

**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP.HỒ CHÍ MINH**

**KHOA ĐIỆN** - **ĐIỆN TỬ**

----------



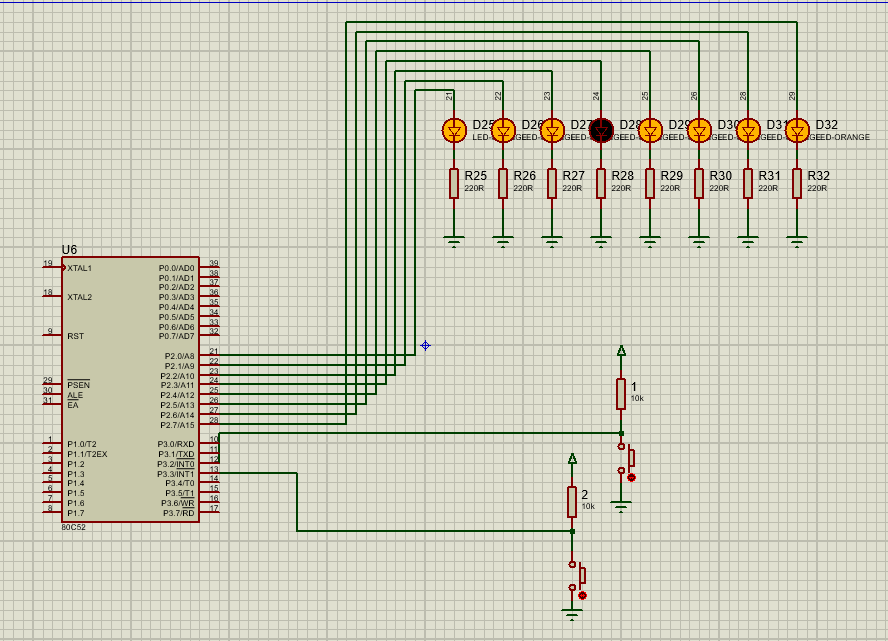
**BÁO CÁO**

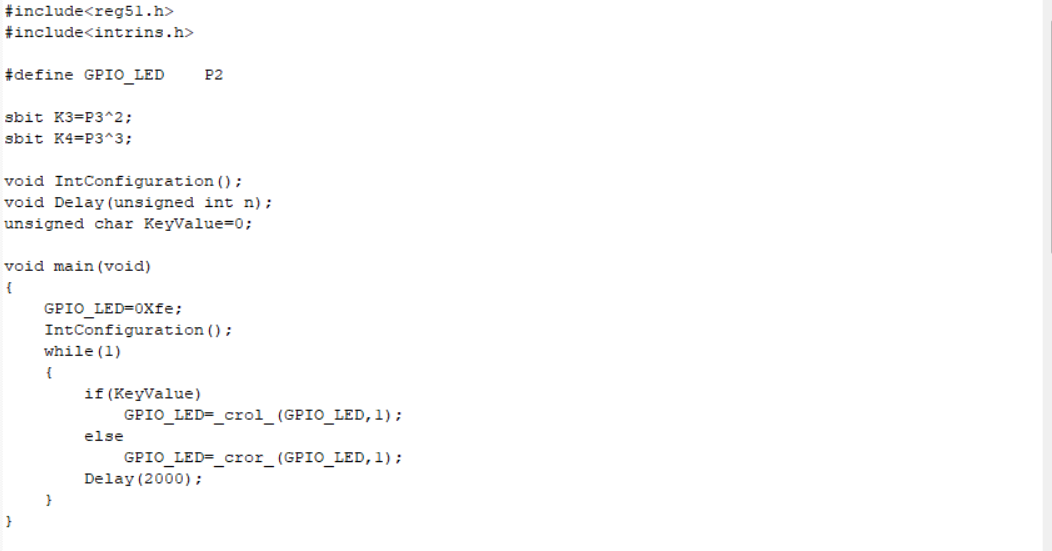
**MÃ MÔN HỌC: TT Kiến trúc máy tính**

**GIÁO VIÊN HƯỚNG DẪN: Huỳnh Hoàng Hà**

TP. Hồ Chí Minh, tháng 11 năm 2023

1. **Button.**







Đoạn mã này là một chương trình vi điều khiển sử dụng vi điều khiển 8051 để điều khiển một đèn LED kết nối với chân P2. Cùng lúc đó, nút nhấn K3 (kết nối với P3.2) và K4 (kết nối với P3.3) được sử dụng để thay đổi trạng thái của LED, khiến nó dịch chuyển sang trái hoặc sang phải.

**Cấu trúc chính của chương trình:**

Khai báo và định nghĩa chân I/O:

Định nghĩa chân GPIO để điều khiển LED là P2.

Xác định hai chân nút nhấn K3 và K4 là P3.2 và P3.3.

**Biến và hàm:**

KeyValue là biến dùng để xác định hướng dịch chuyển của LED (nếu KeyValue = 1 thì LED sẽ dịch phải, ngược lại nếu KeyValue = 0 thì LED dịch trái).

Hàm IntConfiguration() cấu hình các ngắt (interrupts) cho vi điều khiển.

Delay(unsigned int n) là hàm tạo độ trễ thời gian.

**Hàm main:**

Khởi tạo trạng thái ban đầu của LED là 0xFE (1111 1110).

Gọi hàm cấu hình ngắt.

Trong vòng lặp vô hạn:

Nếu KeyValue bằng 1, LED sẽ dịch bit sang trái bằng \_crol\_(GPIO\_LED,1). Ngược lại, LED sẽ dịch sang phải bằng \_cror\_(GPIO\_LED,1).

Tạo một khoảng thời gian độ trễ bằng hàm Delay(2000).

Hàm IntConfiguration():

Cấu hình ngắt INT0 và INT1.

Cho phép các ngắt bằng cách đặt EA=1.

Hàm Delay(unsigned int n):

Tạo một độ trễ thời gian dựa trên một giá trị đầu vào.

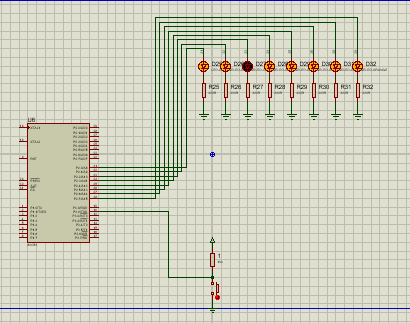
Hàm Int0() và Int1():

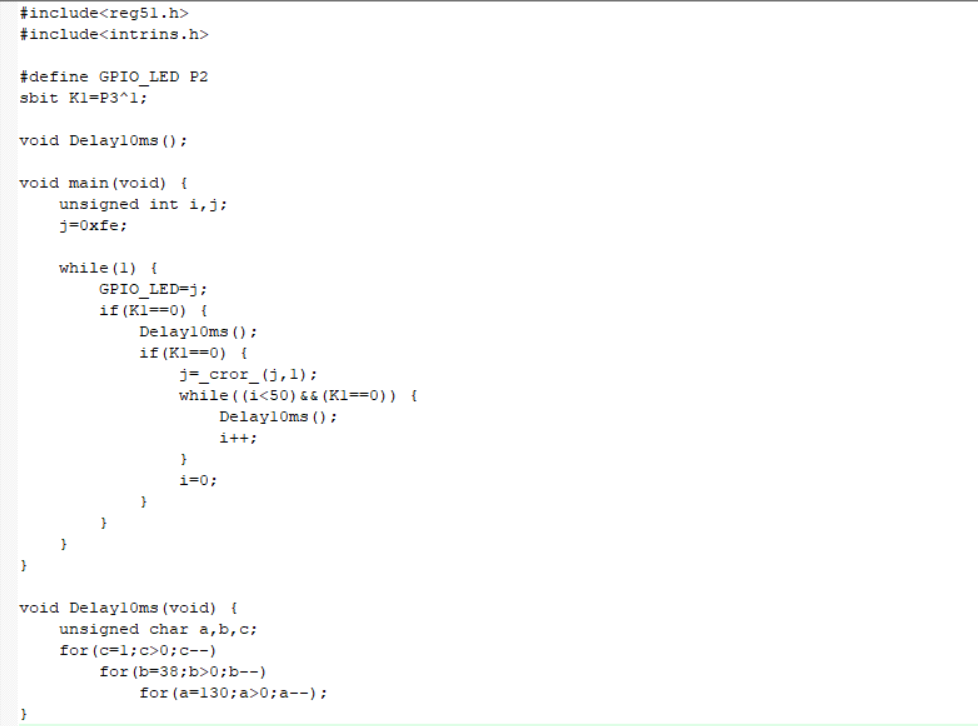
Đây là các hàm xử lý ngắt.

Hàm Int0() xảy ra khi ngắt INT0 được kích hoạt, kiểm tra nút K3 nếu được nhấn (K3 == 0) thì KeyValue được đặt thành 1.

Hàm Int1() xảy ra khi ngắt INT1 được kích hoạt, kiểm tra nút K4 nếu được nhấn (K4 == 0) thì KeyValue được đặt thành 0.

Khi người dùng nhấn nút K3, LED sẽ dịch sang trái, và khi nhấn nút K4, LED sẽ dịch sang phải. Điều này xảy ra bằng cách cài đặt biến KeyValue thông qua các hàm ngắt được gọi khi có sự kiện nhấn nút.





Đoạn code trên là một chương trình vi điều khiển được viết bằng ngôn ngữ C dành cho vi điều khiển 8051. Chương trình này thực hiện việc điều khiển một đèn LED kết nối với chân P2 của vi điều khiển. Ngoài ra, nó sử dụng chân P3.1 (được đặt tên là K1) như một cổng đầu vào để kiểm tra trạng thái nút nhấn.

**Cấu trúc chính của chương trình:**

Khai báo và định nghĩa chân I/O:

Định nghĩa chân GPIO để điều khiển LED là P2.

Xác định chân nút nhấn K1 là P3.1.

Hàm Delay10ms():

Là hàm tạo độ trễ 10ms bằng cách lặp lại một số lần tính toán nhằm đạt được độ trễ mong muốn.

Hàm main():

Khởi tạo biến j với giá trị 0xFE (1111 1110).

Trong vòng lặp vô hạn:

Gán giá trị của biến j vào thanh ghi điều khiển LED (GPIO\_LED=j).

Kiểm tra xem nút nhấn K1 có được nhấn không.

Nếu nút K1 được nhấn:

Tạo độ trễ 10ms (Delay10ms()).

Kiểm tra lại nút K1.

Nếu nút K1 vẫn được nhấn, thực hiện dịch bit của biến j sang phải một vị trí (j=\_cror\_(j,1)).

Tiếp tục tạo độ trễ 10ms và kiểm tra nút K1 cho đến khi không còn nhấn nữa.

Công dụng của chương trình:

Khi nút nhấn K1 được nhấn và giữ, đèn LED sẽ thực hiện một chuỗi thay đổi trạng thái, có thể là nhấp nháy hoặc dịch chuyển vị trí đèn LED tùy thuộc vào cách lập trình cụ thể trong hàm \_cror\_().

Đoạn code này tạo ra một chương trình lặp vô hạn kiểm tra trạng thái của nút nhấn K1 để điều khiển đèn LED theo một cách đặc biệt khi nút được nhấn và giữ.