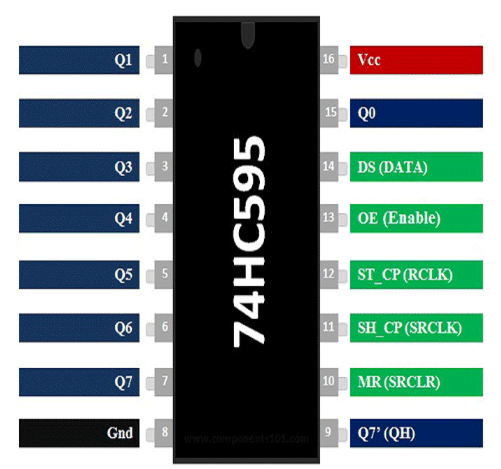
BÁO CÁO LED CUBE 8x8x8

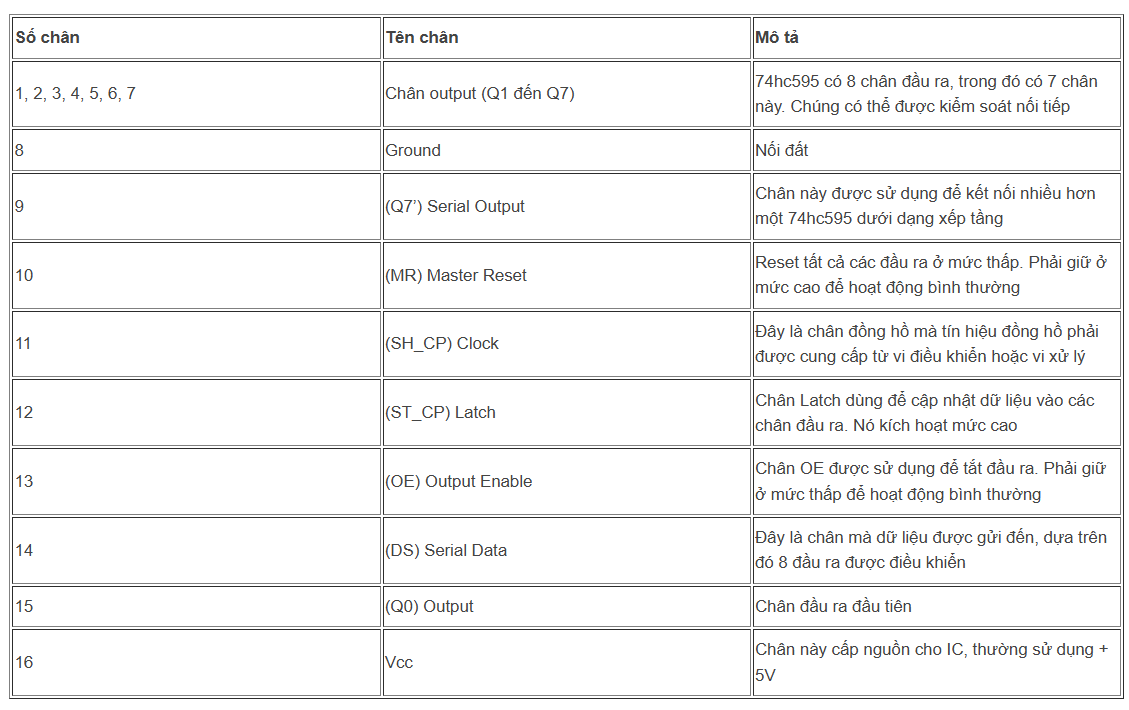
Để điều khiển tất cả đèn LED có trong khối LED, chúng ta chia khối này thành nhiều lớp (Layer) và cột. Khối led có cathode chung. Bằng cách điều khiển các lớp và các cột này, chúng ta có thể kiểm soát từng LED trong khối. Ta cần điều khiển 72 chân gồm 64 anode và 8 cathode.

Để điều khiển nhiều hàng LED (mỗi hàng gồm 8 LED) bằng IC 74HC595 kết hợp với transistor BJT NPN, ta thiết kế mạch theo kiểu quét hàng (row scanning) nhanh liên tục tạo thành hiệu ứng hiển thị toàn bộ nhờ hiệu ứng lưu ảnh của mắt người:

* 8 IC 74HC595 dùng để điều khiển 64 cột.
* 1 IC 74HC595 cùng các BJT NPN 2SC2383 dùng để điều khiển các hàng (bật/tắt từng hàng LED một cách tuần tự).

Sử dụng IC 74HC595 vì là IC xuất được 8 bit điều khiển ra, có thể nối tiếp nhiều con 595 để mở rộng mà chỉ tốn 3 chân GPIO cho cả khối 512 LED.







Chân Clock

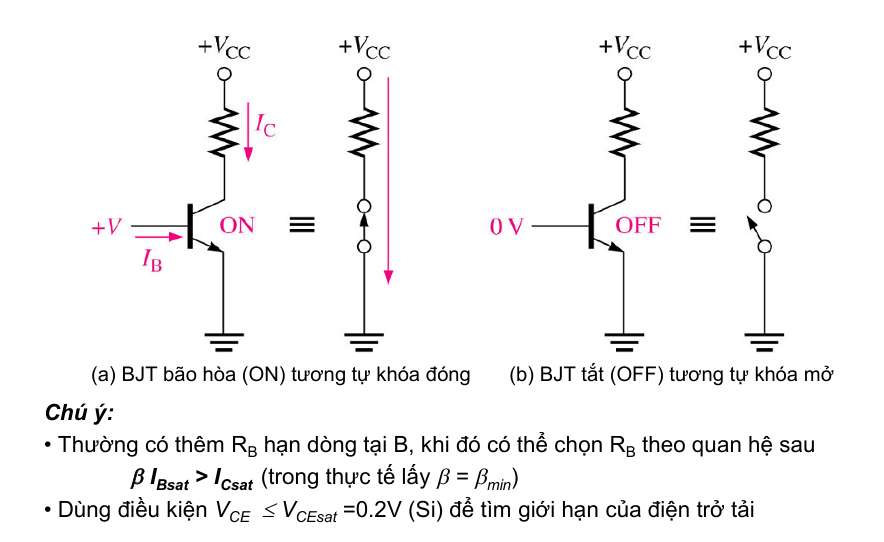
Chân 11: SH\_CP: Clock

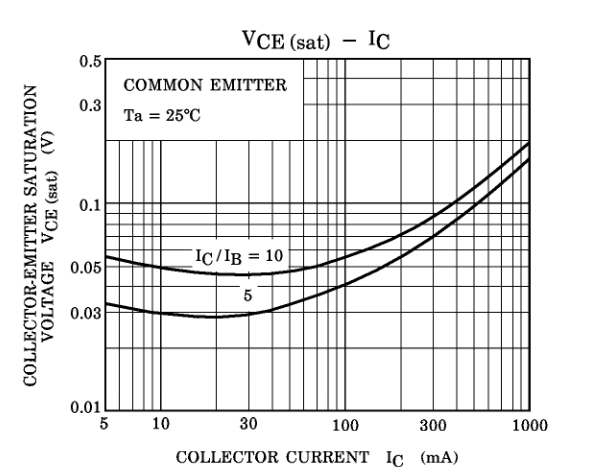
Đây là chân đầu vào xung nhịp của thanh ghi dịch 74HC595. Dữ liệu được chuyển từ chân đầu vào nối tiếp sang thanh ghi dịch 8-bit ở mỗi xung cạnh lên của xung nhịp.

Chân 12:ST\_CP: Latch

Đây là chân đầu vào xung nhịp tích cực mức cao của một thanh ghi lưu trữ. Sự chuyển đổi trạng thái tích cực của tín hiệu tại chân này được sử dụng để đưa dữ liệu vào các chân đầu ra.

IC 74HC595 không thể cấp dòng lớn, dòng tối đa trên mỗi chân output chỉ 20mA, không đủ để điều khiển cùng lúc 64 Led nên ta cần BJT để khuếch đại dòng. Ta chọn BJT NPN 2SC2383





Chọn BJT 2SC2383

🡪 Chọn BJT Q1 là 2SC2383 1A có β=10

Tuy nhiên, giới hạn dòng của GPIO là 20mA nên không cấp đủ, cần thêm BJT nhỏ Q2 là BC547.

GPIO cấp 3,3V 🡪 🡪 Chọn

🡪 Chọn

Dòng tối đa trên mỗi chân output của IC 74HC595 chỉ 20mA điều khiển 1 cột 8 LED

🡪 Chọn

**Cách hoạt động:**

Kéo chân Latch xuống thấp

STM32 gửi 8 byte dữ liệu qua SPI đến 8 IC 74HC595

Kéo Latch lên cao → LED cột cập nhật trạng thái

Bật 2SC2383 tầng mong muốn → tầng đó sáng

Giữ vài ms, đổi tầng → quét tầng tiếp theo

Lặp lại nhanh → thấy sáng toàn khối 3D

**Để bật 1 LED duy nhất, ta cần:**

1. Gửi dữ liệu tầng (64 bit = 8 byte) qua SPI → 74HC595
2. Bật transistor tầng z tương ứng
3. Giữ tầng đó một thời gian ngắn (1–2ms)

**Để bật 1 tổ hợp LED, vì chỉ 1 tầng có thể sáng tại 1 thời điểm → ta quét tầng liên tục:**

Gửi dữ liệu tầng hiện tại

Bật tầng hiện tại

Chuyển tầng tiếp theo

Timer interrupt: là một dạng ngắt được tạo ra khi một bộ đếm thời gian (timer) trong vi điều khiển đếm đến một giá trị nhất định. Nó cho phép thực hiện một đoạn mã (ngắt) định kỳ, không phụ thuộc vào vòng lặp chính (loop() hoặc main()).

Gọi mỗi 1ms → 8ms hoàn thành 1 chu kỳ → ~125Hz → mắt thấy sáng liên tục



**ESP8266 là mô-đun giao tiếp WiFi**, đóng vai trò **kết nối không dây** với hệ thống bên ngoài:

* Giao tiếp với **điện thoại** qua app hoặc web
* Điều khiển LED Cube từ xa qua **WiFi**
* Nhận hiệu ứng hoặc dữ liệu từ **máy tính/web** để hiển thị lên cube

Giao tiếp SPI

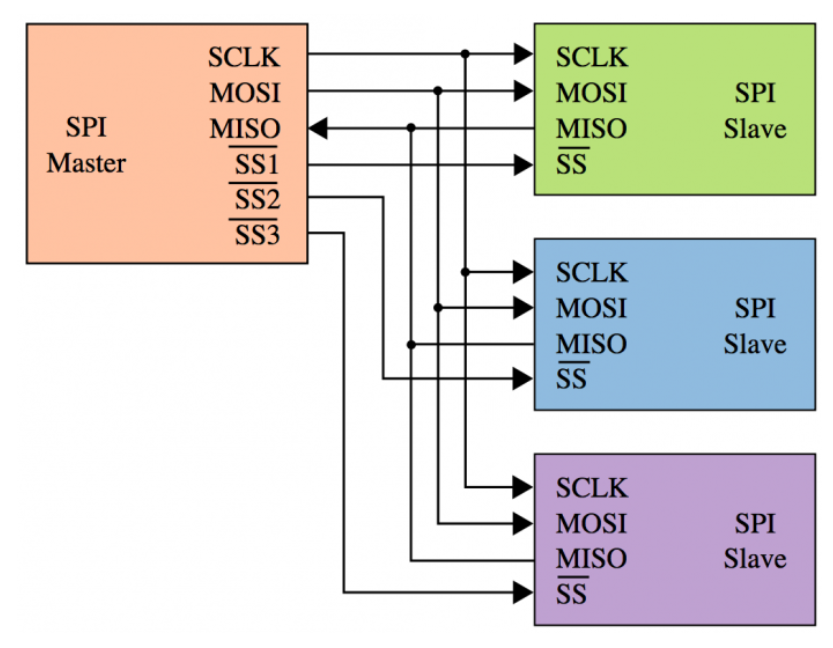
**SPI Serial Peripheral Interface** là một chuẩn truyền [thông nối tiếp đồng bộ](https://vi.wikipedia.org/wiki/Truy%E1%BB%81n_th%C3%B4ng_n%E1%BB%91i_ti%E1%BA%BFp) dùng để chuyền dữ liệu ở chế độ song công toàn phần (full duplex).

Giao thức SPI thường được sử dụng On Board hoặc các đường tín hiệu ngắn. Tốc độ của SPI khá cao thường phụ thuộc vào tốc độ xung truyền vào bộ SPI

Giao thức SPI là dao thức dạng Master –Slave trong đó Master giữ quyền điều khiển xung Clock và chọn Slave nào giao tiếp với mình.

Bus SPI bao gồm 4 đường tín hiệu:

* MOSI (Master Out Slave In): Đường truyền tín hiệu từ Master đến Slave
* MISO (Master In Slave Out): Đường truyền tín hiệu từ Slave đến Master
* CLK (Serial Clock): Đường phát xung đồng hồ được điều khiển bởi Master
* CS (Chip Select): Đường chọn chip giao tiếp với Master, được kéo xuống 0 khi được chọn



<https://tapit.vn/chuan-giao-tiep-spi-tren-stm32f4/>

Nguyên lý hoạt động như sau:

1Khi muốn truyền nhận dữ liệu tới các Slave, đầu tiên Master kéo đường CS kết nối từ Master tới Slave đó xuống 0.

2Gửi xung Clock, tương ứng với mỗi Clock sẽ gửi DATA trên chân MOSI tại thời điểm Clock ở mức cao (hoặc thấp tùy người lập trình)

3Slave cũng có thể gửi ngược lại DATA tại chân MISO tới Master

4Sự truyền nhận dữ liệu là liên tục nên SPI thường có tốc độ rất cao

Có 4 chế độ hoạt động của SPI dựa trên hai Bit **CPOL** và **CPHA**

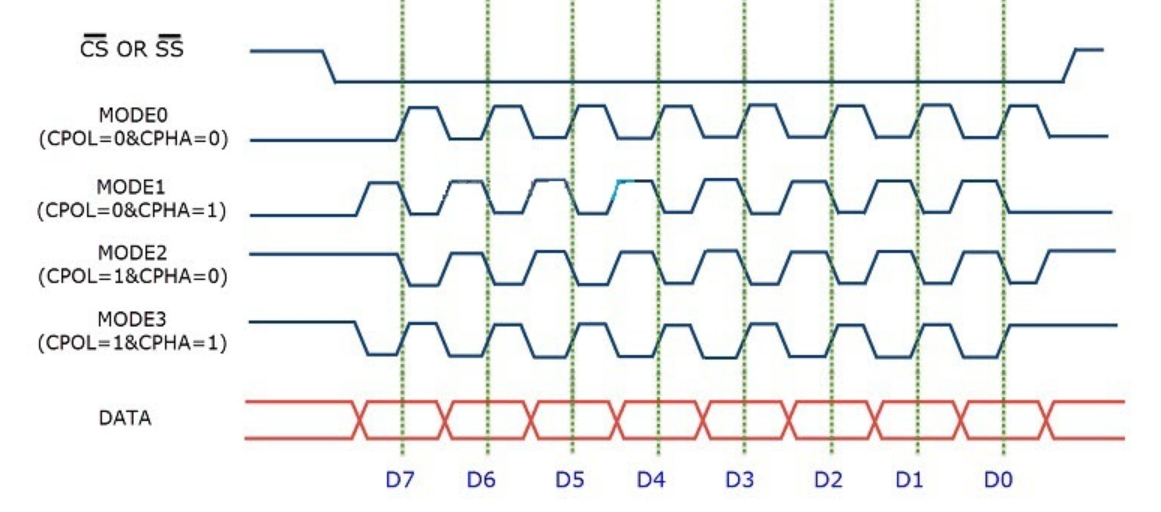
CPOL: Clock Polarity xác định mức tín hiệu tại Clock lúc nhàn rồi (Idle), Nếu CPOL = 0, lúc không gửi dữ liệu (nhàn rỗi) Clock sẽ ở mức thấp CPOL = 1 sẽ ngược lại

CPHA: Phase Clock xác định quá trình truyền dữ liệu tại cạnh lên (CPHA = 0) hay cạnh xuống CPHA = 1

Mode 0:

1. Mode 0: xảy ra khi Clock Polarity và Clock Phase là 0 (CPOL = 0 và CPHA = 0). Trong Mode 0, truyền dữ liệu xảy ra trong khi cạnh lên của xung đồng hồ.
2. Mode 1: xảy ra khi Clock Polarity là 0 và Clock Phase là 1 (CPOL = 0 và CPHA = 1). Trong mode 1, truyền dữ liệu xảy ra trong khi cạnh xuống của xung đồng hồ.
3. Mode 2:  xảy ra khi Clock Polarity là 1 và Clock Phase là 0 (CPOL = 1 và CPHA = 0). Trong mode 2, truyền dữ liệu xảy ra trong khi cạnh lên của xung đồng hồ.
4. Mode 3:  xảy ra khi Clock Polarity là 1 và Clock Phase là 1 (CPOL = 1 và CPHA = 1). Trong mode 3, truyền dữ liệu xảy ra trong khi cạnh lên của xung đồng hồ.

<https://www.studocu.vn/vn/document/truong-dai-hoc-noi-vu-ha-noi/thong-ke-cho-khoa-hoc-xa-hoi/123doc-tim-hieu-giao-dien-spi/63699890>

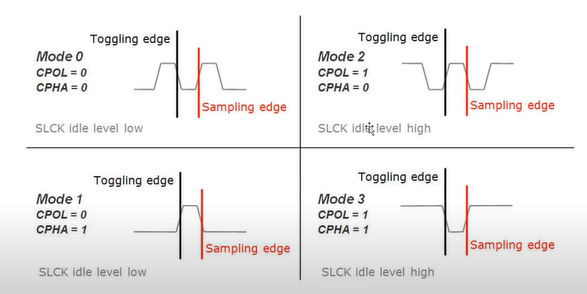


SPI Mode 0: Mô tả: Xung dương, dữ liệu được chốt trước khi dịch

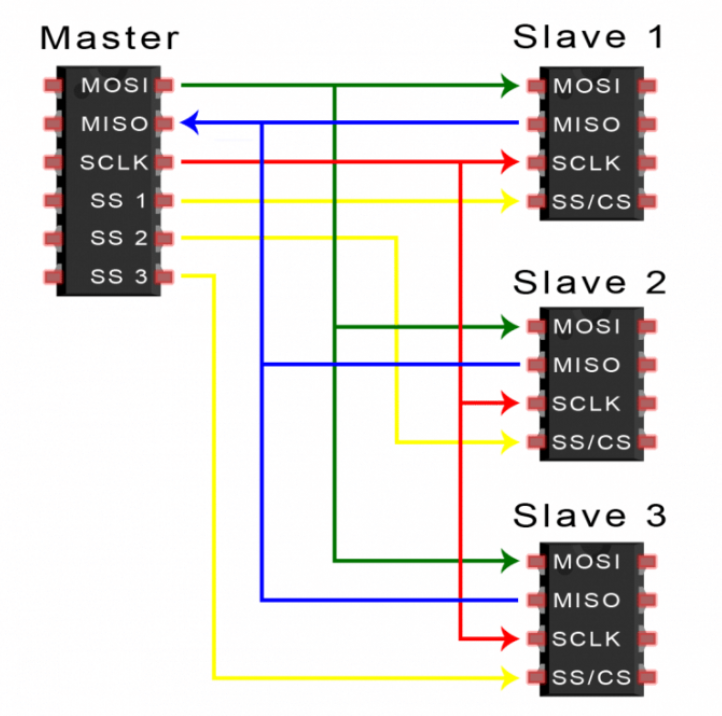
SPI Mode 1: Mô tả: Xung dương, dữ liệu được dịch trước khi chốt

SPI Mode 3: Mô tả: Xung âm, dữ liệu được chốt trước khi dịch

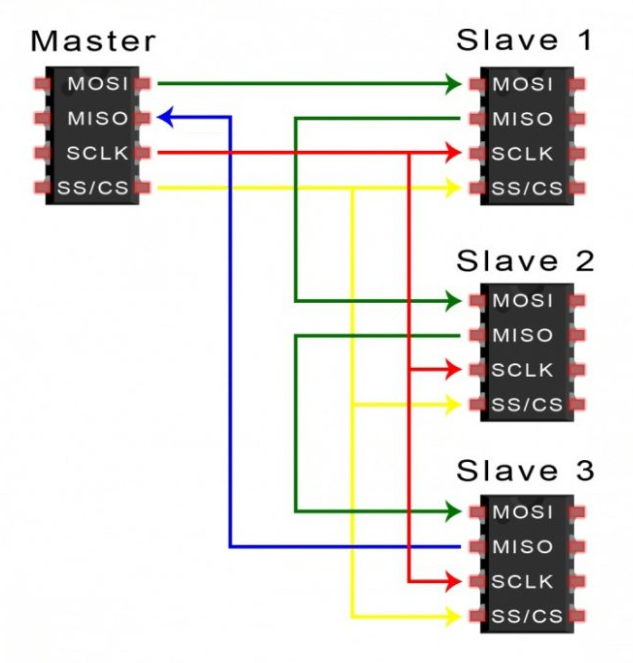
SPI Mode 4: Mô tả: Xung âm, dữ liệu được dịch trước khi chốt



Trong cấu hình Master và các Slave độc ​​lập, Master đã dành riêng các đường Slave Select cho tất cả các Slave và mỗi Slave có thể được chọn riêng lẻ. Tất cả tín hiệu đồng hồ của các Slave được kết nối với chung với SCK của Master.



Trong cấu hình Daisy Chain, chỉ có một đường Slave Select được kết nối với tất cả các Slave. MOSI của Master được kết nối với MOSI của Slave 1. MISO của Slave 1 được kết nối với MOSI của Slave 2 và v.v.. MISO của Slave cuối cùng được kết nối với MISO của Master.



Giao tiếp **SPI** từ các chân:

* PA5: SCK (SRCLK của 74HC595)
* PA7: MOSI (SER – Data input của 74HC595)
* GPIO: LATCH (RCLK)

**PA2 & PA3**: UART2 kết nối ESP8266

**PA1**: Nút nhấn (BUTTON)

Cấp nguồn từ 5V và 3.3V

* Kết nối UART2 với STM32:
  + PA2 (TX2) → RX của ESP
  + PA3 (RX2) ← TX của ESP

→ ESP8266 cho phép **điều khiển từ xa qua WiFi**  
(Có thể qua web/app/MQTT...)



### 4️⃣ **Tầng điều khiển bằng** 2SC2383 **(Q1–Q8)**

* Có 8 transistor 2SC2383, mỗi con điều khiển **1 tầng LED**
* Khi chân base được kéo lên (HIGH), 2SC2383 dẫn → GND cho tầng → LED sáng
* 2SC2383 nhận tín hiệu điều khiển từ các chân GPIO của STM32

→ Đây là phần **multiplexing theo tầng**

## 📡 Vai trò của ESP8266:

* Nhận lệnh từ **điện thoại/app/web**
* Gửi dữ liệu qua UART đến STM32
* STM32 giải mã lệnh và cập nhật LED Cube

 **8 IC 74HC595** (mỗi IC điều khiển 8 LED → 64 LED/tầng)

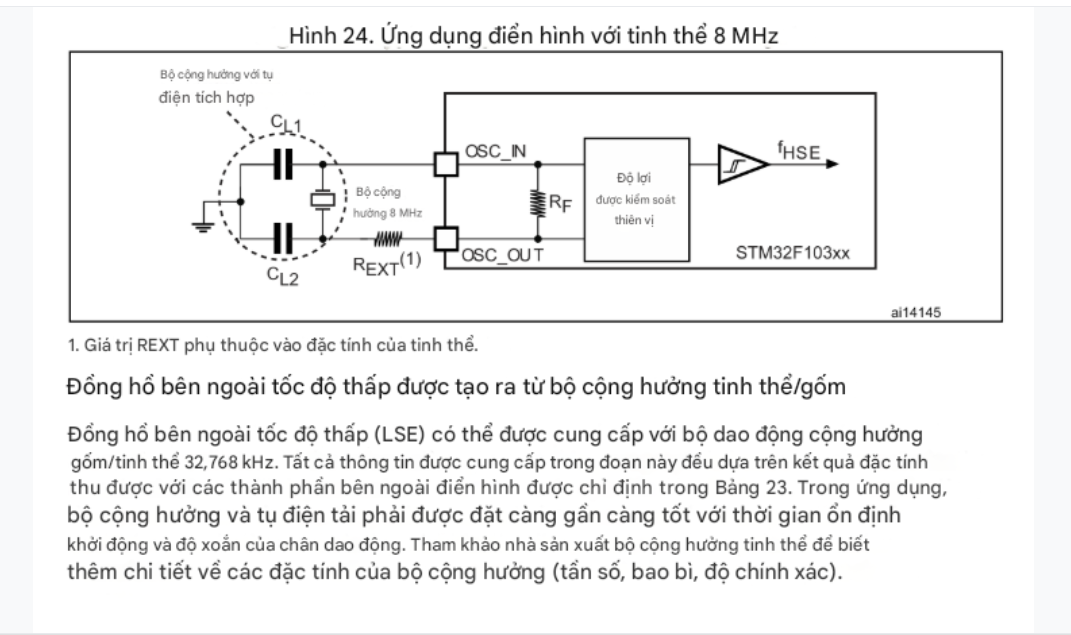
 **8 transistor TIP41** (NPN) để bật từng tầng

 STM32 giao tiếp SPI → 74HC595

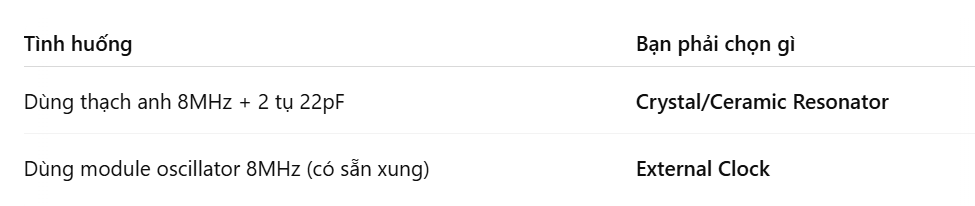
 Quét nhanh từng tầng một (multiplexing)

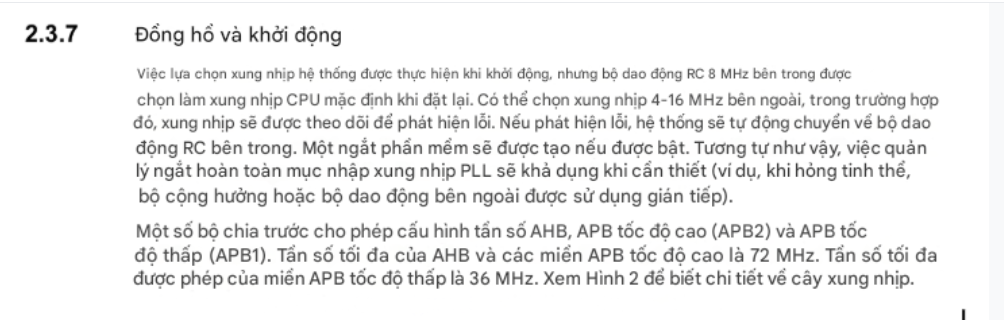
Các chân 11, 14 và 12 được kết nối với các chân GPIO của vi điều khiển. Trong đó chân 11 là Clock phát xung nhịp không đổi để giữ thời gian. Chân 14 là Data gửi dữ liệu về chân đầu ra nào phải ở mức thấp và chân nào sẽ ở mức cao. Chân 12 là Latch cập nhật dữ liệu nhận được vào các chân đầu ra khi đặt ở mức cao, chân này cũng có thể được giữ ở mức cao vĩnh viễn.

MODE 0



Rext: trở giảm chấn feedback resistor

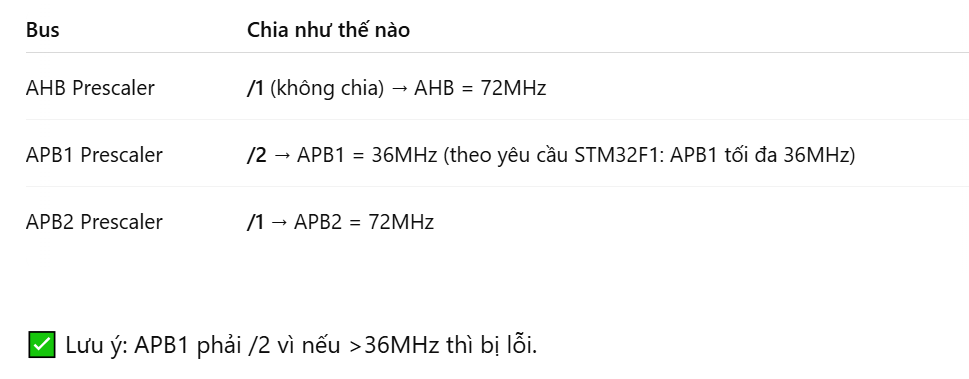


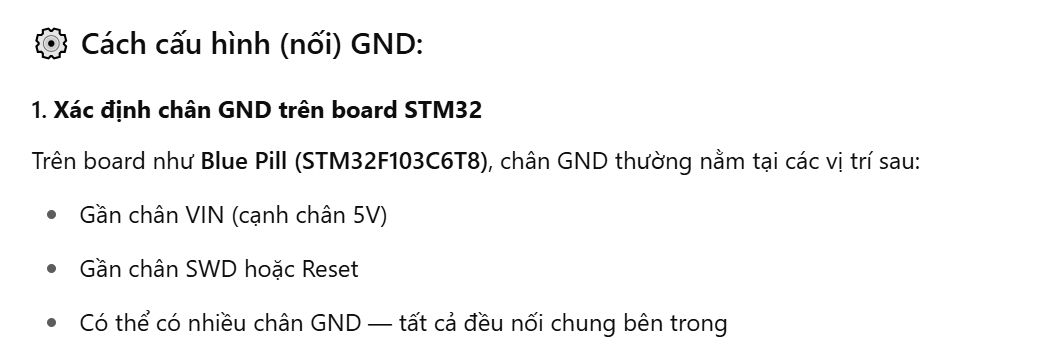


**RCC**, cấu hình:

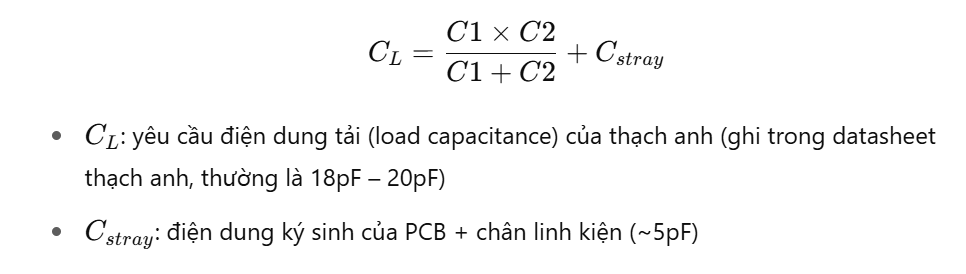
* **HSE**: Enabled ✅
* **HSE Source**: chọn **Crystal/Ceramic Resonator**
* **Chọn PLL Source**:  
  ➔ **HSE** (không phải HSI)
* **PLL Multiplication Factor**:  
  ➔ chọn **x9**

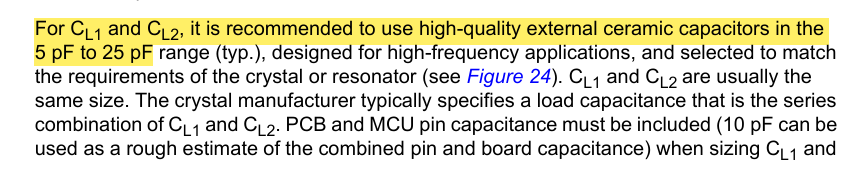
Vì 8MHz × 9 = 72MHz (chuẩn hệ thống STM32F1)



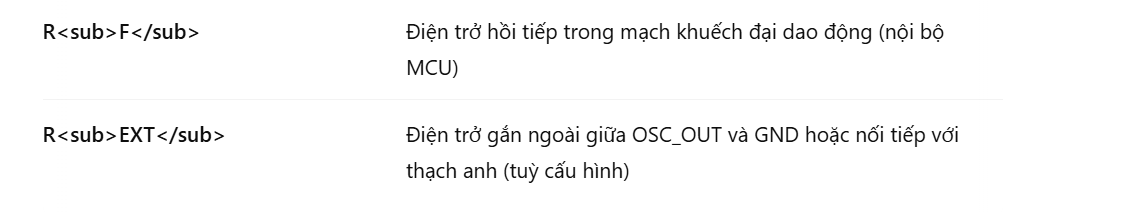


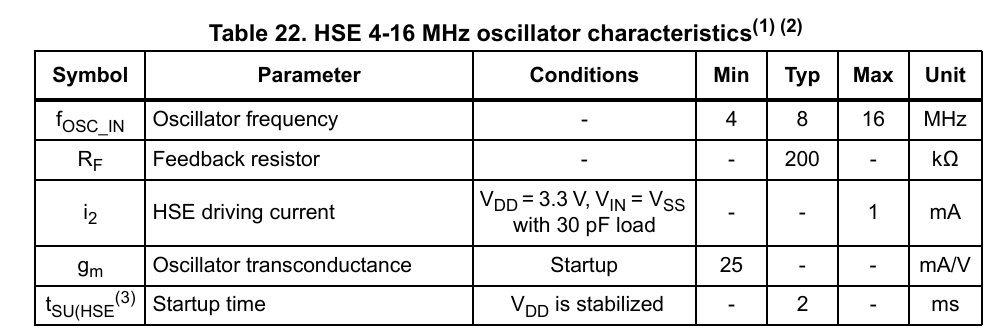
Chọn Thạch Anh 8MHz DIP HC49-S, điện dung tải 18Pf

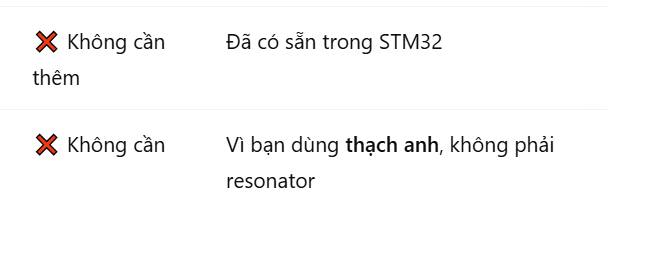


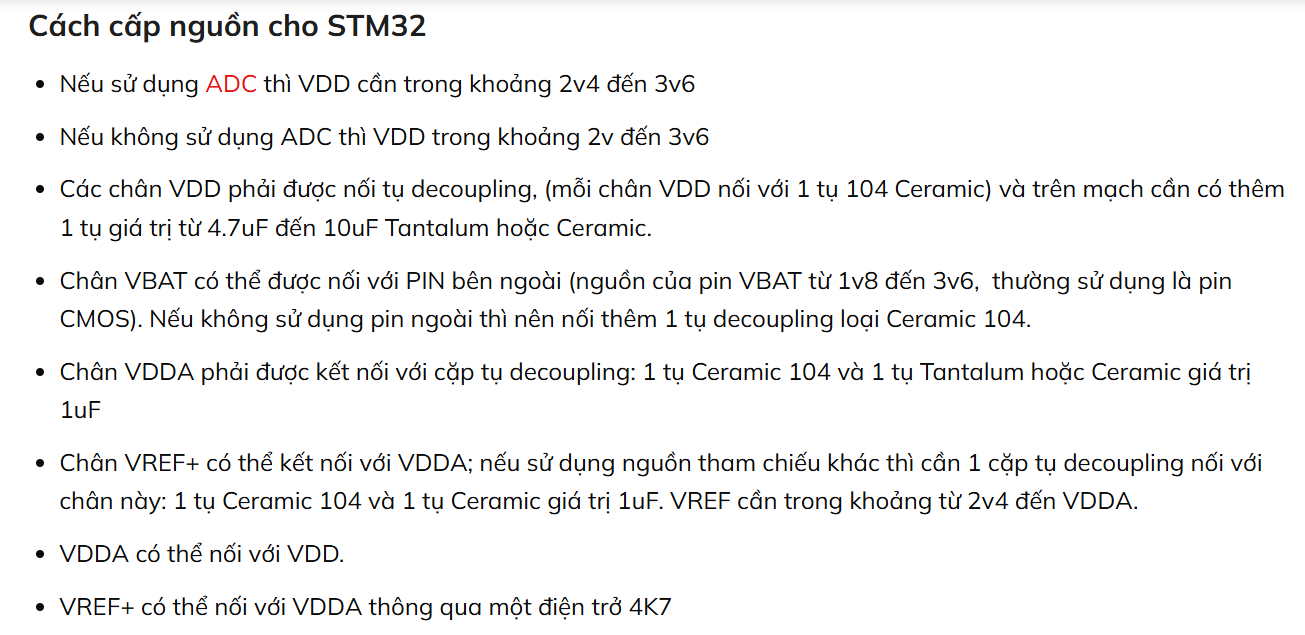


C1=C2=26pf, chọn 22pf

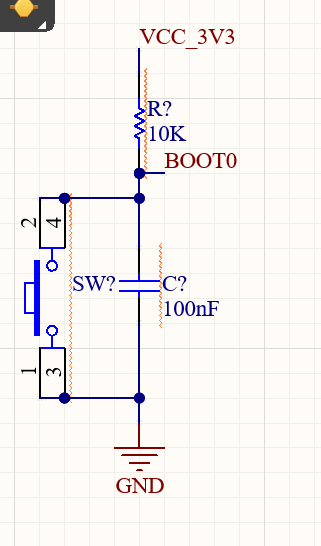


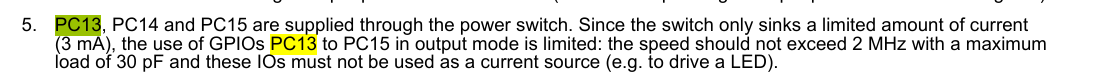




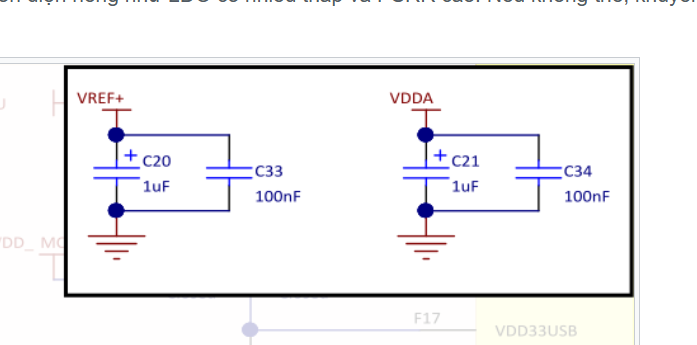


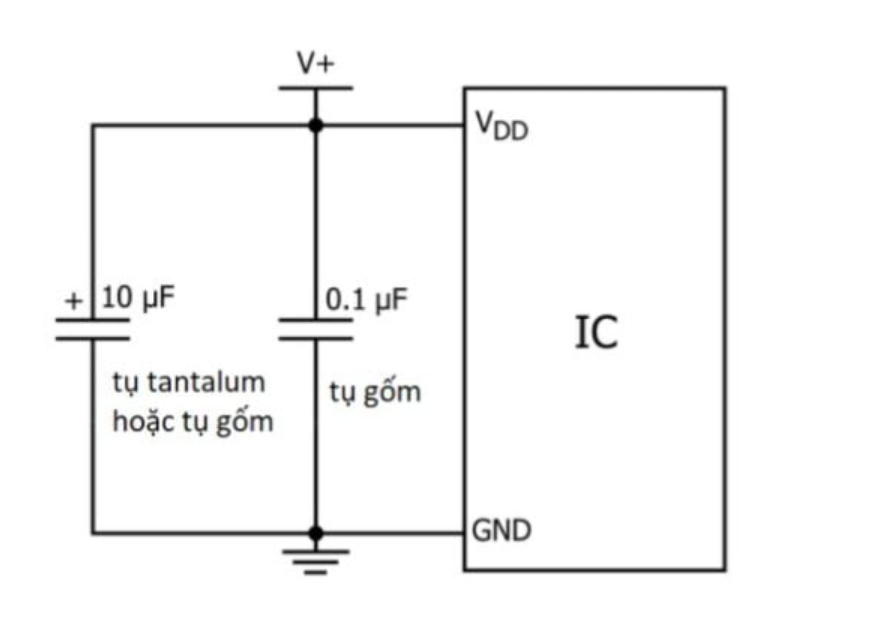


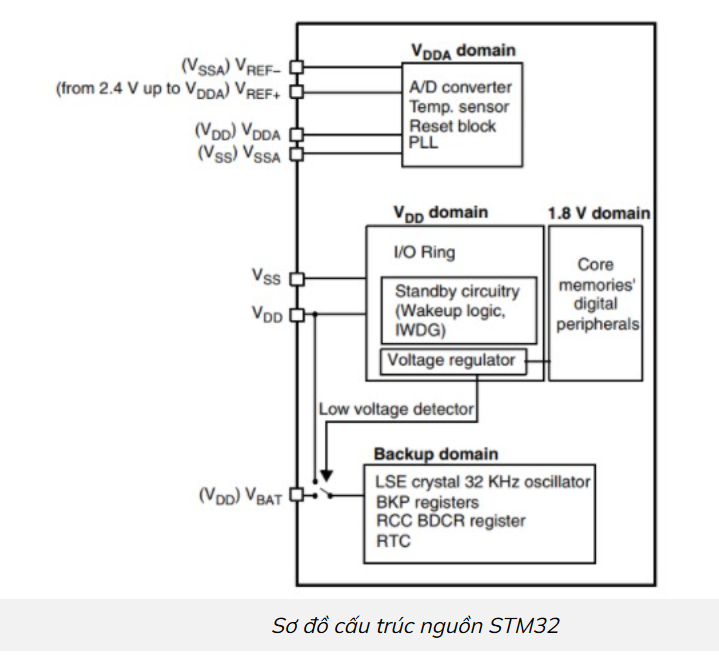


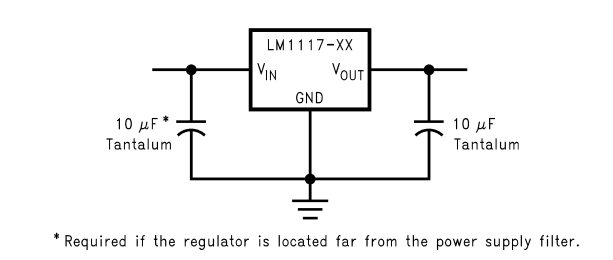


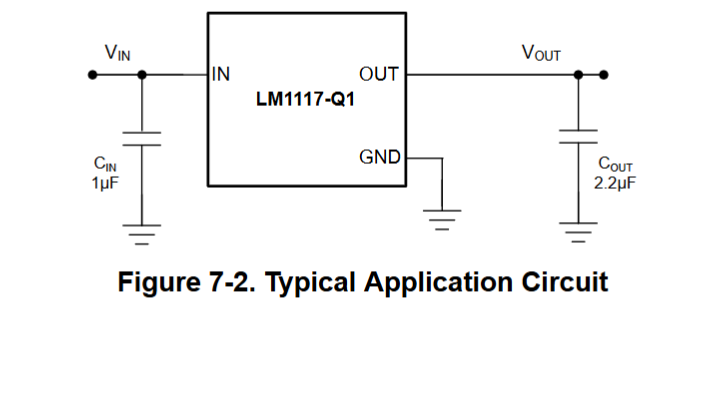
(3.3-2)/2.5Ma=520ohm->510ohm











1A-5V LM1117



