**ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN**

**KHOA VẬT LÝ – VẬT LÝ KỸ THUẬT**

**BỘ MÔN VẬT LÝ TIN HỌC**

**----------------□□----------------**

**A blue and white logo

Description automatically generated**

**BÁO CÁO ĐỒ ÁN MÔN HỌC**

**MÔN MẠCH ĐIỆN TỬ VÀ KỸ THUẬT SỐ**

***Đề tài:***

**BỘ ĐẾM THỜI GIAN**

**Lớp: 21VLTH**

**Nhóm thực hiện: Nhóm 02**

**GV hướng dẫn: Thầy Hồ Văn Bình**

**----------------------------------**

**TP. HỒ CHÍ MINH – 2024**

**Lời Cảm Ơn**

Đầu tiên, chúng em xin gửi lời cảm ơn chân thành đến cán bộ hướng dẫn cảm ơn thầy Bình đã dạy dỗ, truyền đạt những kiến thức quý báu cho chúng em trong suốt thời gian học tập vừa qua. Trong thời gian tham gia lớp học Vi Điều Khiển, chúng em đã có thêm cho mình những kiến thức bổ ích, tinh thần học tập hiệu quả và nghiêm túc. Đây chắc chắn sẽ là những kinh nghiệm quý báu, hành trang vững chắc để chúng em có thể vững bước sau này. Môn học này vô cùng bổ ích và có tính thực tế cao, đảm bảo cung cấp đủ kiến thức cho sinh viên sau khi ra trường.

Kính mong thầy xem xét và góp ý để báo cáo của chúng em hoàn thiện hơn.

Chúng em xin chân thành cảm ơn!

TP Hồ Chí Minh, tháng 5 năm 2024

Danh sách thành viên nhóm:

|  |  |  |
| --- | --- | --- |
| STT | Họ và tên | MSSV |
| 1 | Nguyễn Hoàng Hải Đông | 21130135 |
| 2 | Võ Nguyễn Minh Khang | 21130180 |
| 3 | Lê Hải Đăng | 21130124 |
| 4 | Nguyễn Hoàng Việt | 21130322 |
| 5 | Lê Tấn Phát | 21130230 |
| 6 | Trương Gia Bảo | 21130115 |

**Mục lục**

[PHẦN 1: LỜI MỞ ĐẦU 5](#_Toc166867004)

[PHẦN 2 CƠ SỞ LÝ THUYẾT 6](#_Toc166867005)

[2.1. Linh kiện 6](#_Toc166867006)

[2.2. Chức năng của các bộ phận trong mạch 7](#_Toc166867007)

[2.2.1 PIC 16F877: 7](#_Toc166867008)

[2.2.1.1 Giới thiệu: 7](#_Toc166867009)

[2.2.1.2 Sơ đồ cấu trúc của vi điều khiển PIC 16F887: 13](#_Toc166867010)

[2.2.1.3 Khảo sát sơ đồ chân vi điều khiển PIC16F887: 16](#_Toc166867011)

[2.2.3 Tụ điện: 23](#_Toc166867012)

[2.2.4 Điện trở: 23](#_Toc166867013)

[2.2.5 Transistor: 23](#_Toc166867014)

[2.2.6 Domino 2: 23](#_Toc166867015)

[2.2.7 Nút nhấn 2 chân: 24](#_Toc166867016)

[2.2.8 Rào đơn 6 chân (SIL 6): 24](#_Toc166867017)

[2.2.9 Led 7 đoạn: 24](#_Toc166867018)

[2.2.10 Thạch anh: 24](#_Toc166867019)

[2.2.11 Adapter: 24](#_Toc166867020)

[PHẦN 3: THI CÔNG MẠCH 25](#_Toc166867021)

[3.1 Tạo mạch 25](#_Toc166867022)

[3.1.1 Khối dao động: 25](#_Toc166867023)

[3.1.2 Khối cấp nguồn: 26](#_Toc166867024)

[3.1.3 Khối nạp chương trình: 27](#_Toc166867025)

[3.1.4 Khối điều khiển: 27](#_Toc166867026)

[3.1.5 Khối hiển thị và Khối vi xử lý: 29](#_Toc166867027)

[3.1.1.5 Khối điều khiển: 29](#_Toc166867028)

[3.2 Source code 31](#_Toc166867029)

[3.3 Mạch proteus 34](#_Toc166867030)

[34](#_Toc166867031)

[3.4 Tạo mạch PCB 35](#_Toc166867032)

[3.5 Làm mạch thực tế 35](#_Toc166867033)

[3.5.1 Nguyên liệu 35](#_Toc166867034)

[3.5.2 Các bước làm mạch: 35](#_Toc166867035)

[3.5.3 Mạch thực tế 37](#_Toc166867036)

[PHẦN 4: KẾT LUẬN 39](#_Toc166867037)

[PHỤ LỤC 40](#_Toc166867038)

[BẢNG PHÂN CHIA CÔNG VIỆC 40](#_Toc166867039)

[Tài liệu: 40](#_Toc166867040)

[HẾT. 40](#_Toc166867041)

# PHẦN 1: LỜI MỞ ĐẦU

Chúng em chọn xây dựng mạch đếm thời gian(đồng hồ điện tử) là vì đồng hồ là một yếu tố quan trọng trong việc kiểm soát và quản lý thời gian hiệu quả. Trong hệ thống điều khiển tự động hiện đại, đồng hồ đóng vai trò quan trọng, là “mắt xanh” giúp chúng ta nhìn thấy thời gian một cách rõ ràng và chính xác. Khả năng hiển thị thời gian và thu hút sự chú ý, nó không chỉ là một phần quan trọng của hệ thống điều khiển mà còn là biểu tượng của sự chính xác và sự kiểm soát.

Trong thế giới ngày nay, khi mà yêu cầu về chính xác và hiệu quả ngày càng cao, việc hiểu rõ về mạch đèn thời gian và vai trò của nó là hết sức quan trọng. Chúng em sẽ khám phá những khía cạnh kỹ thuật và ứng dụng thực tế của mạch đèn thời gian, đồng thời đánh giá tầm quan trọng của nó trong việc kiểm soát và quản lý thời gian hiệu quả trong xã hội ngày nay.

Trong trường hợp có bất kỳ sai sót nào xuất hiện trong quá trình tiến hành và triển khai dự án, sự quan trọng của việc nhận được sự hướng dẫn và đánh giá xây dựng từ quý thầy cô là không thể phủ nhận. Việc chấp nhận và sửa lỗi là một phần không thể thiếu của hành trình nghiên cứu và học tập, và chúng em cam kết luôn mở lòng để tiếp nhận mọi ý kiến xây dựng. Điều này không chỉ giúp chúng em hoàn thiện đề tài của mình một cách toàn diện mà còn nâng cao sự hiểu biết của bản thân mình. Chân thành cảm ơn sự hỗ trợ và chỉ dẫn quý báu từ quý thầy cô!

# PHẦN 2 CƠ SỞ LÝ THUYẾT

## 2.1. Linh kiện

Khái quát về linh kiện :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên linh kiện | Giá trị | Số lượng | Note |
| 1 | Tụ 22nF | 22nF 50V | 2 |  |
| 2 | Tụ gốm 100nF | 100nF 50V | 1 |  |
| 3 | Tụ hoá 100uF | 100uF 16V | 1 |  |
| 4 | Trở 10k | 10k 1/4W | 3 |  |
| 5 | Trở 220 | 220W 1/4W | 14 |  |
| 6 | Trở 1k | 1k 1/4W | 4 |  |
| 7 | PIC 16F877A | 16F877A | 1 |  |
| 8 | Đế PIC | 40 Pin | 1 |  |
| 9 | Transistor A1015 | A1015 | 4 |  |
| 10 | Nút nhấn | 6x6x5 | 3 |  |
| 11 | Domino 2 | KF128-2P | 1 | J1 nguồn |
| 12 | Rào đơn 6 chân (SIL 6) | Cái | 1 | J2 nạp code |
| 13 | Led 7 đoạn | 0.8 Inch dương chung | 4 |  |
| 14 | Thạch anh | 20Mhz | 1 |  |
| 15 | Adapter | 5V 2A | 1 |  |

## 2.2. Chức năng của các bộ phận trong mạch

### 2.2.1 PIC 16F877:

### 2.2.1.1 Giới thiệu:

Vi xử lý có rất nhiều loại bắt đầu từ 4 bit cho đến 32 bit, vi xử lý 4 bit hiện nay không còn nhưng vi xử lý 8 bit vẫn còn mặc dù đã có vi xử lý 64 bit

Lý do sự tồn tại của vi xử lý 8 bit là phù hợp với một số yêu cầu điều khiển trong công nghiệp. Các vi xử lý 32 bit, 64 bit thường sử dụng cho các máy tính vì khối lượng dữ liệu của máy tính rất lớn nên cần các vi xử lý càng mạnh càng tốt.

Các hệ thống điều khiển trong công nghiệp sử dụng các vi xử lý 8 bit hay 16 bit như hệ thống điện của xe hơi, hệ thống điều hòa, hệ thống điều khiển các dây chuyền sản xuất, …

A collage of various electronics

Description automatically generated

Hình 1: Các thiết bị sử dụng vi xử lý.

Khi sử dụng vi xử lý thì phải thiết kế một hệ thống gồm có: Vi xử lý, có bộ nhớ, các ngoại vi.

A diagram of a bus

Description automatically generated

Hình 2: Hệ thống vi xử lý.

Bộ nhớ dùng để lưu chương trình cho vi xử lý thực hiện và lưu dữ liệu cần xử lý, các ngoại vi dùng để xuất nhập dữ liệu từ bên ngoài vào xử lý và điều khiển trở lại. Các khối này liên kết với nhau tạo thành một hệ thống vi xử lý.

Yêu cầu điều khiển càng cao thì hệ thống càng phức tạp và nếu yêu cầu điều khiển đơn giản thì hệ thống vi xử lý cũng phải có đầy đủ các khối trên.

Để kết nối các khối trên tạo thành một hệ thống vi xử lý đòi hỏi người thiết kế phải rất hiểu biết về tất cả các thành phần vi xử lý, bộ nhớ, các thiết bị ngoại vi. Hệ thống tạo ra khá phức tạp, chiếm nhiều không gian, mạch in, và vấn đề chính là đòi hỏi người thiết kế hiểu thật rõ về hệ thống. Một lý do nữa là vi xử lý thường xử lý dữ liệu theo byte hoặc word trong khi đó các đối tượng điều khiển trong công nghiệp thường điều khiển theo bit.

Chính vì sự phức tạp nên các nhà chế tạo đã tích hợp bộ nhớ và một số các thiết bị ngoại vi cùng với vi xử lý tạo thành một IC gọi là vi điều khiển – Microcontroller như hình 3

A diagram of a microchip

Description automatically generated

Hình 3: Vi điều khiển được tích hợp từ vi xử lý, bộ nhớ và các ngoại vi.

Khi vi điều khiển ra đời đã mang lại sự tiện lợi là dễ dàng sử dụng trong điều khiển công nghiệp, việc sử dụng vi điều khiển không đòi hỏi người sử dụng phải hiểu biết một lượng kiến thức quá nhiều như người sử dụng vi xử lý.  
 Với một ứng dụng cụ thể thì người thiết kế chỉ cần kết nối vi điều khiển với các thiết bị nhập dữ liệu và các thiết bị xuất dữ liệu để điều khiển. Ví dụ một hệ thống báo chuông giờ học thì cần có vi điều khiển, các thiết bị vào gồm có IC thời gian thực, các nút nhấn để chỉnh thời gian, cài đặt thời gian, các thiết bị ra gồm có màn hìn hiển thị như Led hoặc LCD, có transistor, relay hoặc Triac để điều khiển chuông.

Các thiết bị vào và ra cùng với vi điều khiển được thể hiện tiêu biểu như hình 4.

A diagram of a computer component

Description automatically generated

Hình 4: Các thiết bị vào, ra và vi điều khiển.

Có rất nhiều hãng chế tạo được vi điều khiển, hãng sản xuất nổi tiếng là TI, Microchip, ATMEL, ... tài liệu này sẽ trình bày vi điều khiển tiêu biểu là PIC16F887 của MICROCHIP.

Trong tài liệu này trình bày vi điều khiển PIC16F887, các thông số của vi điều khiển như sau:  
**Đặc điểm thực thi tốc độ cao CPU RISC là:**  
• Có 35 lệnh đơn.  
• Thời gian thực hiện tất cả các lệnh là 1 chu kì máy, ngoại trừ lệnh rẽ nhánh là 2.  
• Tốc độ hoạt động:  
 ▪ Ngõ vào xung clock có tần số 20MHz.

▪ Chu kì lệnh thực hiện lệnh 200ns.

• Có nhiều nguồn ngắt.  
• Có 3 kiểu định địa chỉ trực tiếp, gián tiếp và tức thời.

***Cấu trúc đặc biệt của vi điều khiển***

• Bộ dao động nội chính xác:

▪ Sai số ± 1%  
 ▪ Có thể lựa chọn tần số từ 31 kHz đến 8 Mhz bằng phần mềm.

▪ Cộng hưởng bằng phần mềm.  
▪ Chế độ bắt đầu 2 cấp tốc độ.  
▪ Mạch phát hiện hỏng dao động thạch anh cho các ứng dụng quan trọng.  
▪ Có chuyển mạch nguồn xung clock trong quá trình hoạt động để tiết kiệm  
công suất.

• Có chế độ ngủ để tiết kiệm công suất.  
• Dãy điện áp hoạt động rộng từ 2V đến 5,5V.  
• Tầm nhiệt độ làm việc theo chuẩn công nghiệp.  
• Có mạch reset khi có điện (Power On Reset – POR).  
• Có bộ định thời chờ ổn định điện áp khi mới có điện (Power up Timer – PWRT)  
và bộ định thời chờ dao động hoạt động ổn định khi mới cấp điện (Oscillator Startup Timer – OST).  
• Có mạch tự động reset khi phát hiện nguồn điện cấp bị sụt giảm, cho phép lựa  
chọn bằng phần mềm (Brown out Reset – BOR).  
• Có bộ định thời giám sát (Watchdog Timer – WDT) dùng dao động trong chip cho  
phép bằng phần mềm (có thể định thời lên đến 268 giây).  
• Đa hợp ngõ vào reset với ngõ vào có điện trở kéo lên.  
• Có bảo vệ code đã lập trình.  
• Bộ nhớ Flash cho phép xóa và lập trình 100,000 lần.  
• Bộ nhớ Eeprom cho phép xóa và lập trình 1,000,000 lần và có thể tồn tại trên 40 năm.  
• Cho phép đọc/ghi bộ nhớ chương trình khi mạch hoạt động.  
• Có tích hợp mạch gỡ rối.

***Cấu trúc nguồn công suất thấp***• Chế độ chờ: dòng tiêu tán khoảng 50nA, sử dụng nguồn 2V.  
• Dòng hoạt động:

▪ 11µA ở tần số hoạt động 32kHz, sử dụng nguồn 2V.  
▪ 220µA ở tần số hoạt động 4MHz, sử dụng nguồn 2V.

• Bộ định thời Watchdog Timer khi hoạt động tiêu thụ 1,4µA, điện áp 2V.  
***Cấu trúc ngoại vi***• Có 35 chân I/O cho phép lựa chọn hướng độc lập:  
 ▪ Mỗi ngõ ra có thể nhận/cấp dòng lớn khoảng 25mA nên có thể trực tiếp điều khiển led.

▪ Có các port báo ngắt khi có thay đổi mức logic.  
 ▪ Có các port có điện trở kéo lên bên trong có thể lập trình.

▪ Có ngõ vào báo thức khỏi chế độ công suất cực thấp.

• Có module so sánh tương tự:

▪ Có 2 bộ so sánh điện áp tương tự  
▪ Có module nguồn điện áp tham chiếu có thể lập trình.  
▪ Có nguồn điện áp tham chiếu cố định có giá trị bằng 0,6V.  
▪ Có các ngõ vào và các ngõ ra của bộ so sánh điện áp.  
▪ Có chế độ chốt SR.

• Có bộ chuyển đổi tương tự sang số:

▪ Có 14 bộ chuyển đổi tương tự với độ phân giải 10 bit.

• Có timer0: 8 bit hoạt động định thời/đếm xung ngoại có bộ chia trước có thể lập trình.  
• Có timer1:

▪ 16 bit hoạt động định thời/đếm xung ngoại có bộ chia trước có thể lập trình.  
▪ Có ngõ vào cổng của timer1 để có thể điều khiển timer1 đếm từ tín hiệu bên  
ngoài.  
▪ Có bộ dao động công suất thấp có tần số 32kHz.

• Có timer2: 8 bit hoạt động định thời với thanh ghi chu kỳ, có bộ chia trước và chia sau.  
• Có module capture, compare và điều chế xung PWM+ nâng cao

▪ Có bộ capture 16 bit có thể đếm được xung với độ phân giải cao nhất là  
12,5ns.  
▪ Có bộ điều chế xung PWM với số kênh ngõ ra là 1, 2 hoặc 4, có thể lập trình  
với tần số lớn nhất là 20kHz.  
▪ Có ngõ ra PWM điều khiển lái.

• Có module capture, compare và điều chế xung PWM

▪ Có bộ capture 16 bit có thể đếm được xung với chu kỳ cao nhất là 12,5ns.  
▪ Có bộ so sánh 16 bit có thể so sánh xung đếm với chu kỳ lớn nhất là 200ns  
▪ Có bộ điều chế xung PWM có thể lập trình với tần số lớn nhất là 20kHz.

• Có thể lập trình trên bo ISP thông qua 2 chân.  
• Có module truyền dữ liệu nối tiếp đồng bộ MSSP hổ trợ chuẩn truyền 3 dây SPI, chuẩn I2C ở 2 chế độ chủ và tớ.

A white sheet with black text

Description automatically generated

Hình 5: Trình bày tóm tắt cấu trúc của 5 loại PIC16F88X.

Cấu hình của vi điều khiển được minh họa bằng hình ảnh như hình 6:

A computer chip with many green boxes

Description automatically generated with medium confidence

Hình 6: Cấu hình của vi điều khiển.

### 2.2.1.2 Sơ đồ cấu trúc của vi điều khiển PIC 16F887:

Sơ đồ cấu trúc vi điều khiển được trình bày ở hình 7.  
Các khối bên trong vi điều khiển bao gồm:  
- Có khối thanh ghi định cấu hình cho vi điều khiển.  
- Có khối bộ nhớ chương trình có nhiều dung lượng cho 5 loại khác nhau.  
- Có khối bộ nhớ ngăn xếp 8 cấp (8 level stack).  
- Có khối bộ nhớ Ram cùng với thanh ghi FSR để tính toán tạo địa chỉ cho 2 cách truy  
xuất gián tiếp và trực tiếp.  
- Có thanh ghi lệnh (Instruction register) dùng để lưu mã lệnh nhận về từ bộ nhớchương trình.

- Có thanh ghi bộ đếm chương trình (PC) dùng để quản lý địa chỉ của bộ nhớ chương trình.

A diagram of a machine

Description automatically generated

Hình 7: Cấu trúc bên trong của vi điều khiển.

- Có thanh ghi trạng thái (status register) cho biết trạng thái sau khi tính toán của khối ALU.  
- Có thanh ghi FSR.  
- Có khối ALU cùng với thanh ghi working hay thanh ghi A để xử lý dữ liệu.  
- Có khối giải mã lệnh và điều khiển (Instruction Decode and Control).  
- Có khối dao động nội (Internal Oscillator Block).  
- Có khối dao động kết nối với 2 ngõ vào OSC1 và OSC2 để tạo dao động.  
- Có khối các bộ định thời khi cấp điện PUT, có bộ định thời chờ dao động ổn định, có mạch reset khi có điện, có bộ định thời giám sát watchdog, có mạch reset khi phát hiện sụt giảm nguồn.  
- Có khối bộ dao động cho timer1 có tần số 32kHz kết nối với 2 ngõ vào T1OSI và  
T1OSO.  
- Có khối CCP2 và ECCP.  
- Có khối mạch gỡ rối (In-Circuit Debugger IDC).  
- Có khối timer0 với ngõ vào xung đếm từ bên ngoài là T0CKI.  
- Có khối truyền dữ liệu đồng bộ/bất đồng bộ nâng cao.  
- Có khối truyền dữ liệu đồng bộ MSSP cho SPI và I2C.  
- Có khối bộ nhớ Eeprom 256 byte và thanh ghi quản lý địa chỉ EEADDR và thanh ghi  
dữ liệu EEDATA.  
- Có khối chuyển đổi tín hiệu tương tự sang số ADC.  
- Có khối 2 bộ so sánh với nhiều ngõ vào ra và điện áp tham chiếu.  
- Có khối các port A, B, C, E và D

### 2.2.1.3 Khảo sát sơ đồ chân vi điều khiển PIC16F887:

Sơ đồ chân của vi điều khiển PIC16F887 loại 40 chân được trình bày ở hình 1-7. Vi điều khiển PIC16F887 loại 40 chân, trong đó các chân đều tích hợp nhiều chức năng, chức năng của từng chân được khảo sát theo port.

***a. Chức năng các chân của portA***

• Chân RA0/AN0/ULPWU/C12IN0- (2): có 4 chức năng:

▪ RA0: xuất/ nhập số - bit thứ 0 của port A.  
▪ AN0: ngõ vào tương tự của kênh thứ 0.  
▪ ULPWU (Ultra Low-power Wake up input): ngõ vào đánh thức CPU công suất  
cực thấp.  
▪ C12IN0- (Comparator C1 or C2 negative input): ngõ vào âm thứ 0 của bộ so  
sánh C1 hoặc C2.

• Chân RA1/AN1/C12IN1- (3): có 3 chức năng:

▪ RA1: xuất/nhập số - bit thứ 1 của port A.

▪ AN1: ngõ vào tương tự của kênh thứ 1.

▪ C12IN1- (Comparator C1 or C2 negative input): ngõ vào âm thứ 1 của bộ so sánh C1 hoặc C2.

A diagram of a circuit board

Description automatically generated

Hình 8: Sơ đồ chân của PIC 16F887.

• Chân RA2/AN2/VREF-/CVREF/C2IN+ (4): có 5 chức năng:

▪ RA2: xuất/nhập số - bit thứ 2 của port A.  
▪ AN2: ngõ vào tương tự của kênh thứ 2.  
▪ VREF-: ngõ vào điện áp chuẩn (thấp) của bộ ADC.  
▪ CVREF: điện áp tham chiếu VREF ngõ vào bộ so sánh.  
▪ C2IN+: ngõ vào dương của bộ so sánh C2.

• Chân RA3/AN3/VREF+/C1IN+ (5): có 4 chức năng:

▪ RA3: xuất/nhập số - bit thứ 3 của port A.  
▪ AN3: ngõ vào tương tự kênh thứ 3.  
▪ VREF+: ngõ vào điện áp chuẩn (cao) của bộ A/D.  
▪ C1IN+: ngõ vào dương của bộ so sánh C1.

• Chân RA4/TOCKI/C1OUT (6): có 3 chức năng:

▪ RA4: xuất/nhập số – bit thứ 4 của port A.  
▪ TOCKI: ngõ vào xung clock từ bên ngoài cho Timer0.  
▪ C1OUT: ngõ ra bộ so sánh 1.

• Chân RA5/AN4// C2OUT (7): có 4 chức năng:

▪ RA5: xuất/nhập số – bit thứ 5 của port A.  
▪ AN4: ngõ vào tương tự kênh thứ 4.

▪ : ngõ vào chọn lựa SPI tớ (Slave SPI device).  
▪ C2OUT: ngõ ra bộ so sánh 2.

• Chân RA6/OSC2/CLKOUT (14): có 3 chức năng:

▪ RA6: xuất/nhập số – bit thứ 6 của port A.  
▪ OSC2: ngõ ra dao động thạch anh. Kết nối đến thạch anh hoặc bộ cộng hưởng.  
▪ CLKOUT: ở chế độ RC, ngõ ra của OSC2, bằng ¼ tần số của OSC1.

• Chân RA7/OSC1/CLKIN (13): có 3 chức năng:

▪ RA7: xuất/nhập số – bit thứ 7 của port A.  
▪ OSC1: ngõ vào dao động thạch anh hoặc ngõ vào nguồn xung ở bên ngoài.  
▪ CLKI: ngõ vào nguồn xung bên ngoài.

***b. Chức năng các chân của portB***• Chân RB0/AN12/INT (33): có 3 chức năng:

▪ RB0: xuất/nhập số – bit thứ 0 của port B.  
▪ AN12: ngõ vào tương tự kênh thứ 12.  
▪ INT: ngõ vào nhận tín hiệu ngắt ngoài.

• Chân RB1/AN10/C12IN3- (34): có 3 chức năng:

▪ RB1: xuất/nhập số – bit thứ 1 của port B.  
▪ AN10: ngõ vào tương tự kênh thứ 10.  
▪ C12IN3-: ngõ vào âm thứ 3 của bộ so sánh C1 hoặc C2.

• Chân RB2/AN8 (35): có 2 chức năng:

▪ RB2: xuất/nhập số – bit thứ 2 của port B.  
▪ AN8: ngõ vào tương tự kênh thứ 8.

• Chân RB3/AN9/PGM/C12IN2 (36): có 4 chức năng:

▪ RB3: xuất/nhập số – bit thứ 3 của port B.  
▪ AN9: ngõ vào tương tự kênh thứ 9.  
▪ PGM: Chân cho phép lập trình điện áp thấp ICSP.  
▪ C12IN1-: ngõ vào âm thứ 2 của bộ so sánh C1 hoặc C2

• Chân RB4/AN11 (37): có 2 chức năng:

▪ RB4: xuất/nhập số – bit thứ 4 của port B.  
▪ AN11: ngõ vào tương tự kênh thứ 11.

• Chân RB5/ AN13/(38): có 3 chức năng:

▪ RB5: xuất/nhập số – bit thứ 5 của port B.  
▪ AN13: ngõ vào tương tự kênh thứ 13.  
▪ (Timer1 gate input): ngõ vào Gate cho phép time1 đếm dùng để đếm độ rộng xung.

• Chân RB6/ICSPCLK (39): có 2 chức năng:

▪ RB6: xuất/nhập số.  
▪ ICSPCLK: xung clock lập trình nối tiếp.

• Chân RB7/ICSPDAT (40): có 2 chức năng:

▪ RB7: xuất/nhập số.  
▪ ICSPDAT: ngõ xuất nhập dữ liệu lập trình nối tiếp.

***c. Chức năng các chân của portC***• Chân RC0/T1OSO/T1CKI (15): có 3 chức năng:

▪ RC0: xuất/nhập số – bit thứ 0 của port C.  
▪ T1OSO: ngõ ra của bộ dao động Timer1.  
▪ T1CKI: ngõ vào xung clock từ bên ngoài Timer1.

• Chân RC1/T1OSI/CCP2 (16): có 3 chức năng:

▪ RC1: xuất/nhập số – bit thứ 1 của port C.  
▪ T1OSI: ngõ vào của bộ dao động Timer1.  
▪ CCP2: ngõ vào Capture2, ngõ ra compare2, ngõ ra PWM2.

• Chân RC2 /P1A/CCP1 (17): có 3 chức năng:

▪ RC2: xuất/nhập số – bit thứ 2 của port C.  
▪ P1A: ngõ ra PWM.  
▪ CCP1: ngõ vào Capture1, ngõ ra compare1, ngõ ra PWM1

• Chân RC3/SCK/SCL (18): có 3 chức năng:

▪ RC3: xuất/nhập số – bit thứ 3 của port C.  
▪ SCK: ngõ vào xung clock nối tiếp đồng bộ/ngõ ra của chế độ SPI.  
▪ SCL: ngõ vào xung clock nối tiếp đồng bộ/ngõ ra của chế độ I2C.

• Chân RC4/SDI/SDA (23): có 3 chức năng:

▪ RC4: xuất/nhập số – bit thứ 4 của port C.  
▪ SDI: ngõ vào dữ liệu trong truyền dữ liệu kiểu SPI.  
▪ SDA: xuất/nhập dữ liệu I2C.

• Chân RC5/SDO (24): có 2 chức năng:

▪ RC5: xuất/nhập số – bit thứ 5 của port C.  
▪ SDO: ngõ xuất dữ liệu trong truyền dữ liệu kiểu SPI.

• Chân RC6/TX/CK (25): có 3 chức năng:

▪ RC6: xuất/nhập số – bit thứ 6 của port C.  
▪ TX: ngõ ra phát dữ liệu trong chế độ truyền bất đồng bộ USART.  
▪ CK: ngõ ra cấp xung clock trong chế độ truyền đồng bộ USART.

• Chân RC7/RX/DT (26): có 3 chức năng:

▪ RC7: xuất/nhập số – bit thứ 7 của port C.  
▪ RX: ngõ vào nhận dữ liệu trong chế độ truyền bất đồng bộ EUSART

▪ DT: ngõ phát và nhận dữ liệu ở chế độ truyền đồng bộ EUSART.

***d. Chức năng các chân của portD***• Chân RD0 (19): có 1 chức năng:

▪ RD0: xuất/nhập số – bit thứ 0 của port D.  
• Chân RD1 (20): có 1 chức năng:

▪ RD1: xuất/nhập số – bit thứ 1 của port D.

• Chân RD2 (21): có 1 chức năng:

▪ RD2: xuất/nhập số – bit thứ 2 của port D.  
• Chân RD3 (22): có 1 chức năng:

▪ RD3: xuất/nhập số – bit thứ 3 của port D.  
• Chân RD4 (27): có 1 chức năng:

▪ RD4: xuất/nhập số – bit thứ 4 của port D.

• Chân RD5/ P1B (28): có 2 chức năng:

▪ RD5: xuất/nhập số – bit thứ 5 của port D.  
▪ P1B: ngõ ra PWM.

• Chân RD6/ P1C (29): có 2 chức năng:

▪ RD6: xuất/nhập số – bit thứ 6 của port D.  
▪ P1C: ngõ ra PWM.

• Chân RD7/P1D (30): có 2 chức năng:

▪ RD7: xuất/nhập số – bit thứ 7 của port D.  
▪ P1D: ngõ ra tăng cường CPP1

• Chân RE0/AN5 (8): có 2 chức năng:

▪ RE0: xuất/nhập số.  
▪ AN5: ngõ vào tương tự 5.

• Chân RE1/AN6 (9): có 2 chức năng:

▪ RE1: xuất/nhập số.  
▪ AN6: ngõ vào tương tự kênh thứ 6.

• Chân RE2/AN7 (10): có 2 chức năng:

▪ RE2: xuất/nhập số.  
▪ AN7: ngõ vào tương tự kênh thứ 7.

• Chân RE3/ /VPP (1): có 3 chức năng:

▪ RE3: xuất/nhập số - bit thứ 3 của port E.  
▪ : là ngõ vào reset tích cực mức thấp.  
▪ VPP: ngõ vào nhận điện áp khi ghi dữ liệu vào bộ nhớ nội flash.

• Chân VDD (11), (32):

▪ Nguồn cung cấp dương từ 2V đến 5V.• Chân VSS (12), (31):

▪ Nguồn cung cấp 0V.

***e. Chức năng các chân phân chia theo nhóm chức năng***

• Chức năng là port I/O:

▪ PortA gồm các tín hiệu từ RA0 đến RA7.  
▪ PortB gồm các tín hiệu từ RB0 đến RB7.  
▪ PortC gồm các tín hiệu từ RC0 đến RC7.  
▪ PortD gồm các tín hiệu từ RD0 đến RD7.  
▪ PortE gồm các tín hiệu từ RE0 đến RE3.

• Chức năng tương tự là các ngõ vào bộ chuyển đổi ADC: có 14 kênh

▪ 14 kênh ngõ vào tương tự từ AN0 đến AN13.  
▪ Hai ngõ vào nhận điện áp tham chiếu bên ngoài là Vref+ và Vref-.

• Chức năng tương tự là các ngõ vào bộ so sánh C1 và C2: có 2 bộ so sánh

▪ Có 4 ngõ vào nhận điện áp ngõ vào âm của 2 bộ so sánh là: C12IN0-, C12IN1-, C12IN2-, C12IN3-.  
▪ Có 2 ngõ vào nhận điện áp tương tự dương cho 2 bộ so sánh là: C1IN+ và  
C2IN+.  
▪ Có 2 ngõ ra của 2 bộ so sánh là: C1OUT và C2OUT.  
▪ Có 1 ngõ vào nhận điện áp tham chiếu chuẩn cấp cho 2 bộ so sánh là: CVREF.

• Chức năng dao động cấp xung cho CPU hoạt động:

▪ Có 2 ngõ vào nối với tụ thạch anh để tạo dao động là OSC1 và OSC2.  
▪ Có 1 ngõ vào nhận tín hiệu dao động từ nguồn khác là CLKIN nếu không dùng  
tụ thạch anh, có 1 ngõ ra cấp xung clock cho thiết bị khác là CLKOUT.  
• Chức năng nhận xung ngoại của T0 và T1:  
▪ Có 1 ngõ vào nhận xung ngoại cho timer/counter T0 có tên là T0CKI.  
▪ Có 1 ngõ vào nhận xung ngoại cho timer/counter T1 có tên là T1CKI.  
▪ Có 2 ngõ vào tạo dao động riêng cho Timer1 hoạt động độc lập có tên là T1OSO  
và T1OSI.

• Chức năng truyền dữ liệu SPI:

▪ Có 1 ngõ vào nhận dữ liệu là SDI.  
▪ Có 1 ngõ ra phát dữ liệu là SDO.  
▪ Có 1 ngõ ra phát xung clock là SCK.  
▪ Có 1 ngõ vào chọn chip khi hoạt động ở chế độ tớ là *SS* .

• Chức năng truyền dữ liệu I2C:

▪ Có 1 ngõ truyền/nhận dữ liệu là SDA.  
▪ Có 1 ngõ ra phát xung clock là SCL.

• Chức năng truyền dữ liệu đồng bộ ESUART:

▪ Có 1 ngõ truyền/nhận dữ liệu là DT.  
▪ Có 1 ngõ ra phát xung clock là CK.

• Chức năng truyền dữ liệu không đồng bộ ESUART:

▪ Có 1 ngõ nhận dữ liệu là RX.  
▪ Có 1 ngõ phát dữ liệu là TX.

• Chức năng ngắt:

▪ Có 1 ngõ nhận tín hiệu ngắt cứng là INT.  
• Chức năng CCP (capture, compare, pulse width modulation):

▪ Có 2 tín hiệu cho khối CCP là CCP1 và CCP2.  
▪ Có 4 tín hiệu cho khối PWM là P1A, P1B, P1C, P1D.

• Chức năng nạp chương trình vào bộ nhớ flash:

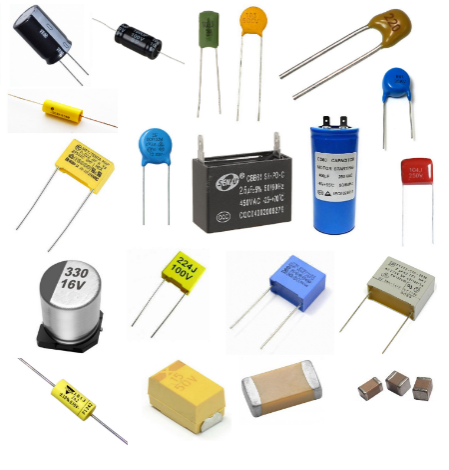
▪ Có 1 tín hiệu để truyền dữ liệu là ICSPDAT.  
▪ Có 1 tín hiệu để nhận xung clock là ICSPCLK.  
▪ Có 1 tín hiệu để điều khiển nạp là PGM.

▪ Có 1 tín hiệu để nhận điện áp lập trình là VPP.

• Có 1 ngõ vào reset có tên là MCLR (master clear).

•Có 4 chân cấp nguồn: VDD cấp nguồn dương, VSS nối với 0V.

### 2.2.3 Tụ điện:

 Bất kể tụ một chiều hay phân cực, khi đấu dây phải nối đúng cực âm-dương. Quy ước cực âm bằng cách sơn một vạch màu sáng dọc theo thân tụ. Tụ điện là một linh kiện thụ động, cấu tạo đơn giản gồm hai bản cực bằng kim loại ghép các nhau một khoảng d ở giữa hai bản tụ là dung dịch hay chất điện môi cách điện có điện dung C. Tụ điện phẳng gồm hai bàn phẳng kim loại diện tích đặt song song và cách nhau khoảng d.

### A row of resistors with different colors Description automatically generated2.2.4 Điện trở:

Là linh kiện thụ động có tác dụng cản trở cả dòng và áp. Điện trở được sử dụng rất nhiều trong các mạch điện tử.

### 2.2.5 Transistor:

[Transistor, còn được gọi là tranzito, là một loại linh kiện bán dẫn chủ động](https://vi.wikipedia.org/wiki/Transistor" \t "_blank)**.** [Nó thường được sử dụng như một phần tử khuếch đại hoặc một khóa điện tử](https://vi.wikipedia.org/wiki/Transistor" \t "_blank). [Transistor nằm trong đơn vị cơ bản tạo thành một cấu trúc mạch ở máy tính điện tử và tất cả các thiết bị điện tử hiện đại khác](https://vi.wikipedia.org/wiki/Transistor" \t "_blank).

### Domino và Connector - Nshop 2.2.6 Domino 2:

[Domino điện hay còn được gọi là cầu đấu dây điện, chức năng của nó là kết nối dây điện đến các thiết bị động lực hoặc thiết bị điều khiển3](https://thuykhidien.com/domino-dien/" \t "_blank). [Domino 2- Nối nguồn 2 chân -Terminal 2 KF128-2P được dùng để kết cuối, nối các dây vào trong mạch](https://robocon.vn/detail/co32-domino-2--noi-nguon-2-chan--terminal-2-kf128-2p.html" \t "_blank).

### nut-nhan-2-chan-trang-12.2.7 Nút nhấn 2 chân:

Là công tắc ngắt bật điện, cầu nối điện hay làm nút nhấn đổi trạng thái.

### Beteiligt Mitschüler Eisbär lenovo battery connector pinout Pfirsich ... 2.2.8 Rào đơn 6 chân (SIL 6):

[Nó được sử dụng trong các mạch điện tử để kết nối các thành phần điện tử lại với nhau](https://suachuatulanh.edu.vn/so-do-chan-ro-le-trung-gian-8-chan-1665243325/" \t "_blank).

### 2.2.9 Led 7 đoạn:

A black and white digital display

Description automatically generated [LED 7 đoạn thường được sử dụng trong các thiết bị hiển thị số như đồng hồ, bảng điện tử, thiết bị đo lường như máy đo nhiệt độ, công suất, áp suất và các ứng dụng điều khiển thời gian, đếm số, hiển thị kết quả, và thông báo lỗi](https://ledhd.vn/tin-tuc/led-7-doan-la-gi.html" \t "_blank).

### A close-up of a transistor Description automatically generated 2.2.10 Thạch anh:

[Thạch anh được sử dụng làm bộ cộng hưởng hiệu suất cao trong các bộ lọc và bộ dao động điện tử](https://dientutuonglai.com/tim-hieu-thach-anh-dien-tu.html" \t "_blank).

### 2.2.11 Adapter:

[Adapter trong điện tử, còn được gọi là bộ chuyển đổi nguồn điện](https://fptshop.com.vn/tin-tuc/danh-gia/adapter-la-gi-khac-nhau-giua-sac-va-adapter-la-gi-144769" \t "_blank). Nó có nhiều chưc năng như chuyển đổi điện áp, hạ áp, chuyển đổi nguồn điện. Nó [là một thiết bị điện tử quan trọng, giúp đảm bảo an toàn và hiệu quả khi sử dụng các thiết bị điện tử](https://fptshop.com.vn/tin-tuc/danh-gia/adapter-la-gi-khac-nhau-giua-sac-va-adapter-la-gi-144769" \t "_blank).

# PHẦN 3: THI CÔNG MẠCH

## 3.1 Tạo mạch

### 3.1.1 Khối dao động:

A diagram of a circuit

Description automatically generated

Hình 9: Khối dao động.

Trong mạch đếm thời gian (counter time), khối dao động sử dụng crystal (thạch anh) đóng vai trò rất quan trọng. [Nó tạo ra tín hiệu xung nhịp với tần số ổn định cao, điều này giúp mạch đếm thời gian hoạt động chính xác](https://www.studocu.com/vn/document/truong-dai-hoc-phenikaa/nguyen-duc-nam/thiet-ke-mach-thach-anh-cho-pic16f887/27095155" \t "_blank).

Mạch đếm thời gian thường sử dụng các xung nhịp từ khối dao động để đếm. Mỗi xung nhịp đại diện cho một đơn vị thời gian cụ thể. [Vì vậy, tần số ổn định của khối dao động rất quan trọng để đảm bảo độ chính xác của mạch đếm](https://monhoc.vn/tai-lieu/giao-trinh-thuc-tap-ky-thuat-so-bai-9-mach-dem-counters-6300/" \t "_blank).

[Ngoài ra, thạch anh trong mạch dao động còn giúp tạo ra tần số được ổn định vì tần số của thạch anh tạo ra rất ít bị ảnh hưởng bởi nhiệt độ hơn là các mạch dao động RC1](https://www.studocu.com/vn/document/truong-dai-hoc-phenikaa/nguyen-duc-nam/thiet-ke-mach-thach-anh-cho-pic16f887/27095155" \t "_blank). [Điều này đặc biệt quan trọng trong các ứng dụng yêu cầu độ chính xác cao, như đồng hồ số](https://hqdt.vn/baiviet/mach-dong-ho-so-don-gian-thach-anh-7490-7447.html" \t "_blank).

### 3.1.2 Khối cấp nguồn:

A diagram of a circuit

Description automatically generated

Hình 10: Khối cấp nguồn.

[Khối cấp nguồn 5V trong mạch điện tử có tác dụng chính là cung cấp nguồn điện áp ổn định 5V cho các thiết bị điện tử](https://memart.vn/tin-tuc/blog/tim-hieu-5vdc-la-gi-va-cach-su-dung-trong-dien-tu-va-vien-thong-vi-cb.html" \t "_blank). [Đây là nguồn điện áp phổ biến nhất được sử dụng để cung cấp năng lượng cho nhiều loại thiết bị như đèn LED, các loại module, vi điều khiển, máy tính nhúng, các thiết bị điện thoại di động, tablet, camera và nhiều thiết bị khác](https://memart.vn/tin-tuc/blog/tim-hieu-5vdc-la-gi-va-cach-su-dung-trong-dien-tu-va-vien-thong-vi-cb.html" \t "_blank). Trong mạch này chúng em sử dụng PIC 16F887A với dòng điện nạp vào từ 2-5.5V nên sẽ hoạt động ổn định khi sử dụng nguồn cấp này.

[Trong mạch cấp nguồn 5V, tụ điện 100uF và 100nF được sử dụng để lọc nhiễu và ổn định nguồn](https://www.se.com/vn/vi/work/solutions/local/capacitor.jsp).

[Tụ điện 100uF (tụ hóa): Đây là loại tụ điện có phân cực, thường được sử dụng trong các bộ lọc cung cấp nguồn](https://www.se.com/vn/vi/work/solutions/local/capacitor.jsp). [Nó giúp lọc nhiễu và ổn định nguồn cung cấp, đặc biệt khi thiếu nguồn pin sạc để cung cấp dòng điện tần số thấp](https://www.se.com/vn/vi/work/solutions/local/capacitor.jsp). [Tụ hóa có khả năng lưu trữ năng lượng điện lớn trên mỗi đơn vị thể tích so với nhiều loại tụ điện khác](https://www.se.com/vn/vi/work/solutions/local/capacitor.jsp).

[Tụ điện 100nF: Đây là loại tụ điện không phân cực, thường được sử dụng để lọc nhiễu tần số cao](https://www.se.com/vn/vi/work/solutions/local/capacitor.jsp). [Tụ điện này giúp loại bỏ các nhiễu tần số cao có thể gây ra bởi các thiết bị điện tử khác trong hệ thống](https://www.se.com/vn/vi/work/solutions/local/capacitor.jsp).

[Cả hai loại tụ điện này đều đóng vai trò quan trọng trong việc đảm bảo nguồn điện ổn định và giảm thiểu nhiễu, giúp mạch cấp nguồn hoạt động hiệu quả và bền bỉ](https://www.se.com/vn/vi/work/solutions/local/capacitor.jsp" \t "_blank).

### 

### 3.1.3 Khối nạp chương trình:

A diagram of a circuit

Description automatically generated

Hình 11: Khối nạp chương trình.

Khối nạp chương trình dùng conn h6 thường được sử dụng để nạp chương trình vào vi điều khiển như PIC 16F887A.

Conn h6 là một loại kết nối, thường được sử dụng để kết nối giữa mạch nạp và vi điều khiển. [Nó giúp truyền tải dữ liệu từ máy tính qua mạch nạp và vào vi điều khiển](https://toilamkythuat.com/lap-trinh-stm32f4-phan-6-nap-chuong-trinh-cho-stm32f4-discovery-board-20051.html" \t "_blank).

[Quá trình nạp chương trình thường bao gồm các bước sau](https://toilamkythuat.com/lap-trinh-stm32f4-phan-6-nap-chuong-trinh-cho-stm32f4-discovery-board-20051.html" \t "_blank):

1. Kết nối mạch nạp với vi điều khiển thông qua conn h6.
2. Sử dụng phần mềm nạp chương trình để mở file chương trình (.hex hoặc .cof) cần nạp.
3. Chọn vi điều khiển cần nạp trong phần mềm.
4. Bắt đầu quá trình nạp chương trình.

### 3.1.4 Khối điều khiển:

Khối điều khiển gồm hai khối:

A diagram of a circuit

Description automatically generatedA diagram of a circuit

Description automatically generated

Hình 12: Khối điều khiển led CA. Hình 13: Khối nút nhấn.

[Khối nút nhấn sử dụng button được sử dụng trong mạch điện tử để điều khiển các chức năng start](https://bachkhoadientu.vn/dien-tu-co-ban-20--su-dung-transistor-trong-thuc-te-.html), pause và reset của mạch.

[Transistor là một linh kiện bán dẫn 3 chân quan trọng trong kỹ thuật điện tử](https://bachkhoadientu.vn/dien-tu-co-ban-20--su-dung-transistor-trong-thuc-te-.html). [Nó hoạt động như một cái van cách ly hay điều chỉnh dòng điện, điện áp trong mạch](https://bachkhoadientu.vn/dien-tu-co-ban-20--su-dung-transistor-trong-thuc-te-.html" \t "_blank).

[Trong khối nút nhấn, transistor được sử dụng để điều khiển trạng thái của mạch khi nút nhấn được nhấn](https://bachkhoadientu.vn/dien-tu-co-ban-20--su-dung-transistor-trong-thuc-te-.html" \t "_blank). [Khi nút nhấn được nhấn, nó tạo ra một tín hiệu điện, tín hiệu này sau đó được dẫn vào cổng điều khiển của transistor](https://bachkhoadientu.vn/dien-tu-co-ban-20--su-dung-transistor-trong-thuc-te-.html" \t "_blank). [Khi transistor nhận được tín hiệu này, nó sẽ thay đổi trạng thái của mình, cho phép dòng điện chảy qua hoặc ngắt dòng điện](https://bachkhoadientu.vn/dien-tu-co-ban-20--su-dung-transistor-trong-thuc-te-.html" \t "_blank).

### 3.1.5 Khối hiển thị và Khối vi xử lý:

A diagram of a circuit board

Description automatically generated

Hình 14: Khối hiển thị led.

### 3.1.1.5 Khối điều khiển:

[Khối hiển thị LED 7 đoạn trong mạch đếm thời gian có chức năng hiển thị thời gian được đếm](https://huynhnhattung.com/dong-ho-timer-delay-dem-thoi-gian-quet-4led-7-doan-lap-trinh-pic/). [Mỗi LED 7 đoạn bao gồm 7 đèn LED được sắp xếp theo hình số '8’](https://huynhnhattung.com/dong-ho-timer-delay-dem-thoi-gian-quet-4led-7-doan-lap-trinh-pic/). [Mỗi đèn LED được gọi là một đoạn, và khi được chiếu sáng, nó tạo thành một phần của chữ số](https://huynhnhattung.com/dong-ho-timer-delay-dem-thoi-gian-quet-4led-7-doan-lap-trinh-pic/).

Khối hiển thị LED 7 đoạn được điều khiển bởi hàm hienthi(). [Hàm này điều khiển từng đèn LED trên mỗi đoạn của LED 7 đoạn để hiển thị giây và phút](https://huynhnhattung.com/dong-ho-timer-delay-dem-thoi-gian-quet-4led-7-doan-lap-trinh-pic/).

void hienthi()

{

output\_d(ma7doan[phut / 10]);

output\_low(pin\_b3);

delay\_ms(1);

output\_high(pin\_b3);

output\_d(ma7doan[phut % 10]);

output\_low(pin\_b4);

delay\_ms(1);

output\_high(pin\_b4);

output\_d(ma7doan[giay / 10]);

output\_low(pin\_b1);

delay\_ms(1);

output\_high(pin\_b1);

output\_d(ma7doan[giay % 10]);

output\_low(pin\_b2);

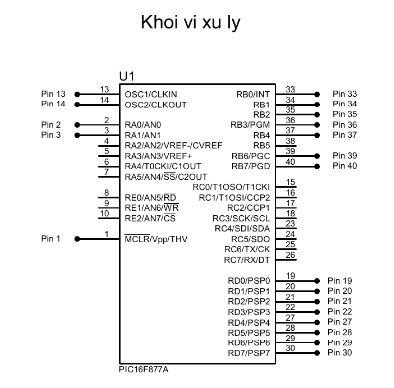
delay\_ms(1);

output\_high(pin\_b2);

}

[Trong đó, ma7doan[] là một mảng chứa mã của mỗi số từ 0 đến 9](https://huynhnhattung.com/dong-ho-timer-delay-dem-thoi-gian-quet-4led-7-doan-lap-trinh-pic/). [Mỗi số được biểu diễn bằng một mã nhất định, và mã này được sử dụng để điều khiển các đèn LED trên mỗi đoạn của LED 7 đoạn](https://huynhnhattung.com/dong-ho-timer-delay-dem-thoi-gian-quet-4led-7-doan-lap-trinh-pic/).

[Cụ thể, output\_d(ma7doan[phut / 10]) và output\_d(ma7doan[phut % 10]) được sử dụng để hiển thị số phút, trong khi output\_d(ma7doan[giay / 10]) và output\_d(ma7doan[giay % 10]) được sử dụng để hiển thị số giây](https://huynhnhattung.com/dong-ho-timer-delay-dem-thoi-gian-quet-4led-7-doan-lap-trinh-pic/).



Hình 15: Khối vi xử lý.

[Vi điều khiển PIC16F887A là một vi điều khiển 8 bit được thiết kế theo công nghệ CPU RSIC](https://mecsu.vn/ho-tro-ky-thuat/vi-dieu-khien-pic16f887.B1B). [Nó có tổng cộng 40 chân và có nhiều gói chân đáp ứng yêu cầu mạch nhỏ gọn và hiện đại](https://mecsu.vn/ho-tro-ky-thuat/vi-dieu-khien-pic16f887.B1B).

Dưới đây là một số thông tin chi tiết về vi điều khiển PIC16F887A:

**[Bộ nhớ](https://mecsu.vn/ho-tro-ky-thuat/vi-dieu-khien-pic16f887.B1B)**[: Vi điều khiển này có 8K x 14 bits/word Flash ROM, 368 x 8 Bytes RAM, và 256 x 8 Bytes EEPROM](https://mecsu.vn/ho-tro-ky-thuat/vi-dieu-khien-pic16f887.B1B)[1](https://mecsu.vn/ho-tro-ky-thuat/vi-dieu-khien-pic16f887.B1B" \t "_blank).

* [Cổng I/O: PIC16F887A có 5 cổng xuất/nhập (A, B, C, D, E) tương ứng với 33 chân ra](https://mecsu.vn/ho-tro-ky-thuat/vi-dieu-khien-pic16f887.B1B).
* [Bộ định thời: Vi điều khiển này có 2 bộ định thời 8 bit (Timer 0 và Timer 2) và 1 bộ định thời 16 bit (Timer 1) có thể hoạt động ở chế độ tiết kiệm năng lượng (SLEEP MODE) với nguồn xung clock ngoài](https://mecsu.vn/ho-tro-ky-thuat/vi-dieu-khien-pic16f887.B1B).
* [Cổng giao tiếp dữ liệu nối tiếp: PIC16F887A hỗ trợ các giao thức dữ liệu hiện đại cho các nhà thiết kế và do có nhiều chân nên hầu hết các giao thức có thể được thực hiện cùng một lúc mà không ảnh hưởng đến các giao thức khác](https://mecsu.vn/ho-tro-ky-thuat/vi-dieu-khien-pic16f887.B1B).
* [Chân ngắt: Vi điều khiển này chỉ có một chân ngắt bên ngoài](https://mecsu.vn/ho-tro-ky-thuat/vi-dieu-khien-pic16f887.B1B).
* [Tần số hoạt động tối đa: 20 MHz](https://mecsu.vn/ho-tro-ky-thuat/vi-dieu-khien-pic16f887.B1B)[1](https://mecsu.vn/ho-tro-ky-thuat/vi-dieu-khien-pic16f887.B1B" \t "_blank).

## 3.2 Source code

Dưới đây là các chú thích cho từng dòng của đoạn mã bạn đã cung cấp:

#include <16f877A.h> // Thư viện cho vi điều khiển PIC16F877A

#Fuses hs // Cấu hình sử dụng dao động ngoại vi (HS - High Speed)

#use delay(clock=20M) // Đặt tần số dao động là 20MHz

#define START\_BUTTON PIN\_A0 // Định nghĩa nút bắt đầu là chân A0

#define RB 0x06 // Định nghĩa RB là 0x06

unsigned int8 ma7doan[] = {0xc0, 0xf9, 0xa4, 0xb0, 0x99, 0x92, 0x82, 0xf8, 0x80, 0x90}; // Mảng chứa mã hiển thị cho LED 7 đoạn

signed int32 ptgiay, giay; // Khai báo biến ptgiay và giay

boolean isCounting = FALSE; // Biến kiểm tra trạng thái đếm

#INT\_EXT // Ngắt ngoại vi

void EXT\_isr(void) // Hàm xử lý ngắt ngoại vi

{

isCounting = FALSE; // Dừng đếm

}

void hienthi() // Hàm hiển thị số lên LED 7 đoạn

{

output\_d(ma7doan[giay / 10]); // Hiển thị chữ số hàng chục của giây

output\_low(pin\_b3); // Kích hoạt chân B3

delay\_ms(2); // Trì hoãn 2ms

output\_high(pin\_b3); // Tắt chân B3

output\_d(ma7doan[giay % 10]); // Hiển thị chữ số hàng đơn vị của giây

output\_low(pin\_b4); // Kích hoạt chân B4

delay\_ms(2); // Trì hoãn 2ms

output\_high(pin\_b4); // Tắt chân B4

output\_d(ma7doan[ptgiay / 10]); // Hiển thị chữ số hàng chục của phần trăm giây

output\_low(pin\_b1); // Kích hoạt chân B1

delay\_ms(2); // Trì hoãn 2ms

output\_high(pin\_b1); // Tắt chân B1

output\_d(ma7doan[ptgiay % 10]); // Hiển thị chữ số hàng đơn vị của phần trăm giây

output\_low(pin\_b2); // Kích hoạt chân B2

delay\_ms(2); // Trì hoãn 2ms

output\_high(pin\_b2); // Tắt chân B2

delay\_ms(2); // Trì hoãn 2ms

}

void nn\_on() // Hàm kiểm tra nút bắt đầu

{

if(input(START\_BUTTON)==0) // Nếu nút bắt đầu được nhấn

{

isCounting = TRUE; // Bắt đầu đếm

}

}

void main() // Hàm chính

{

enable\_interrupts(INT\_EXT); // Kích hoạt ngắt ngoại vi

enable\_interrupts(GLOBAL); // Kích hoạt ngắt toàn cục

set\_tris\_a(0xFF); // Cài đặt PORTA là input

set\_tris\_c(0x00); // Cài đặt PORTC là output

set\_tris\_d(0x00); // Cài đặt PORTD là output

ptgiay = 0; // Khởi tạo ptgiay = 0

giay = 0; // Khởi tạo giay = 0

while (true) // Vòng lặp vô hạn

{

nn\_on(); // Kiểm tra trạng thái của nút bắt đầu

if (isCounting) // Nếu đang đếm, thực hiện đếm và hiển thị

{

if (ptgiay > 99) // Nếu ptgiay > 99

{

giay++; // Tăng giay lên 1

ptgiay = 0; // Đặt ptgiay = 0

if (giay > 99) // Nếu giay > 99

{

giay = 0; // Đặt giay = 0

}

}

hienthi(); // Hiển thị số

ptgiay++; // Tăng ptgiay lên 1

}

else // Nếu không đang đếm

{

hienthi(); // Chỉ hiển thị số

}

}

}

**Định nghĩa và khởi tạo**: Mã nguồn bắt đầu bằng việc định nghĩa vi điều khiển sử dụng (16f877A), loại dao động (hs), tần số hoạt động (20M), và các chân nút nhấn (START\_BUTTON trên PIN\_A0). Mảng ma7doan[] chứa các mã để hiển thị các số từ 0 đến 9 trên LED 7 đoạn. Biến isCounting được sử dụng để kiểm soát việc đếm thời gian.

**Ngắt ngoại vi(hàm pause)**: Hàm EXT\_isr() được gọi khi có một sự kiện ngắt ngoại vi. Trong trường hợp này, nó đặt isCounting thành FALSE, tạm dừng việc đếm thời gian.

**Hiển thị LED 7 đoạn**: Hàm hienthi() được sử dụng để hiển thị thời gian hiện tại trên LED 7 đoạn. Nó điều khiển từng đèn LED trên mỗi đoạn của LED 7 đoạn để hiển thị giây và phút.

**Bắt đầu đếm(hàm start)**: Hàm nn\_on() kiểm tra trạng thái của nút bắt đầu. Khi nút này được nhấn, isCounting được đặt thành TRUE, cho phép mạch bắt đầu đếm thời gian.

**Hàm Reset:** Trong code này không có hàm Reset vì chúng em sử dụng  [(Master Clear) và nó có chức năng reset](https://dientutuonglai.com/gioi-thieu-pic-16f877a.html" \t "_blank).

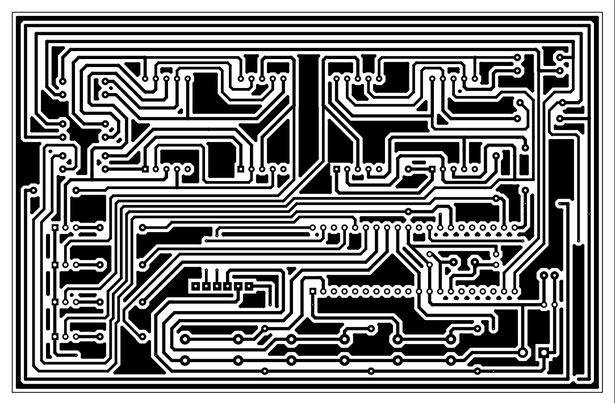
**Hàm main**: Hàm main() là hàm chính của chương trình. Nó khởi tạo các giá trị ban đầu, sau đó vào một vòng lặp vô hạn. Trong vòng lặp này, nếu isCounting là TRUE thì thời gian sẽ được đếm và hiển thị. Ngược lại nếu isCounting là thì chỉ hiển thị thời gian hiện tại mà không đếm

## 3.3 Mạch proteus

## A diagram of electrical wiring Description automatically generated

Hình 16: Mạch đếm thời gian.

## 3.4 Tạo mạch PCB



## 3.5 Làm mạch thực tế

### 3.5.1 Nguyên liệu

Phíp đồng  
 Cuộn phim bóng  
 Dung dịch FeCl3  
 Bùi nhùi sắt  
 Hai thau hoặc hoặc khay để chứa hóa chất rửa mạch (1 nhỏ và 1 lớn)  
 Một máy khoan cầm tay (một lưỡi cắt nhỏ để cắt mạch)  
 Bút lông  
 Aceton (xăng thơm)

### 3.5.2 Các bước làm mạch:

-**Bước 1: Thiết kế bảng mạch in**

+ Sử dụng mẫu PCB ở trên.

**-Bước 2: In bảng mạch từ phần mềm ra giấy.**

+ In bảng mạch đã thiết kế trên phần mềm ra giấy bóng đã chuẩn bị sẵn

**-Bước 3: Cắt tấm phíp đồng theo kích thước phù hợp và vệ sinh.**

+ Đầu tiên sử dụng máy khoan cầm tay(chuyển đầu mài thành đầu cắt tròn). Sau đó, cắt phíp đồng theo kích thước chính xác với bảng mạch muốn tạo ra.

+ Tiếp đến hãy sử dụng bùi nhùi hoặc miếng rửa chén (sử dụng cả nước rửa chén) và chà nhẹ lên bề mặt tấm đồng vừa cắt để làm sạch hết các vết bẩn trên bề mặt.

**-Bước 4: Chuyển mạch in từ tấm phim lên bảng mạch**

+ Đầu tiên hãy đặt tấm giấy bóng đã in đường mạch lên trên phíp đồng, lưu ý mặt có mực in sẽ nằm áp vào miếng đồng.

+ Tiếp đến hãy sử dụng phần giấy không có keo còn lại của quấn đều xung quanh để cố định tấm giấy bóng trên phíp đồng, phần này cần phải kĩ tay hãy quấn thật chặt không chừa các khoảng bọng khí để quá trình ủi (gia nhiệt) để chuyển mục từ giấy bóng sang phóp đồng đạt hiệu quả cao nhất

+ Sau đó, sẽ đến một bước đơn giản hơn đó là điều chỉnh bàn ủi lên mức nhiệt độ cao nhất và ủi vào miếng dán để làm chảy mực từ phim bóng sang phíp đồng. Di nhẹ bàn ủi chậm rãi với một lực vừa phải trong vòng 15 – 20p.

**-Bước 5: Rửa và vệ sinh giấy bóng còn sót lại**

+ Sau quá trình gia nhiệt, không chỉ có mực in mà cả giấy in củng dính chặt vào phíp đồng để loại bỏ giấy thừa hãy ngâm chúng vào một thố nước sạch khoảng 10p để giấy rã ra, dễ dàng vệ sinh hơn (lưu ý sau khi ủi nhiệt độ tấm đồng khá cao, không lấy tay cầm nó nếu không muốn bị phỏng nhé)

+ Khi đã làm sạch toàn bộ phần giấy trên mạch, hãy quan sát lại tổng thể lại mạch đã được in đều trên phíp đồng chưa nếu có nét nào bị đứt (hoặc bị mờ không), nếu có hãy dùng bút lông để vẽ đè lên

**-Bước 6: Ngâm và ăn mòn đồng**

+ Đầu tiên, đun sôi khoảng 500ml nước nóng để bỏ vào FeCl3 để tạo thành dung dịch kiềm.

+  Sau đó cho phần nước sôi còn lại vào khay nhựa lớn và đặt khay nhựa nhỏ vào khay lớn. Nhúng bảng mạch in vào dung dịch ăn mòn trong khoảng 30 phút.

+ Cuối cùng lấy mạch ra xem đồng đã tan hết hay chưa nếu chưa tiếp tục nhúng đồng vào dung dịch ăn mòn cho đến khi đồng tan hết chỉ còn lại đường mạch.

**-Bước 7: Làm sạch mực in trên bề mặt**

+ Khi hoàn thành bước (6) tiến hành vệ sinh mạch dưới nước sạch để làm sạch lớp hóa chất còn sót lại trên mạch. Tiến hành xử lý hóa chất ăn mòn đồng, vì đây là loại chất độc hại nên sử dụng chai nhựa chứa chúng và để ở một góc nào đó chứ không xả ra môi trường.

+ Sau khi mạch đã sạch và xử lý được hóa chất, PCB gần như đã hoàn thành chỉ cần tẩy lớp mực ở trên mặt để lớp dẫn bằng đồng lộ ra là có thể sử dụng. Để loại bỏ lớp mực, hãy sử dụng bông gòn nhúng xăng thơm để tẩy lớp mực in ra khỏi mạch là xong.

**-Bước 8: Khoan lỗ cho mạch PCB**

+ Sử dụng máy khoan cầm tay (dùng để cắt phíp đồng) thay đầu cắt bằng mũi khoan phù hợp để tạo các lỗ khoan cho mạch.

**-Bước 9: Lắp ráp linh kiện cho mạch**

+ Sử dụng các linh kiện ở phần trên tiến hành hành hàn mạch để có một mạch in hoàn chỉnh.

### 3.5.3 Mạch thực tế

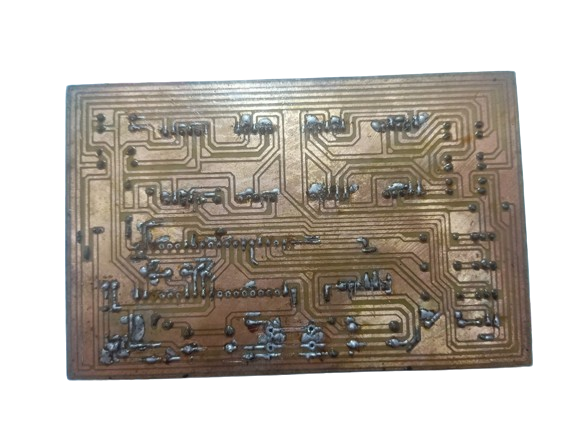
Sau khi hoàn thành công đoạn hàn mạch với các linh kiện, tiến hành kiểm tra xem mạch có hoạt động đúng như thiết kế trên proteus không?

Kết quả mạch hoạt động tốt và ổn định ở nguồn 5V.

A circuit board with numbers and wires

Description automatically generated

Hình 17: Mặt trước



Hình 18: Mặt sau.



Hình 19: Test mạch.

# PHẦN 4: KẾT LUẬN

Trong bài báo cáo này, chúng em đã thảo luận về các chức năng cơ bản, vai trò quan trọng của mạch đếm thời gian(đồng hồ điện tử) trong nhiều công việc khác nhau. Trong suốt quá trình nghiên cứu và phân tích, chúng em nhận thức được sự đa dạng và linh hoạt của nó. Điều này không chỉ nằm ở việc sử dụng nó như một chiếc đông hồ điện tử mà còn là một cơ hội để trao đồi nhiều kỹ năng hữu ích như việc thông thạo sử dụng phần mềm, thiết kế mạch hay công việc hoàn thành và chọn mua nguyên liệu.

Cuối cùng, chúng em tin rằng sự phát triển và ứng dụng tiên tiến của công nghệ này sẽ ngày càng trở thành chìa khóa quan trọng để bảo trì và nâng cao chất lượng cuộc sống hàng ngày. Bằng cách tận dụng công nghệ, chúng ta có thể tạo ra những giải pháp sáng tạo và hiệu quả để giải quyết các vấn đề thực tế. Chúng em cũng muốn nhấn mạnh rằng quá trình học tập không chỉ là việc thu nhận kiến thức, mà còn là việc học cách tư duy, giải quyết vấn đề và làm việc nhóm. Sự chấp nhận và sửa sai không chỉ là một phần quan trọng của quá trình học tập, mà còn là cơ hội để chúng em phát triển kỹ năng giao tiếp, lắng nghe và phản hồi một cách xây dựng.

Trong tương lai, chúng em hy vọng sẽ tiếp tục khám phá và tìm hiểu thêm về các ứng dụng và khả năng của mạch đếm thời gian và các công nghệ liên quan khác. Chúng em cảm ơn sự hỗ trợ và đóng góp từ phía thầy cô và mong muốn tiếp tục nhận được sự hướng dẫn và khích lệ trong quá trình học tập và phát triển. Chúng em tin rằng, với sự kiên trì, sáng tạo và lòng nhiệt huyết, chúng em sẽ tiếp tục đạt được những thành tựu đáng kể trong lĩnh vực này. Chúng em xin chân thành cảm ơn!

# PHỤ LỤC

## BẢNG PHÂN CHIA CÔNG VIỆC

Vị trí A: code nguyên lý đồng hồ + proteuss\_Đăng.

Vị trí B: hoàn thiện phần mềm (thêm button pause và start, vẽ mạch PCB) Việt.

Vị trí C: in mạch và lắp linh kiện Phát.

Vị trí D: làm word + thuyết trình Đông.

Vị trí E: thuyết trình Khang.

Vị trí F: làm slide Bảo.

## Tài liệu:

SACH\_VXL\_PIC\_2016\_08-Nguyễn Đình Phú 08/2016.

https://www.youtube.com/watch?v=iSdOG7WP\_hg

**HẾT.**