3.1.3. Các lệnh C cơ bản
- 3.2. Trình biên dịch C.
 - 3.2.1. Các thành phần mở rộng của trình biên dịch C
 - 3.2.2. Khai báo biến và hằng số
 - 3.2.3. Khai báo các biến truy xuất bit
 - 3.2.4. Khai báo con trỏ, mảng.
 - 3.2.5. Khai báo và sử dụng hàm
 - 3.2.6. Khai báo chương trình con phục vụ ngắt.
 - 3.2.7. Cấu trúc của một chương trình C cho vi điều khiển PIC
 - 3.2.8. File thư viện của vi điều khiển PIC
- 3.3. Phần mềm MPLAB_IDE_v8_70

Bài 4: PORT XUẤT NHẬP, TIMER – COUNTER (Thời gian: 30 giờ)

*** Mục tiêu của bài:** Sau khi học xong bài học này, sinh viên có thể

- Trình bày được về cấu hình, thông số điện áp, dòng của các port vi điều khiển PIC 18f4520
- Trình bày được cách truy xuất port điều khiển PIC 18f4520
- Lập trình được các ứng dụng truy xuất port bằng ngôn ngữ C với LED đơn, 7 thanh, LCD
- Lập trình được các ứng dụng Timer/counter của vi điều khiển PIC 18f4520

- Cẩn thận, sáng tạo đảm bảo an toàn thiết bị và dụng cụ.
- Nghiêm túc, khoa học, tỉ mỉ.

*** Nội dung của bài:**

4.1. Cấu hình, thông số điện áp, dòng của các port

4.1.1. Truy xuất port điều khiển của vi điều khiển 8 bit.

4.1.2. Định cấu hình cho port

4.1.3. Lập trình truy xuất port bằng ngôn ngữ C

4.2. Các ứng dụng dùng port điều khiển của vi điều khiển 8 bit.

4.2.1. Các ứng dụng điều khiển led đơn.

4.2.2. Các ứng dụng điều khiển led 7 đoạn trực tiếp.

4.2.3. Các ứng dụng điều khiển led 7 đoạn quét.

4.2.4. Giao tiếp vi điều khiển 8 bit với LCD

4.3. Timer/counter của vi điều khiển PIC 18f4520

4.3.1. Khảo sát các timer của vi điều khiển PIC 18f4520.

4.3.2. Cách truy xuất các thành phần của timer trong ngôn ngữ lập trình C.

4.3.3. Ứng dụng timer/counter của vi điều khiển PIC 18f4520

Bài 1. ĐIỀU KHIỂN PIC 18f4520: ĐẶC TÍNH, CẤU TRÚC, CHỨC NĂNG CÁC PORT

*** Mục tiêu**

- Trình bày được về sơ đồ cấu trúc điều khiển và các ngoại vi tích hợp của vi điều khiển PIC 18f4520
- Phân tích được sơ đồ chân vi điều khiển PIC 18F4520.
- Cẩn thận, sáng tạo đảm bảo an toàn thiết bị và dụng cụ.
- Nghiêm túc, khoa học, tỉ mỉ.

*** Nội dung của bài:**

1.1. Giới thiệu

1.1.1. Cấu tạo vi điều khiển.

- Vi điều khiển là một máy tính được tích hợp trên một chip, nó thường được sử dụng để điều khiển các thiết bị điện tử. Vi điều khiển thực chất gồm một vi xử lý có hiệu suất đủ cao và giá thành thấp (so với các vi xử lý đa năng dùng trong máy tính) kết hợp với các thiết bị ngoại vi như các bộ nhớ, các mô đun vào/ra, các mô đun biến đổi từ số sang tương tự và từ tương tự sang số, mô đun điều chế độ rộng xung (PWM)...
- Vi điều khiển thường được dùng để xây dựng hệ thống nhúng. Nó xuất hiện nhiều trong các dụng cụ điện tử, thiết bị điện, máy giặt, lò vi sóng, điện thoại, dây truyền tự động...
- Hầu hết các loại vi điều khiển hiện nay có cấu trúc Harvard là loại cấu trúc mà bộ nhớ chương trình và bộ nhớ dữ liệu được phân biệt riêng.
- Cấu trúc của một vi điều khiển gồm CPU, bộ nhớ chương trình (thường là bộ nhớ ROM hoặc bộ nhớ Flash), bộ nhớ dữ liệu (RAM), các bộ định thời, các cổng vào/ra để giao tiếp với các thiết bị bên ngoài, tất cả các khối này được tích hợp trên một vi mạch.

Các loại vi điều khiển trên thị trường hiện nay:

- VDK MCS-51: 8031, 8032, 8051, 8052,...
- VDK ATMEL: 89Cxx, AT89Cxx51...
- VDK AVR AT90Sxxxx
- VDK PIC 16C5x, 17C43, 18F4520,

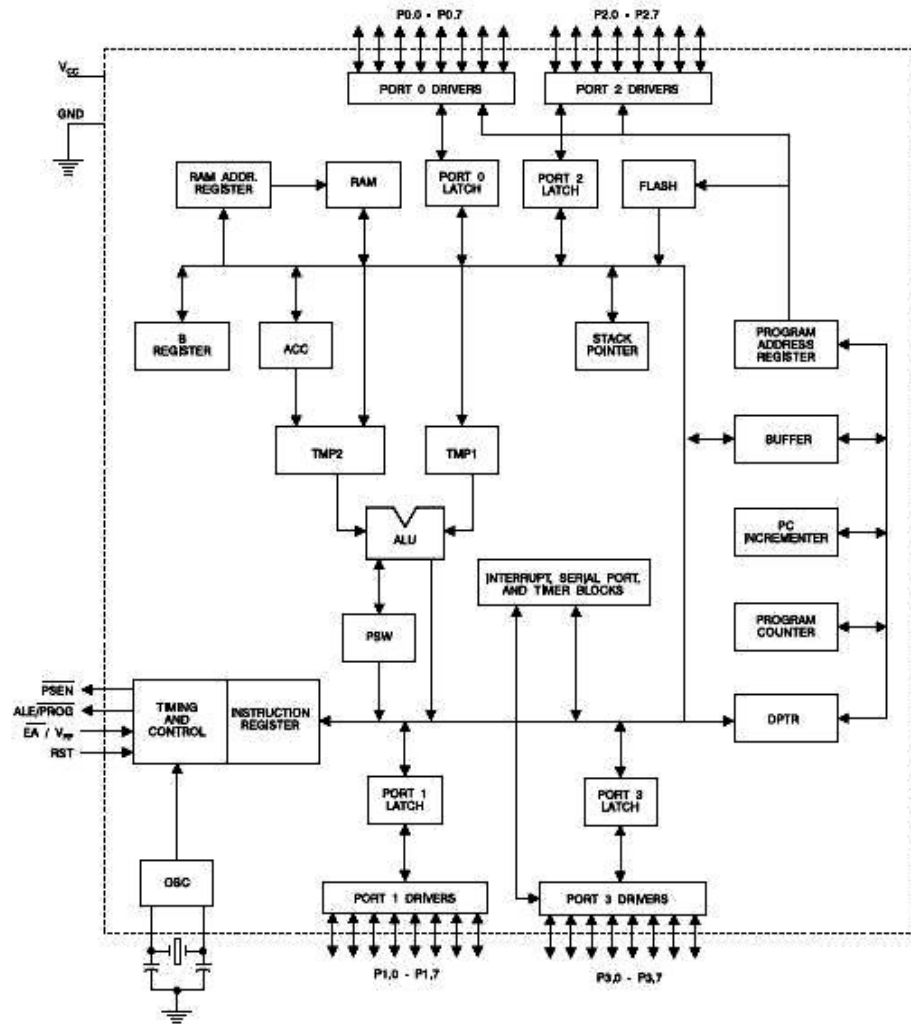
1.1.2. Nguyên lý hoạt động của Vi điều khiển

Mặc dù đã có rất nhiều họ vi điều khiển được phát triển cũng như nhiều chương trình điều khiển tạo ra cho chúng, nhưng tất cả chúng vẫn có một số điểm chung cơ bản. Do đó nếu ta hiểu cặn kẽ một họ thì việc tìm hiểu thêm một họ vi điều khiển mới là hoàn toàn đơn giản. Một kịch bản chung cho hoạt động của một vi điều khiển như sau:

- Khi không có nguồn điện cung cấp, vi điều khiển chỉ là một con chip có chương trình nạp sẵn vào trong đó và không có hoạt động gì xảy ra.
- Khi có nguồn điện, mọi hoạt động bắt đầu được xảy ra với tốc độ cao. Đơn vị điều khiển logic có nhiệm vụ điều khiển tất cả mọi hoạt động. Nó khóa tất cả các mạch khác, trừ mạch giao động thạch anh. Sau vài mili giây đầu tiên tất cả đã sẵn

sàng hoạt động.

- Điện áp nguồn nuôi đạt đến giá trị tối đa của nó và tần số giao động trở nên ổn định. Các bit của các thanh ghi SFR cho biết trạng thái của tất cả các mạch trong vi điều khiển. Toàn bộ vi điều khiển hoạt động theo chu kỳ của chuỗi xung chính.
- Thanh ghi bộ đếm chương trình (Program Counter) được xóa về 0. Câu lệnh từ địa chỉ này được gửi tới bộ giải mã lệnh sau đó được thực thi ngay lập tức.
- Giá trị trong thanh ghi PC được tăng lên 1 và toàn bộ quá trình được lặp lại vài ... triệu lần trong một giây.



Hình 1.1 Cấu trúc bên trong của vi điều khiển.

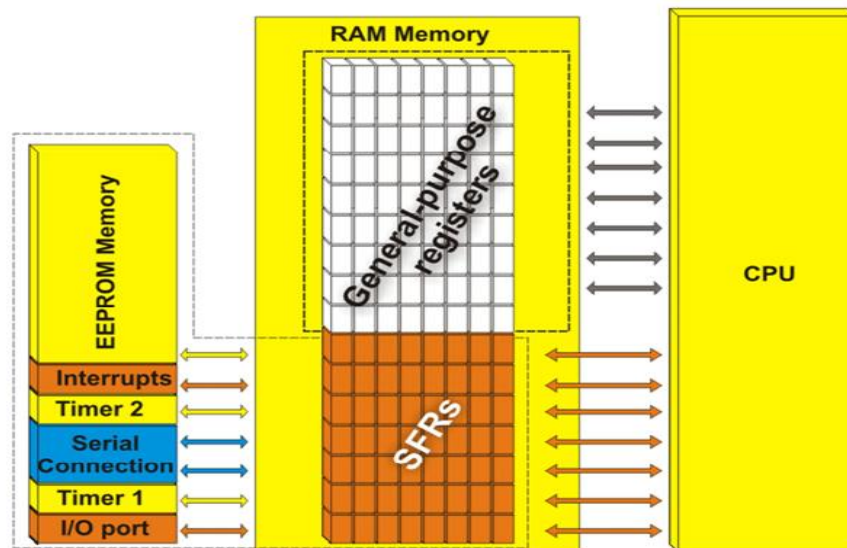
- Memory (bộ nhớ): là ROM/RAM lưu trữ chương trình hay các kết quả trung gian.
- + Read Only Memory (ROM): Read Only Memory (ROM) là một loại bộ nhớ được sử dụng để lưu vĩnh viễn các chương trình được thực thi. Kích cỡ của chương trình có thể được viết phụ thuộc vào kích cỡ của bộ nhớ này. ROM có thể được tích hợp trong vi điều khiển hay thêm vào như là một chip gắn bên ngoài, tùy thuộc vào loại vi điều khiển. Cả hai tùy chọn có một số nhược điểm. Nếu ROM được thêm vào như là một chip bên ngoài, các vi điều khiển là rẻ hơn và các chương trình có thể tồn tại lâu hơn đáng kể. Đồng thời, làm giảm số lượng các

chân vào/ra để vi điều khiển sử dụng với mục đích khác. ROM nội thường là nhỏ hơn và đắt tiền hơn, nhưng có thêm lá ghim sẵn để kết nối với môi trường ngoại vi. Kích thước của dãy ROM từ 512B đến 64KB.

+ Random Access Memory (RAM): Random Access Memory (RAM) là một loại bộ nhớ sử dụng cho các dữ liệu lưu trữ tạm thời và kết quả trung gian được tạo ra và được sử dụng trong quá trình hoạt động của bộ vi điều khiển. Nội dung của bộ nhớ này bị xóa một khi nguồn cung cấp bị tắt.

- Electrically Erasable Programmable ROM (EEPROM) (hình 1.2)

EEPROM là một kiểu đặc biệt của bộ nhớ chỉ có ở một số loại vi điều khiển. Nội dung của nó có thể được thay đổi trong quá trình thực hiện chương trình (tương tự như RAM), nhưng vẫn còn lưu giữ vĩnh viễn, ngay cả sau khi mất điện (tương tự như ROM). Nó thường được dùng để lưu trữ các giá trị được tạo ra và được sử dụng trong quá trình hoạt động (như các giá trị hiệu chuẩn, mã, các giá trị để đếm, v.v..), mà cần phải được lưu sau khi nguồn cung cấp ngắt. Một bất lợi của bộ nhớ này là quá trình ghi vào tương đối chậm.



Hình 1.2 Giao tiếp bộ nhớ

- Bộ đếm chương trình (PC: Program Counter): Bộ đếm chương trình chứa địa chỉ chỉ đến ô nhớ chứa câu lệnh tiếp theo sẽ được kích hoạt. Sau mỗi khi thực hiện lệnh, giá trị của bộ đếm được tăng lên 1. 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).

- CPU-Central Processing Unit(Đơn vị xử lý trung tâm): 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).

+ Bộ giải mã lệnh có nhiệm vụ nhận dạng câu lệnh và điều khiển các mạch khác theo lệnh đã giải mã. Việc giải mã được thực hiện nhờ có tập lệnh “instruction set”. Mỗi họ vi điều khiển thường có các tập lệnh khác nhau.

+ *Thanh ghi tích lũy (Accumulator)* là một thanh ghi SFR liên quan mật thiết với hoạt động của ALU. Nó lưu trữ tất cả các dữ liệu cho quá trình tính toán và lưu giá trị kết quả để chuẩn bị cho các tính toán tiếp theo. Một trong các thanh ghi SFR khác được gọi là thanh ghi trạng thái (Status Register) cho biết trạng thái của các giá trị lưu trong thanh ghi tích lũy.

+ *Arithmetical Logical Unit (ALU)*: Thực thi tất cả các thao tác tính toán số học và logic.

- Các thanh ghi chức năng đặc biệt (SFR): Thanh ghi chức năng đặc biệt (Special Function Registers) là một phần của bộ nhớ RAM. Mục đích của chúng được định trước bởi nhà sản xuất và không thể thay đổi được. Các bit của chúng được liên kết vật lý tới các mạch trong vi điều khiển như bộ chuyển đổi A/D, modul truyền thông nối tiếp,... Mỗi sự thay đổi trạng thái của các bit sẽ tác động tới hoạt động của vi điều khiển hoặc các vi mạch.

- Các cổng vào/ra (I/O Ports): Để vi điều khiển có thể hoạt động hữu ích, nó cần có sự kết nối với các thiết bị ngoại vi. Mỗi vi điều khiển sẽ có một hoặc một số thanh ghi (được gọi là cổng) được kết nối với các chân của vi điều khiển. Chúng có thể thay đổi chức năng, chiều vào/ra theo yêu cầu của người dùng.

- Address bus(Bus địa chỉ): Là các đường tín hiệu song song 1 chiều nối từ CPU đến bộ nhớ, CPU gửi giá trị địa chỉ của ô nhớ cần truy nhập (đọc/ghi) trên các đường tín hiệu này. Độ rộng của bus địa chỉ là n (là số các đường tín hiệu, với n có thể là 8, 18, 20, 24, 32 hay 64), khi đó số ô nhớ mà CPU có thể địa chỉ hóa được sẽ là 2^n

- Data bus(Bus dữ liệu): Là các đường tín hiệu song song 2 chiều, nhiều thiết bị khác nhau có thể được nối với bus dữ liệu, nhưng tại một thời điểm chỉ có 1 thiết bị duy nhất có thể được phép đưa dữ liệu lên bus. Độ rộng Bus dữ liệu là m (với m có thể là 4, 8, 16, 32 hay 64), khi đó số bit mà mỗi một chu kỳ đọc/ghi có thể truyền trên bus là m bits.

- Control bus(Bus điều khiển): CPU gửi tín hiệu thông qua bus này để điều khiển mọi hoạt động của hệ thống. Các tín hiệu điều khiển thường là: đọc/ ghi bộ nhớ, đọc/ ghi cổng vào/ra,...

1.1.3. Lĩnh vực và ứng dụng.

Về cơ bản, vi điều khiển rất đơn giản. Chúng chỉ bao gồm tối thiểu một số thành phần sau:

- Một bộ vi xử lý tối giản được sử dụng như bộ não của hệ thống.
- Tùy theo công nghệ của mỗi hãng sản xuất, có thể có thêm bộ nhớ, các chân nhập/xuất tín hiệu, bộ đếm, bộ định thời, các bộ chuyển đổi tương tự/số (A/D), ...
- Tất cả chúng được đặt trong một vỏ chip tiêu chuẩn.
- Một phần mềm đơn giản có thể điều khiển được toàn bộ hoạt động của vi điều khiển và có thể dễ dàng cho người sử dụng nắm bắt.

Dựa trên nguyên tắc cơ bản trên, rất nhiều họ vi điều khiển đã được phát triển và ứng dụng một cách thâm nhập nhưng mạnh mẽ vào mọi mặt của đời sống của con người. Một số ứng dụng cơ bản thành công có thể kể ra sau đây.

1.1.4. 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.
- + 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.

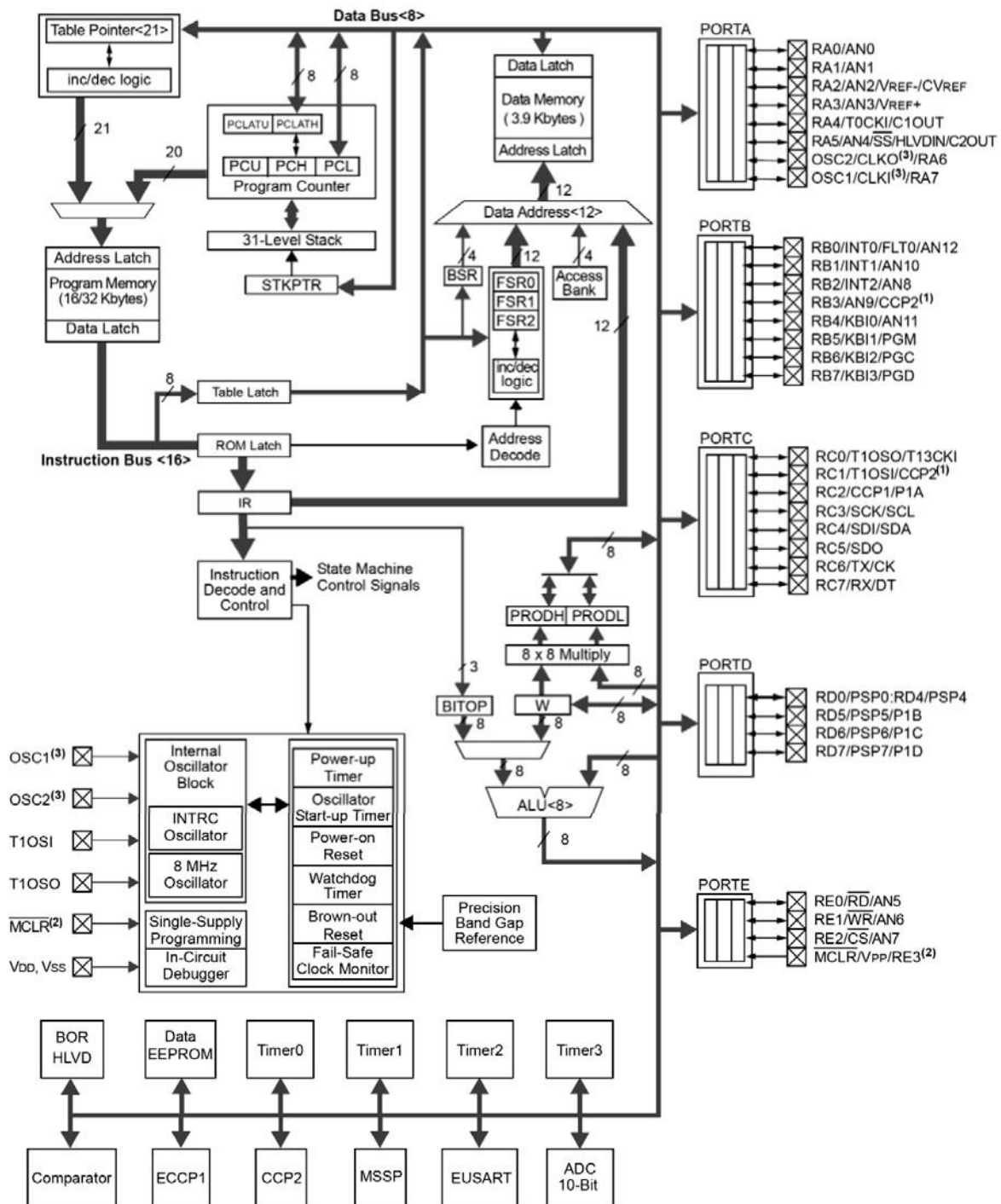
- + Điều khiển động cơ
- + Đo lường (đo điện áp, đo dòng điện, áp suất, nhiệt độ,...)
- + Cân bằng tải, cân toa xe, cân ô tô,...
- + Điều khiển các dây truyền sản xuất công nghiệp
- + Làm bộ điều khiển trung tâm cho Robot

Hướng phát triển.

- + Kết hợp các bộ vi xử lý và vi điều khiển trong các sản phẩm hệ thống nhúng.
- + Sử dụng tốt nhất các tính năng của vi điều khiển: tốc độ mà bộ vi điều khiển hỗ trợ, dung lượng bộ nhớ RAM và ROM trên chip,...
- + Tìm hiểu được khả năng phát triển các sản phẩm xung quanh.
- + Nghiên cứu các bộ vi điều khiển 8051 từ các hãng khác nhau: 8751, AT89C51, DS500,...

1.2. Khảo sát vi điều khiển Microchip

1.2.1. Cấu hình vi điều khiển PIC 18f4520.



Hình 1.3 Sơ đồ khối của PIC18F4520

1.2.2. Cấu trúc của vi điều khiển PIC 18f4520.

Các khối chính trên PIC 18F4520 bao gồm:

- **Bộ xử lý trung tâm CPU(Central Processing Unit):**
 - Tần số làm việc tối đa 40 MHz, sản xuất bằng công nghệ nanoWatt.
 - Thiết kế theo kiến trúc Harvard, tập lệnh RISC.
 - Sử dụng kỹ thuật đường ống lệnh (Instruction Pipelining).

- Đơn vị logic học (ALU: Arithmetic Logical Unit).
- Thanh ghi làm việc (WREG: work register).
- Bộ nhân bằng phần cứng (8x8 Multiply), kết quả được chứa trong cặp thanh ghi (PRODH, PRODL).
- Thanh ghi đếm chương trình (PC: Program Counter) có 21 bit bao gồm thanh ghi PCL(PC-Low) chứa các bit từ 7-0, thanh ghi PCH(PC-High) chứa các bit từ 15-8, thanh ghi PCU(PC-upper) chứa các bit từ 20-16.
- Thanh ghi con trỏ ngăn xếp STKPTR (Stack Pointer).
- 31 mức ngăn xếp (31 level stack).
- Thanh ghi lựa chọn băng (BSR: Bank Select Register).
- Thanh ghi con trỏ dữ liệu gián tiếp FSR (Indirect Data Memory Address Pointer).
- **Bộ nhớ (Memory):**
 - Bộ nhớ chương trình (Program Memory) bao gồm 32 Kbytes bộ nhớ ROM (Read-only Memory) kiểu Flash.
 - Bộ nhớ dữ liệu (Data Memory) bao gồm 1536 byte SRAM (Static Random Access Memory), 256 byte EEPROM.
- Bộ phát xung hệ thống (Oscillator): Nguồn xung từ bên ngoài hoặc từ bộ phát xung của hệ thống sẽ được đi qua bộ nhân hoặc chia tần số để lựa chọn lấy tần số thích hợp để làm xung hệ thống.
 - Nguồn xung chính được đưa vào chip qua chân OSC1 và OSC2.
 - Nguồn xung phụ được đưa vào chip qua các chân T1OSI, T1OSO.
 - Bộ phát xung nội INTRC tần số 31 kHz.
 - Bộ phát xung nội trên chip tần số 8 MHz.
- Watch dog Timer (WDT): WDT là một bộ timer có chức năng đặc biệt. Nếu được cho phép WDT sẽ hoạt động và khi tràn sẽ khởi động lại hệ thống. Thời gian khởi động lại hệ thống có thể lựa chọn được từ 4ms đến 131,072s. WDT sẽ được khởi tạo ở đầu chương trình, trong thân chương trình sẽ được -chèn các lệnh reset WDT sao cho khi MC thực hiện các đúng tuần tự các lệnh, WDT chưa bị tràn. Mục đích chính của việc sử dụng WDT là tránh cho vi điều khiển vô tình thực hiện phải một vòng lặp chết (dead loop) mà không thoát ra được. Khi đó do không thực hiện được các lệnh reset WDT nên MC sẽ tràn, tự động reset lại hệ thống, thoát khỏi tình trạng "bị treo" trong vòng lặp chết. Ngoài ra do có thể hoạt động trong khi MC -ngủ (Sleep Mode) nên WDT còn được sử dụng trong các ứng dụng tiết kiệm năng lượng.
- Bộ nạp chương trình: Bộ nạp chương trình nối tiếp trên chip (Single - Supply In - Circuit Serial Programming) sẽ giúp nạp chương trình từ mạch nạp vào bộ nhớ ROM qua các chân PGM, PGC và PGD.
- Bộ Debugger (In-Circuit Debugger): Mạch Debugger trên chip sẽ giúp người lập trình kiểm soát lỗi chương trình bằng cách cho vi điều khiển hoạt động ở chế độ chạy từng lệnh, nhóm lệnh hay toàn bộ chương trình.
- Khởi phát hiện tín hiệu reset: Mạch phát hiện tín hiệu reset có khả năng phát

hiện 03 nguồn reset:

- Reset từ chân MCLR.
- Reset khi bật nguồn (POR: Power-on Reset).
- Reset khi nguồn yếu (BOR : Brown-out Reset).
- Khối quản lý lỗi bộ phát xung (Fail-Safe Clock Monitor): Khối này được sử dụng để quản lý an toàn bộ phát xung hệ thống.
- Khối định thời khởi động bộ phát xung (Oscillator Start-up Timer): Khối này sử dụng để tạo thời gian trễ chờ cho bộ phát xung ổn định.
- Thiết bị ngoại vi (Peripheral): PIC18f4520 được tích hợp các thiết bị ngoại vi sau:

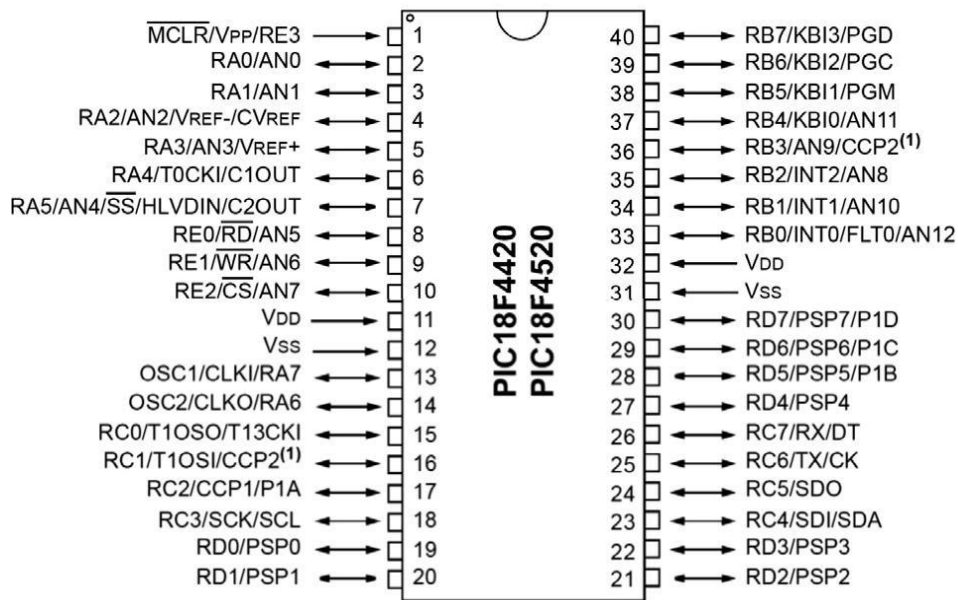
- Bộ phát hiện điện áp cao/thấp HLVD(High/low-Voltage Detect).
- Bộ nhớ lưu dữ liệu khi tắt nguồn EEPROM.
- 04 bộ đếm, định thời 16 bit: Timer0, Timer1, Timer2 và Timer3.
- 01 bộ so sánh tín hiệu tương tự (Comparator).
- 02 bộ CCP1, CCP2 (Capture, Compare, Pwm : Chụp, So sánh, xung Pwm); 01 bộ ECCP (Enhanced CCP).
- 01 cổng truyền thông nối tiếp đồng bộ (Master Synchronous Serial Port) có thể hoạt động được ở chế độ SPI hoặc I2C.
- 01 cổng truyền thông nối tiếp đồng bộ/không đồng bộ tăng EUSART (Enhanced Universal Synchronous Asynchronous Receiver Transmitter), giúp vi điều khiển PIC có thể giao tiếp với nhau hoặc giao tiếp với cổng COM của máy tính.
- 13 kênh biến đổi tương tự - số (ADC) độ phân giải 10 bit.
- Khối giao tiếp vào/ra số:

Vi điều khiển PIC18F4520 có 5 cổng vào/ra A, B, C, D và E. Mỗi cổng có một thanh ghi đệm dữ liệu tương ứng là PORTA, PORTB, PORTC, PORTD và PORTE, các thanh ghi này được định địa chỉ theo byte và theo bit.

- PORTA : RA7 - RA0.
- PORTB : RB7 - RB0.
- PORTC : RC7 - RC0.
- PORTD : RD3 - RD0.
- PORTE : RE3-RE0.

1.2.3. Khảo sát sơ đồ chân vi điều khiển PIC 18f4520.

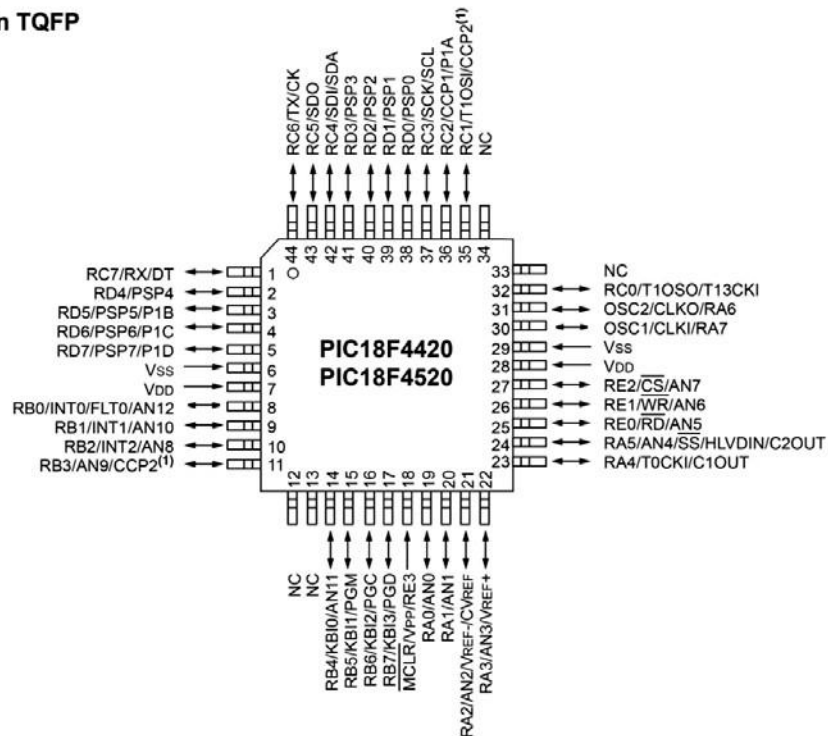
- Sơ đồ chân dạng PDIP (Lead Plastic Dual In-Line Package) hai hàng chân cắm hai bên.



Hình 1.4 Sơ đồ chân PIC 18F4520 dạng PDIP

- TQFP (Thin Quad Flat Package) bốn hàng chân dán vỏ mỏng

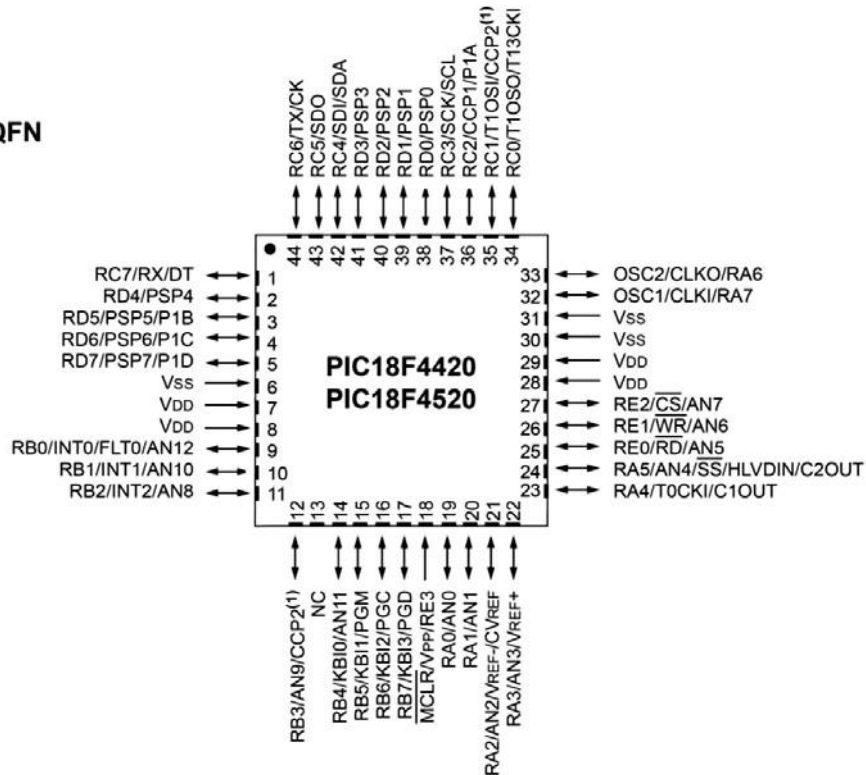
44-pin TQFP



Hình 1.5 Sơ đồ chân PIC 18F4520 dạng TQFP

- Sơ đồ chân dạng QFN (Quad Flat No-lead) bốn hàng chân dán dưới đế

44-pin QFN



Hình 1.6 Sơ đồ chân PIC 18F4520 dạng QFN

Pin Name	Pin Number			Pin Type	Description
	PDIP	QFN	TQFP		
MCLR/VPP/RE3 MCLR VPP RE3	1	18	18	I P I	Master Clear (input) or programming voltage (input). Master Clear (Reset) input. This pin is an active-low Reset to the device. Programming voltage input. Digital input.
OSC1/CLKI/RA7 OSC1 CLKI RA7	13	32	30	I I I/O	Oscillator crystal or external clock input. Oscillator crystal input or external clock source input. ST buffer when configured in RC mode; analog otherwise. External clock source input. Always associated with pin function OSC1. (See related OSC1/CLKI, OSC2/CLKO pins.) General purpose I/O pin.
OSC2/CLKO/RA6 OSC2 CLKO RA6	14	33	31	O O I/O	Oscillator crystal or clock output. Oscillator crystal output. Connects to crystal or resonator in Crystal Oscillator mode. In RC mode, OSC2 pin outputs CLKO which has 1/4 the frequency of OSC1 and denotes the instruction cycle rate. General purpose I/O pin.

Pin Name	Pin Number			Pin Type	Description
	PDIP	QFN	TQFP		

					PORTA is a bidirectional I/O port.
RA0/AN0	2	19	19		
RA0				I/O	Digital I/O.
AN0				I	Analog input 0.
RA1/AN1	3	20	20		
RA1				I/O	Digital I/O.
AN1				I	Analog input 1.
RA2/AN2/VREF-/CVREF	4	21	21		
RA2				I/O	Digital I/O.
AN2				I	Analog input 2.
VREF-				I	A/D reference voltage (low) input.
CVREF				O	Comparator reference voltage output.
RA3/AN3/VREF+	5	22	22		
RA3				I/O	Digital I/O.
AN3				I	Analog input 3.
VREF+				I	A/D reference voltage (high) input.
RA4/T0CKI/C1OUT	6	23	23		
RA4				I/O	Digital I/O.
T0CKI				I	Timer0 external clock input.
C1OUT				O	Comparator 1 output.
RA5/AN4/SS/HLVDIN/C2OUT	7	24	24		
RA5				I/O	Digital I/O.
AN4				I	Analog input 4.
SS				I	SPI slave select input.
HLVDIN				I	High/Low-Voltage Detect input.
C2OUT				O	Comparator 2 output.
RA6					See the OSC2/CLKO/RA6 pin.
RA7					See the OSC1/CLKI/RA7 pin.

Pin Name	Pin Number			Pin Type	Description
	PDIP	QFN	TQFP		
RB0/INT0/FLT0/AN12 RB0 INT0 FLT0 AN12	33	9	8	I/O I I I	PORTB is a bidirectional I/O port. PORTB can be software programmed for internal weak pull-ups on all inputs. Digital I/O. External interrupt 0. PWM Fault input for Enhanced CCP1. Analog input 12.
RB1/INT1/AN10 RB1 INT1 AN10	34	10	9	I/O I I	Digital I/O. External interrupt 1. Analog input 10.
RB2/INT2/AN8 RB2 INT2 AN8	35	11	10	I/O I I	Digital I/O. External interrupt 2. Analog input 8.
RB3/AN9/CCP2 RB3 AN9 CCP2 ⁽¹⁾	36	12	11	I/O I I/O	Digital I/O. Analog input 9. Capture 2 input/Compare 2 output/PWM2 output.

RB4/KBI0/AN11 RB4 KBI0 AN11	37	14	14	I/O I I	Digital I/O. Interrupt-on-change pin. Analog input 11.
RB5/KBI1/PGM RB5 KBI1 PGM	38	15	15	I/O I I/O	Digital I/O. Interrupt-on-change pin. Low-Voltage ICSP™ Programming enable pin.
RB6/KBI2/PGC RB6 KBI2 PGC	39	16	16	I/O I I/O	Digital I/O. Interrupt-on-change pin. In-Circuit Debugger and ICSP programming clock pin.
RB7/KBI3/PGD RB7 KBI3 PGD	40	17	17	I/O I I/O	Digital I/O. Interrupt-on-change pin. In-Circuit Debugger and ICSP programming data pin.

Pin Name	Pin Number			Pin Type	Description
	PDIP	QFN	TQFP		
					PORTC is a bidirectional I/O port.
RC0/T1OSO/T13CKI	15	34	32		
RC0				I/O	Digital I/O.
T1OSO				O	Timer1 oscillator output.
T13CKI				I	Timer1/Timer3 external clock input.
RC1/T1OSI/CCP2	16	35	35		
RC1				I/O	Digital I/O.
T1OSI				I	Timer1 oscillator input.
CCP2 ⁽²⁾				I/O	Capture 2 input/Compare 2 output/PWM2 output.
RC2/CCP1/P1A	17	36	36		
RC2				I/O	Digital I/O.
CCP1				I/O	Capture 1 input/Compare 1 output/PWM1 output.
P1A				O	Enhanced CCP1 output.
RC3/SCK/SCL	18	37	37		
RC3				I/O	Digital I/O.
SCK				I/O	Synchronous serial clock input/output for SPI mode.
SCL				I/O	Synchronous serial clock input/output for I²C™ mode.
RC4/SDI/SDA	23	42	42		
RC4				I/O	Digital I/O.
SDI				I	SPI data in.
SDA				I/O	I²C data I/O.
RC5/SDO	24	43	43		
RC5				I/O	Digital I/O.
SDO				O	SPI data out.
RC6/TX/CK	25	44	44		
RC6				I/O	Digital I/O.
TX				O	EUSART asynchronous transmit.
CK				I/O	EUSART synchronous clock (see related RX/DT).
RC7/RX/DT	26	1	1		
RC7				I/O	Digital I/O.
RX				I	EUSART asynchronous receive.
DT				I/O	EUSART synchronous data (see related TX/CK).

Pin Name	Pin Number			Pin Type	Description
	PDIP	QFN	TQFP		
					PORTD is a bidirectional I/O port or a Parallel Slave Port (PSP) for interfacing to a microprocessor port. These pins have TTL input buffers when PSP module is enabled.
RD0/PSP0	19	38	38		
RD0				I/O	Digital I/O.
PSP0				I/O	Parallel Slave Port data.
RD1/PSP1	20	39	39		
RD1				I/O	Digital I/O.
PSP1				I/O	Parallel Slave Port data.
RD2/PSP2	21	40	40		
RD2				I/O	Digital I/O.
PSP2				I/O	Parallel Slave Port data.
RD3/PSP3	22	41	41		
RD3				I/O	Digital I/O.
PSP3				I/O	Parallel Slave Port data.
RD4/PSP4	27	2	2		
RD4				I/O	Digital I/O.
PSP4				I/O	Parallel Slave Port data.
RD5/PSP5/P1B	28	3	3		
RD5				I/O	Digital I/O.
PSP5				I/O	Parallel Slave Port data.
P1B				O	Enhanced CCP1 output.
RD6/PSP6/P1C	29	4	4		
RD6				I/O	Digital I/O.
PSP6				I/O	Parallel Slave Port data.
P1C				O	Enhanced CCP1 output.
RD7/PSP7/P1D	30	5	5		
RD7				I/O	Digital I/O.
PSP7				I/O	Parallel Slave Port data.
P1D				O	Enhanced CCP1 output.

Pin Name	Pin Number			Pin Type	Description
	PDIP	QFN	TQFP		
					PORTE is a bidirectional I/O port.
RE0/RD/AN5 RE0 RD AN5	8	25	25	I/O I I	Digital I/O. Read control for Parallel Slave Port (see also WR and CS pins). Analog input 5.
RE1/WR/AN6 RE1 WR AN6	9	26	26	I/O I I	Digital I/O. Write control for Parallel Slave Port (see CS and RD pins). Analog input 6.
RE2/CS/AN7 RE2 CS AN7	10	27	27	I/O I I	Digital I/O. Chip Select control for Parallel Slave Port (see related RD and WR). Analog input 7.
RE3	—	—	—	—	See MCLR/VPP/RE3 pin.