

Đề Tài Ứng Dụng

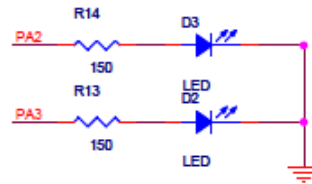
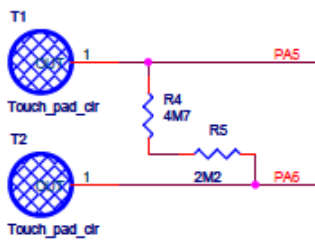
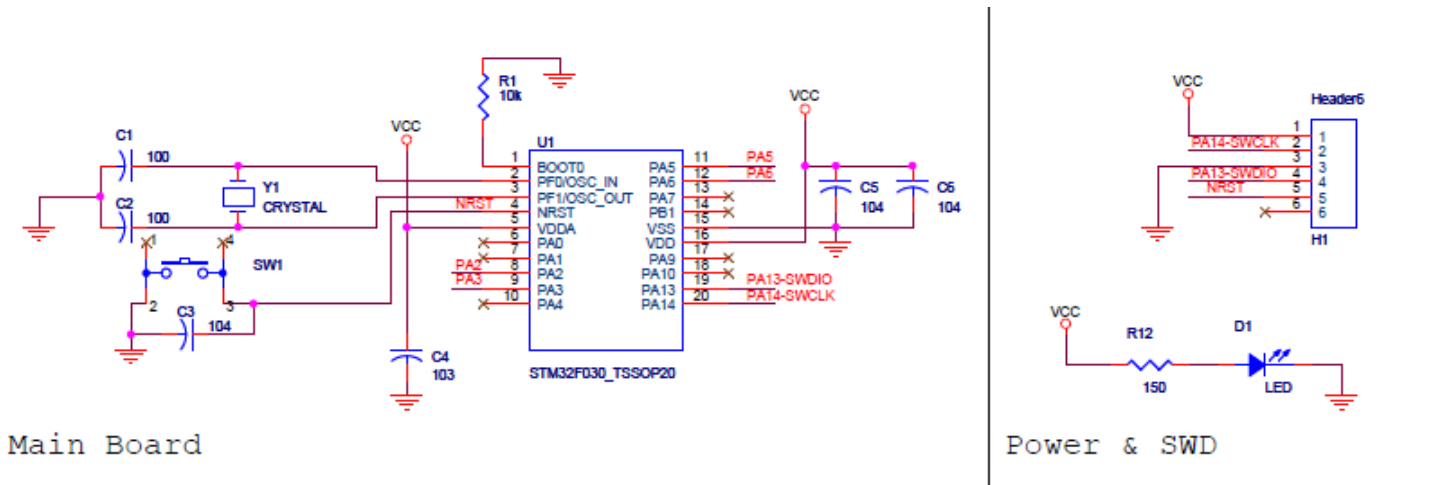
---

TOUCH PAD SENSOR

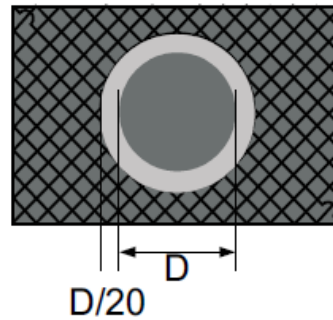
\_Hoàng Anh Hiệp\_

## 1. Thiết bị phần cứng:

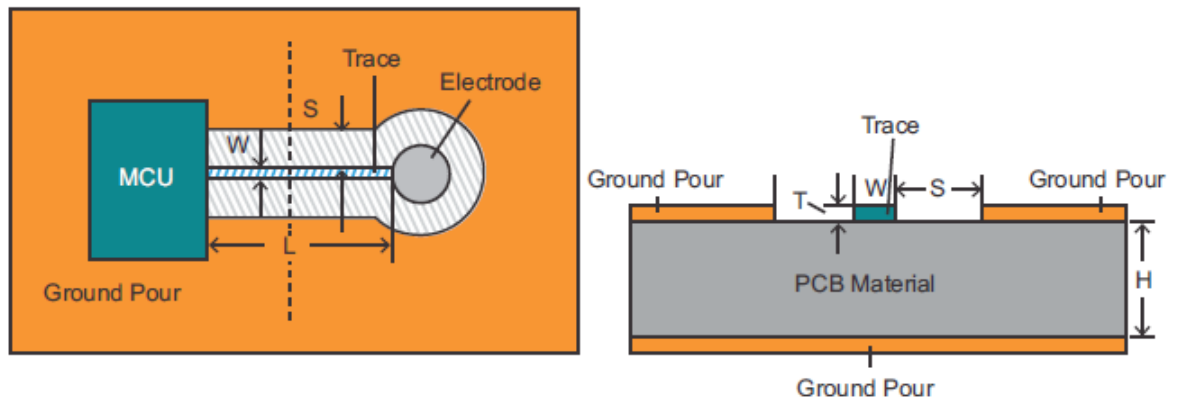
- MCU STM32F030F4P6
  - Sơ đồ mạch nguyên lý:



Touch Pad Sensor



Phần cách ly giữa pad và lớp gnd chọn ở tầm  $= 1/20$  đường kính pad

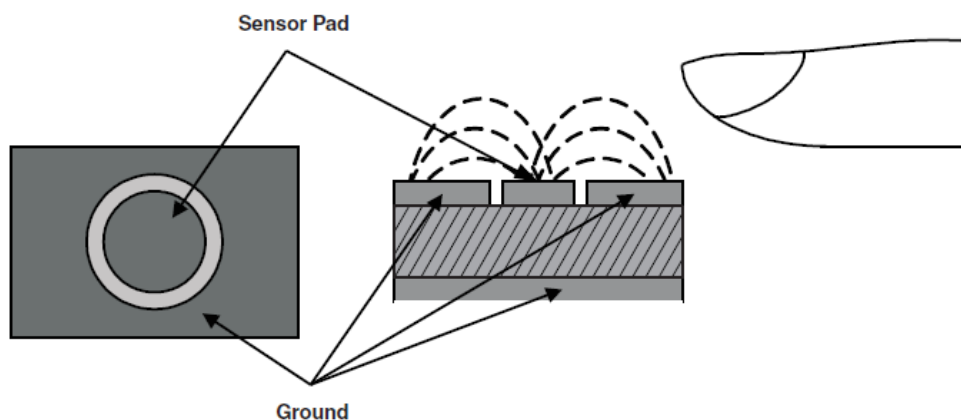


**Figure 6. Top and Cross-Sectional Views of Trace Line in PCB**

Mặt dưới của Pad nên cho phủ lớp GND để tránh nhiễu từ mặt dưới của pad.

## **2. Nguyên lý hoạt động:**

- Pad đồng được bao quanh bởi lớp mass, sẽ hình thành một điện trường, và tạo nên một tụ điện nhỏ, khi tay chạm vào mặt pad đồng sẽ làm gián đoạn lớp điện trường đó và làm tăng giá trị của tụ lên. Từ nguyên lý trên ta sẽ đo giá trị điện dung của tụ để suy ra được pad đồng có được chạm hay không.



**Figure 1. Open Capacitor Acting as a Sensor**

- Để đo được điện dung của tụ, ta sử dụng phương pháp cơ bản nhất là đo thời gian nạp xả tụ thông qua mạch RC.
- Các bước thực hiện để đo chu kỳ nạp xả tụ như sau:
  - Một đầu của R được nối vào touch pad ,đầu còn lại được nối gnd, touch pad được nối vào chân I/O của MCU.
  - Chân I/O được set lên output mức cao, nó sẽ nạp cho tụ của touch pad đến vị trí Vcc rất nhanh.
  - Tiếp theo đó set chân I/O thành input với interrupt cạnh xuống, đồng thời cho chạy 1 free timer để lấy thời gian bắt đầu. Ngay lúc đó tụ sẽ xả thông qua trở R xuống gnd.

- Khi điện áp trên tụ xuống dưới mức  $V_{il}$ , interrupt sẽ được set, bắt đầu đọc giá trị của timer lần nữa và tính toán thời gian xả tụ.
- Tiếp tục thực hiện lại các thao tác trên.

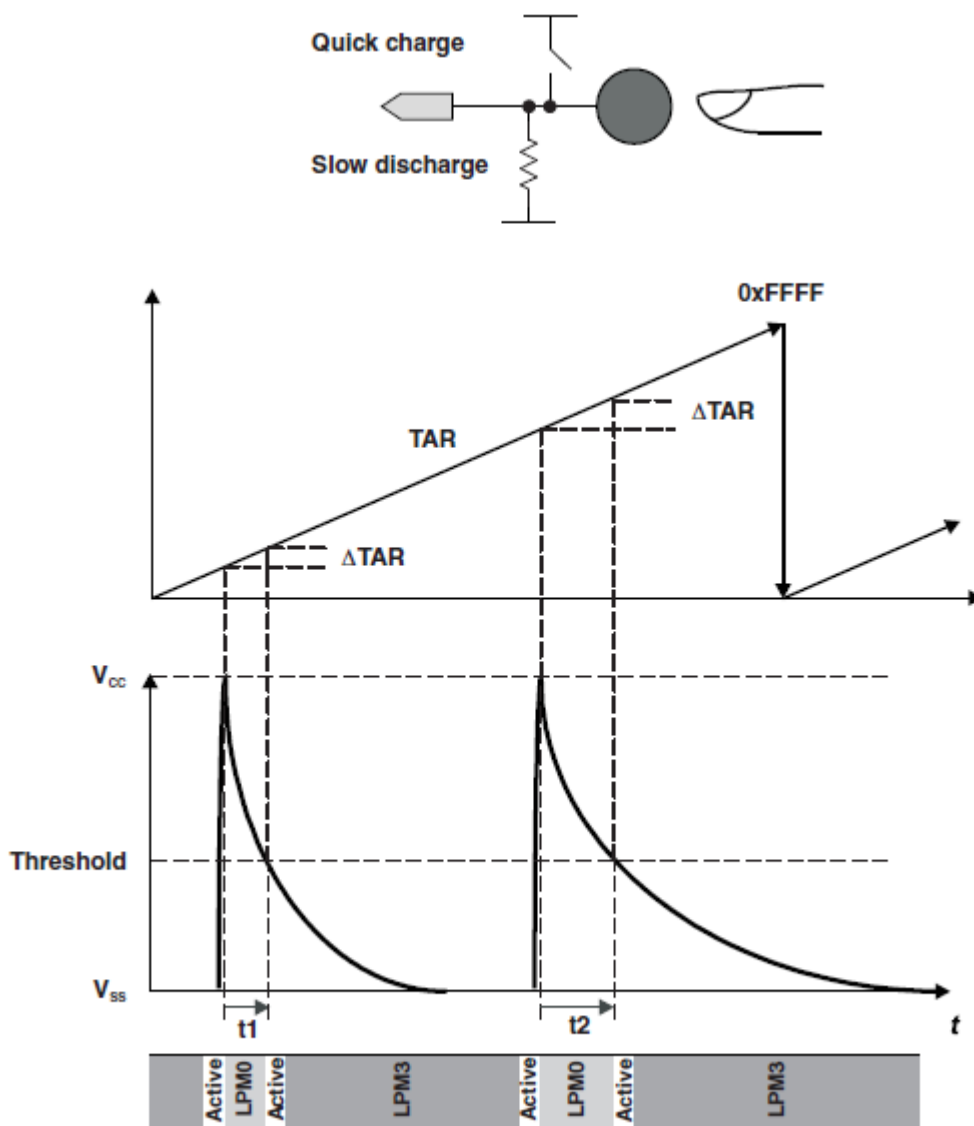
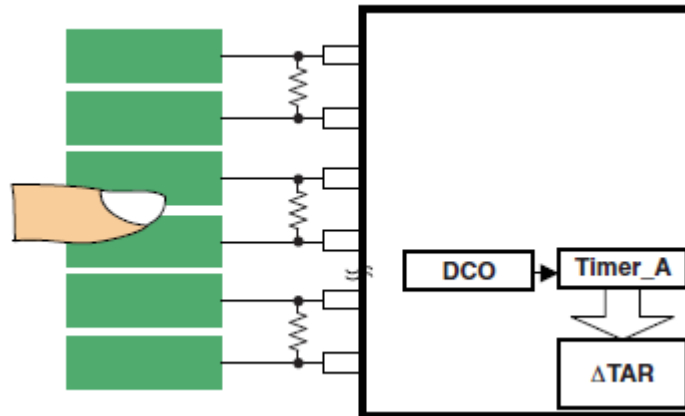


Figure 2. Charge-Discharge Sequence

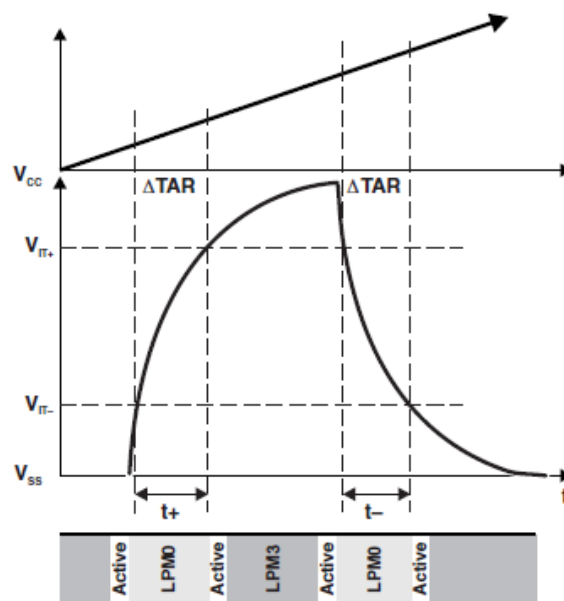
- Thời gian xả tỷ lệ thuận với giá trị của tụ. Nhưng vì giá trị của C thường rất nhỏ ( $<10\text{pF}$ ) nên người ta sử dụng điện trở R lớn để hạn dòng xả của tụ. Chọn R thường lớn hơn 6M.

### 3. Phương pháp dùng để loại bỏ nhiễu và tăng khả năng cảm ứng:

- Đo vi sai thời gian nạp xả tụ:

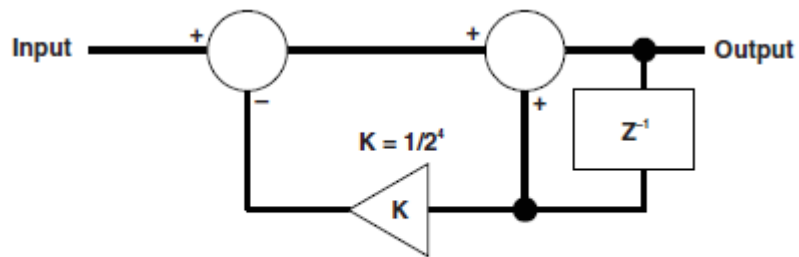


- Phương pháp này giúp giảm số lượng R cần cho mạch xuống 1 nửa và giảm được sự phức tạp khi thiết kế mạch.
- Điện trở được đặt giữa 2 chân của 2 touch pad nối vào I/O của MCU ( PA5 và PA6). Phương pháp đo như sau: khi PA5 bắt đầu được đặt input interrupt thì PA6 sẽ được đặt output, đầu tiên ghi lại giá trị timer và sau đó cho PA6 lên mức cao, khi PA5 interrupt thì tính ra giá trị thời gian nạp tụ; sau đó cho PA6 xuống mức thấp đồng thời ghi lại giá trị timer, khi PA5 có interrupt mức thấp thì tính ra giá trị xả tụ. Quá trình được đảo lại khi chuyển sang đo touch pad cho PA6.
- Phương pháp này kéo dài thời gian nạp lần xả tụ, do khi nạp lần xả đều phải qua điện trở R lớn.



#### b) Lọc tín hiệu bằng bộ lọc thông thấp:

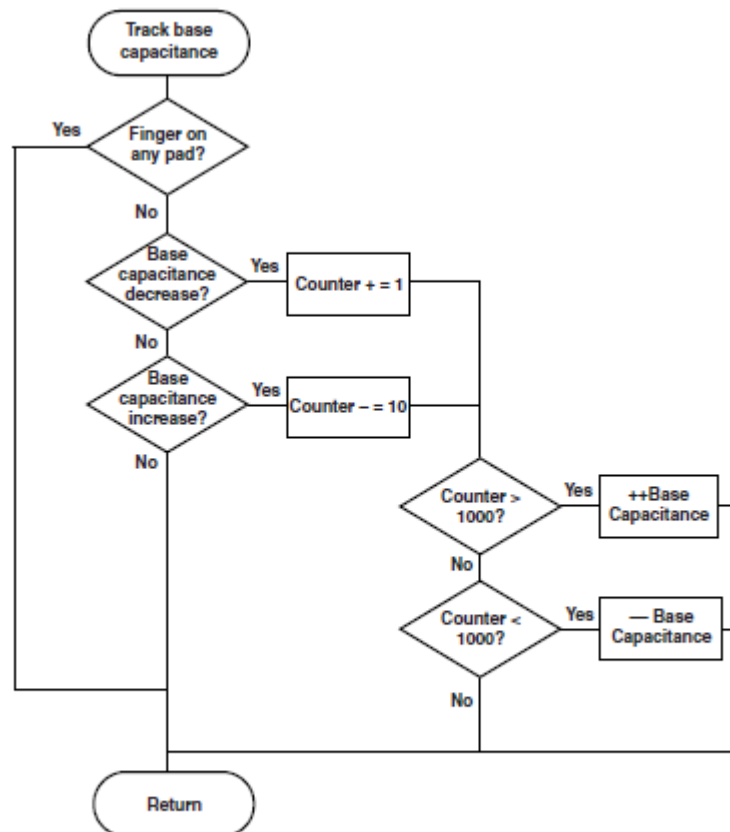
- Thời gian nạp xả tụ đo được từ bước trên cho qua bộ lọc IIR để giảm bớt sự nhấp nhô tín hiệu, làm tăng tính cảm ứng của touch pad.



**Figure 5. Software Low-Pass Filter**

c) Track theo giá trị điện dung khi hở:

- Điện dung của tụ có thể thay đổi theo môi trường, nhiệt độ độ ẩm... vì vậy cần phải bám theo giá trị của tụ khi không chạm tay vào touch pad để có được tín hiệu reference cho các bộ lọc.
- Phương pháp được thực hiện như hình dưới, quá trình track sẽ ngưng khi có tay chạm vào touch pad.

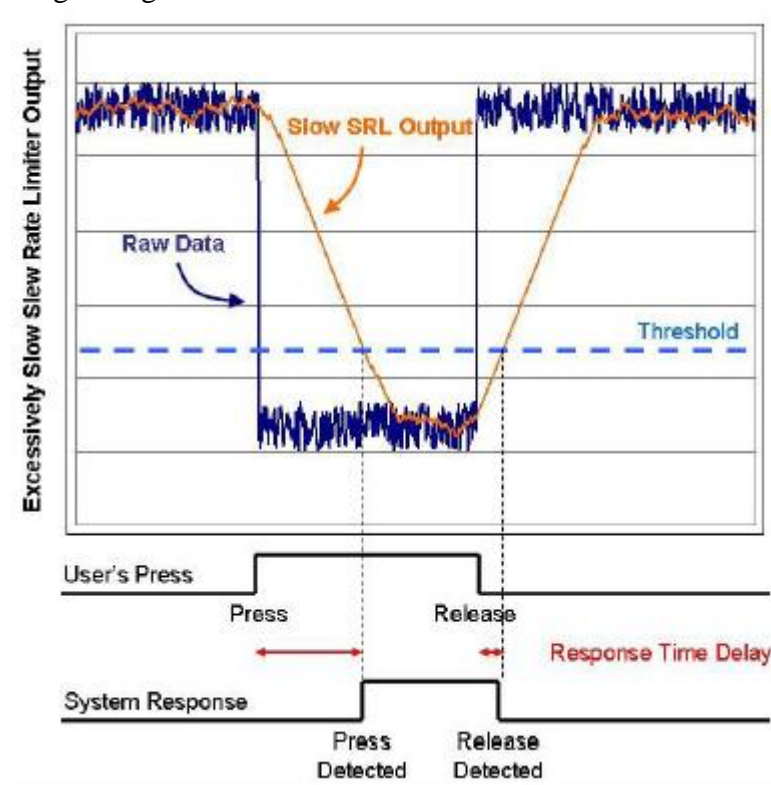


**Figure 8. Tracking Base Capacitance Flow Chart**

d) Bộ lọc slew rate limiter

- Đáp ứng ngõ ra sau khi qua các bước lọc trên vẫn có thể còn nhấp nhô bất chợt, làm giảm độ tin cậy của tín hiệu ngõ ra.
- Giải pháp là sử dụng 1 mức ngưỡng để định mức cho ngõ ra, và 1 biến đếm. Mỗi lần đo được tín hiệu lớn hơn mức 2 lần tụ base ở bước trên thì biến đếm sẽ cộng 1. Khi đếm liên tục đến 1 mức

ngưỡng ( ví dụ 1000 lần) nào đó thì ngõ ra sẽ lên 1, đây sẽ là tín hiệu ngõ ra tuyệt đối. Tương tự như vậy, khi thả tay ra khỏi touch pad, mức tín hiệu đo được sẽ nhỏ hơn mức 2 lần tụ base và biến đếm sẽ giảm, khi xuống dưới mức ngưỡng sẽ cho tín hiệu ngõ ra bằng không.



#### 4. Thiết kế phần mềm điều khiển:

a) Các hàm đã viết trên Keil uVision 5 sử dụng thư viện STM32 Peripheral Driver:

- File IO\_init.c:
  - Hàm led\_init() tạo GPIO cho 2 led trên board.
  - Hàm PadL\_init() tạo GPIO input và interrupt cho pad bên trái đồng thời tạo GPIO cho pad bên phải.
  - Hàm PadR\_init() tạo GPIO input và interrupt cho pad bên phải đồng thời tạo GPIO cho pad bên trái.
  - Hàm TIM\_Config() init TIM3 cho chạy chế độ free timer để đo thời gian xả tụ.
  - Hàm TIM14\_Config() init TIM14 để tạo sample time cho việc đo nhều touch pad.
- File Function.c:
  - Hàm pad\_sense(uint16\_t in,uint16\_t out) thực hiện việc đọc input chân này và xuất output chân kia khi đúng thời điểm.
  - Hàm Cap\_track() thực hiện việc track theo giá trị cơ sở của tụ.
  - Hàm slew\_rate\_limit() bộ lọc slew rate limit cho ngõ ra sau cùng.

- Hàm `sampletime()` là thời gian lấy mẫu để thực hiện sense từng pad
- Hàm `EXTI4_15_IRQHandler()` là hàm interrupt cho PA5 và PA6, tính toán giá trị xấp xỉ của tụ, cho ra giá trị tạm, sau đó tính luôn giá trị ngõ ra sau khi qua lọc thông thấp, đồng thời cho chạy bộ lọc slew rate limiter.