BÁO CÁO THỰC HÀNH BÀI 1

Môn học: **CHUYÊN ĐỀ THIẾT KẾ HỆ THỐNG NHÚNG 1**- Mã lớp: **CE437.N11**

Giảng viên hướng dẫn thực hành: Phạm Minh Quân

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Thông tin sinh viên** | |  |  | | --- | --- | | MSSV | Họ và tên | | 20520211 | Trương Hữu Khang | | 20520219 | Nguyễn Linh Anh Khoa | | 20520597 | Phan Duy Thông | |
| **Link các tài liệu tham khảo** *(nếu có)* |  |
| **Đánh giá của giảng viên**:  *+ Nhận xét*  *+ Các lỗi trong chương trình*  *+ Gợi ý* |  |

*[Báo cáo chi tiết các thao tác, quy trình sinh viên đã thực hiện trong quá trình làm bài thực hành. Chụp lại hình ảnh màn hình hoặc hình ảnh kết quả chạy trên sản phẩm. Mô tả và giải thích chương trình tương ứng để cho ra kết quả như hình ảnh đã trình bày. Sinh viên xuất ra file .pdf và đặt tên theo cấu trúc: MSSV\_HoTen\_Labx\_Report.pdf (Trong đó: MSSV là mã số sinh viên, HoTen là họ và tên, x trong Labx là chỉ số của bài thực hành tương ứng]*

Mục lục

[Câu 1. Tìm hiểu GPIO 3](#_Toc119858126)

[Câu 2. Viết chương trình để thực hiện điều khiển GPIO: 4](#_Toc119858127)

1. Tìm hiểu GPIO

Tìm hiểu GPIO:

GPIO — General Purpose Input/Output là chức năng ngoại vi cơ bản của vi điều khiển, cho phép nguời dùng lập trình các tín hiệu đầu vào và đầu ra.

Các khái niệm cần biết:

* Pin: Các chân tín hiệu Input/Output.
* Port: Mỗi khối GPIO trong vi điều khiển có thể điều khiển tối đa 1 số lượng Pin nhất định, gọi là Port( cổng).

Các vi điều khiển 32bit có đến 16 chân IO trên 1 port

Có 2 khối điều khiển chính của mỗi GPIO (2 khối được vẽ đứt trong hình), đó chính là :

* Input driver
* Output driver

GPIO bao gồm 8 chức năng chính sau đây:

* Input floating: Có cấu hình chân I/O với chức năng làm ngõ vào và để nổi.
* Input pull-up: Có cấu hình chân I/O với nhiệm vụ làm ngõ vào và trở kéo lên nguồn.
* Input-pull-down: Có cấu hình tương tự như hai chế độ trên, với nhiệm vụ làm ngõ vào và trở kéo xuống nguồn GND.
* Analog: Là Analog cấu hình chân I/O được áp dụng cho các mode có ứng dụng ADC hoặc DAC.
* Output open - drain with pull - up or pull - down capability: Có cấu hình chân I/O với vai trò làm ngõ ra có hai trường hợp xảy ra:
  + Trường hợp 1: Khi output control = 0 thì N-MOS sẽ dẫn và chân I/O sẽ nối với VSS.
  + Trường hợp 2: Khi output control = 1 thì P-MOS và N-MOS không dẫn và chân I/O được để nổi.
* Output push - pull with pull - up or pull - down capability: Có cấu hình chân I/O với vai trò là ngõ ra có hai trường hợp xảy ra:
* Trường hợp 1: Khi output control = O thì N-MOS sẽ dẫn và chân I/O sẽ nối với VSS tương tự như trường hợp đầu tiên của chế độ Output Open – Drain.
* Trường hợp 2: Khi output control = 1 thì P-MOS dẫn và chân I/O được nối với VDD.
* Alternate function push - pull with pull - up or pull - down capability: Có cấu hình I/O tích hợp vừa làm ngõ ra và ngõ vào. Chế độ này sẽ không có chức năng kéo lên và kéo xuống như ở Input. Tuy nhiên, Alternate function push-pull có chức năng output giống với Output push-pull. Bên cạnh đó, chế độ này còn được sử dụng cho chức năng khác là remap.
* Alternate function open - drain with pull - up or pull - down capability: Có cấu hình I/O tích hợp, chức năng Input và ứng dụng trong chức năng remap giống với chế độ Alternate function push-pull. Nhưng khác là chế độ này lại có chức năng output giống với Output open - drain.

Mặc định khi lập trình viên không cấu hình gì, trạng thái của các chân I/O sẽ là Input Floating.

1. Viết chương trình để thực hiện điều khiển GPIO:

**Diagram

Description automatically generated**4 User LEDs trên KIT Open405R-C Package A, STM32F4 Development Board được kết nối sẵn với các chân PB0, PB1, PB2, PB3 của vi điều khiển.

Cài đặt cấu hình tại thẻ Pinout & Configuration

Tại GPIO

Table

Description automatically generated

Tại Timers

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application

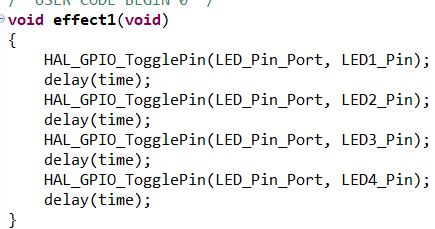
Description automatically generated

Graphical user interface

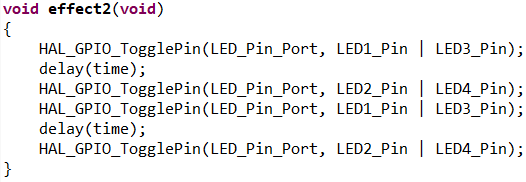
Description automatically generated

Khai báo các hàm

Tại hàm effect1 chớp lần lượt cái đèn



Hàm effect2 chớp xen kẽ 2 đèn



Hàm swap dùng để swap con trỏ lệnh của 2 hiệu ứng

Hàm delay \_HAL\_TIM\_SetCounter để reset giá trị của counter và sử dụng vòng while để tạo delay

Text, letter

Description automatically generated

Tác vụ 1: quản lí hiệu ứng của LED.

Khi nhận được cờ hiệu swapped do tác vụ 2 phất thì sẽ tiến hành tắt hết tất cả đèn để reset hiệu ứng.

:Graphical user interface, text, application

Description automatically generated

Tác vụ 2: dùng để quản lí việc tăng giảm thời gian delay của hiệu ứng và việc thay đổi hiệu ứng.

Ở tác vụ 2 ta thực hiện đọc các KEY. Ta sử dụng hàm HAL\_GPIO\_ReadPin để đọc các KEY..

Nếu nhấn KEY0 thì sẽ giảm thời gian và đèn sẽ chớp nhanh hơn

Nếu nhấn KEY1 thì sẽ tăng thời gian và đèn sẽ chớp chậm hơn

Nếu nhấn cả hai KEY0 và KEY1 thì sẽ đổi hiệu ứng.

Graphical user interface, text, application

Description automatically generated

Cách thức hoạt động: Hàm thực hiện hiệu ứng sẽ được thực hiện thông qua một con trỏ hàm. Ta sử dụng một lệnh swap để có thể thay đổi giá trị của con trỏ lệnh.

Link video demo

<https://drive.google.com/file/d/1mzhK87Vm95_hd3gV5wOq5qDRpTMhbOlo/view?usp=share_link>