

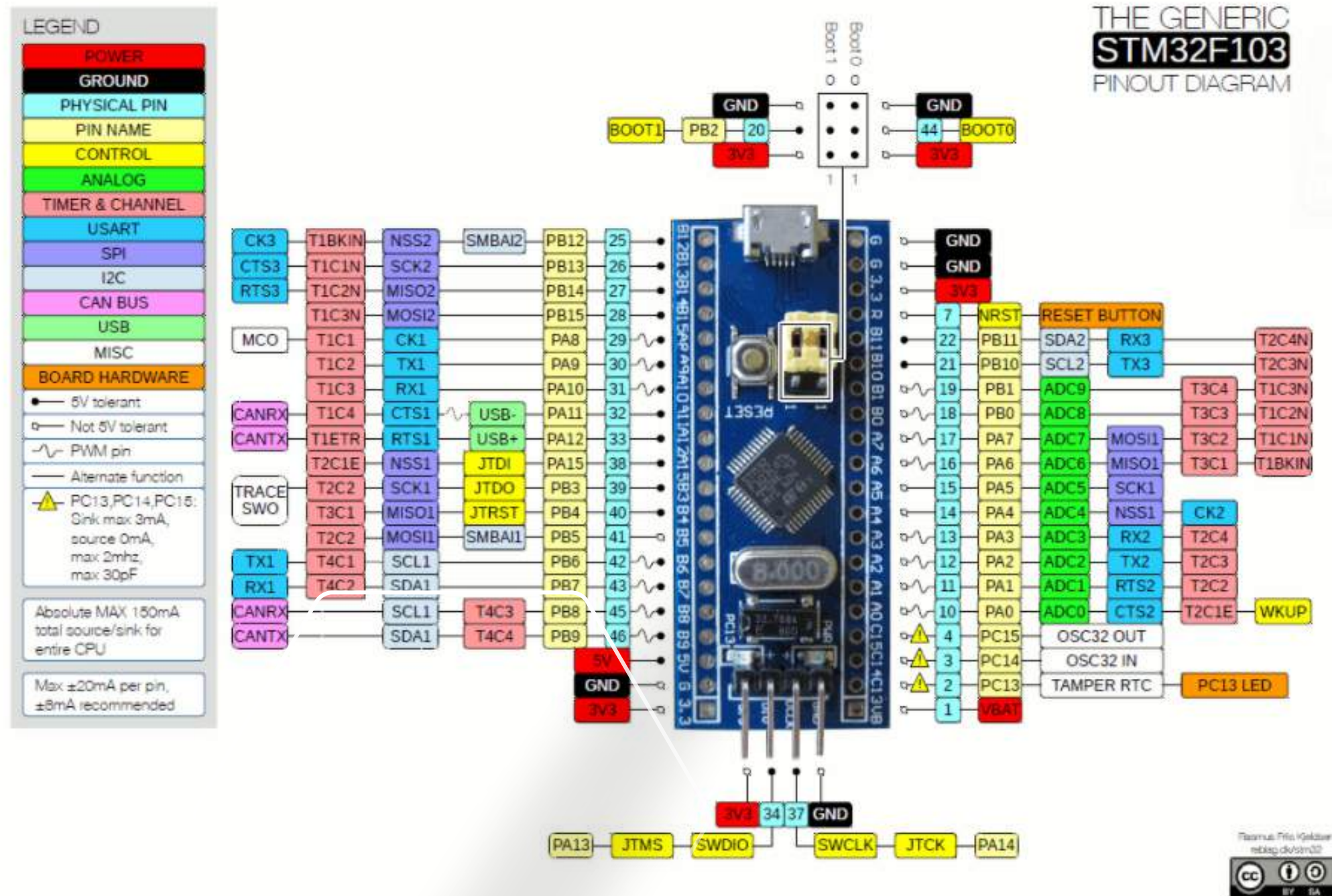
LẬP TRÌNH ARM - STM32

TỔNG QUAN
STM32F103C8T6

Date: July 25, 2025



GIỚI THIỆU



STM32F103C8T

family type series Sub series Pin count Flash size package



GIỚI THIỆU

STM32

Family (Dòng sản phẩm)

Dòng vi điều khiển 32-bit của
hãng STMicroelectronic
ARM Cortex-M -> STM32

1

Series (Dòng vi xử lý)

0: Cortex-M0
1,2: Cortex-M3
3,4: Cortex-M4
7: Cortex-M7

F

Type (Loại vi điều khiển)

L: Low power
F: Mainstream
H: High performance
W: Wireless

03

Sub - series (Dòng con)

Xác định cấu hình chi tiết hơn trong
cùng một series.
Các dòng con có số cao hơn
thường có nhiều tính năng hơn.
Ex: F101 < F102 < F103 < F105...

GIỚI THIỆU

C

Pin count (số lượng chân)

C: 48 pins

R: 64 pins

V: 100 pins

Z: 144 pin

8

Flash size

6: 32KB

E: 512KB

8: 64KB

G: 1MB

B: 128KB

C: 256KB

T

Package

T: LQFP

U: UFQFPN

Y: WLCSP



- **Lõi: ARM 32 bit Cortex M3**
- **Tần số hoạt động lên tới 72 MHz**
- **Bộ nhớ: 64 Kb Flash , 20Kb SRAM**
- **Điện áp: 2~3.6 VDC**
- **Tổng số I/O: 37**
- **ADC: 2x12 bit, tần số lấy mẫu 1MHz**
- **DAC: Không**
- **DMA: Điều khiển 7 kênh DMA**
- **Timer: 4 bộ, 16 bit (IC, OC, PWM)**
- **Giao diện kết nối: 2xI2C, 3xUSART, 2xSPI, CAN, 1xUSB 2.0 full- speed, 1xCAN**
- **Kiểu chân: LQFP48**

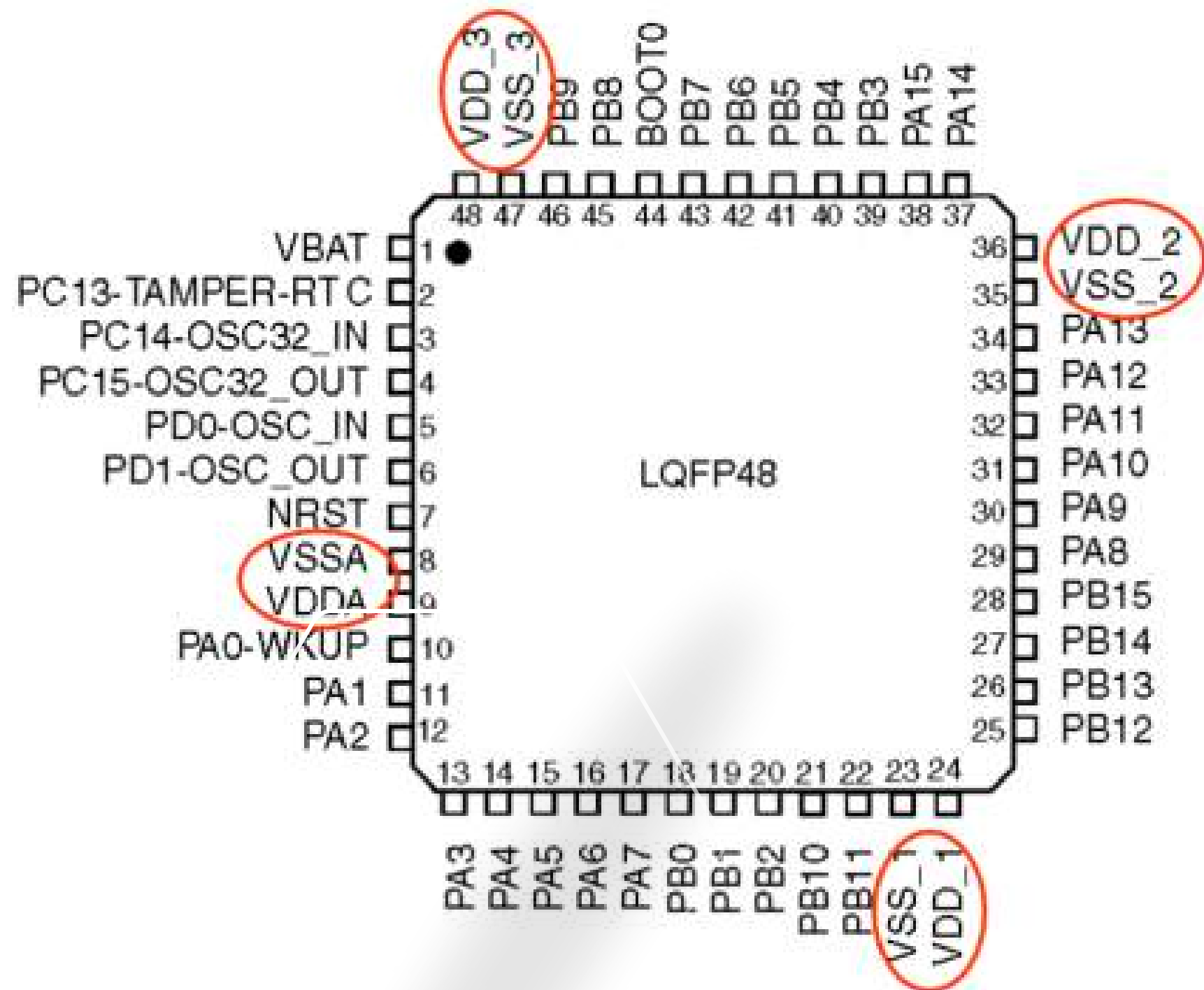
LẬP TRÌNH ARM - STM32

GPIO

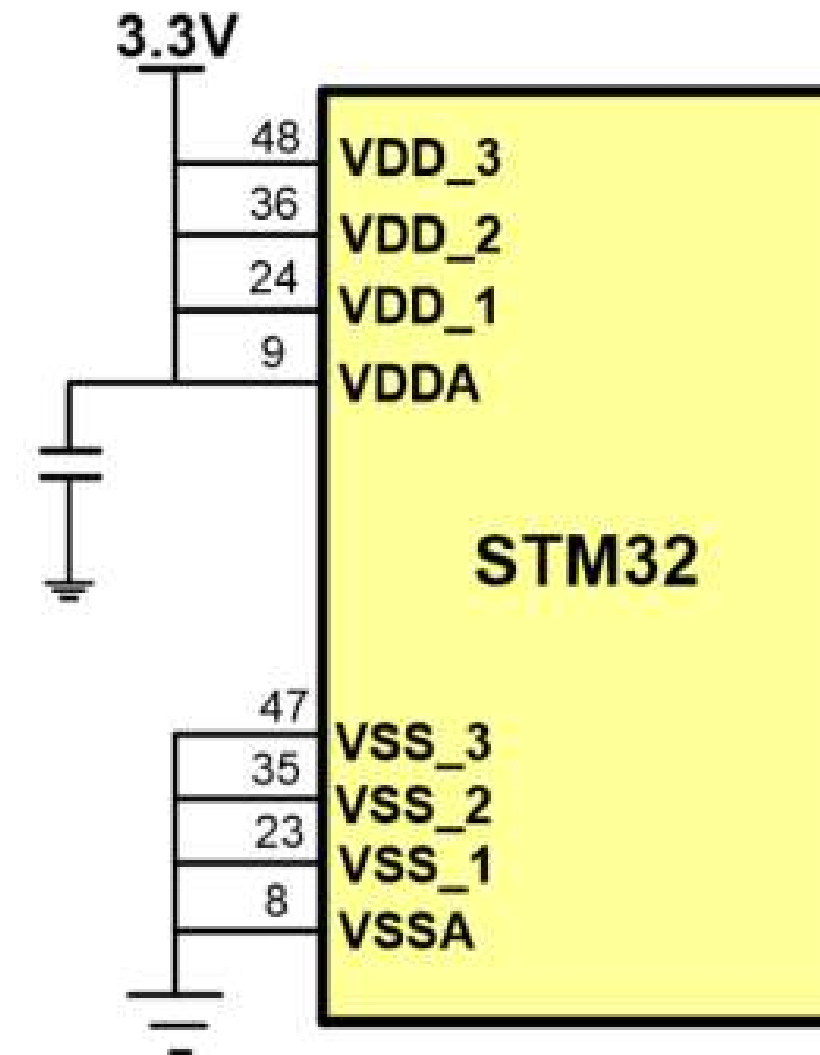
Date: July 25, 2025



STM32F1 - PINOUT

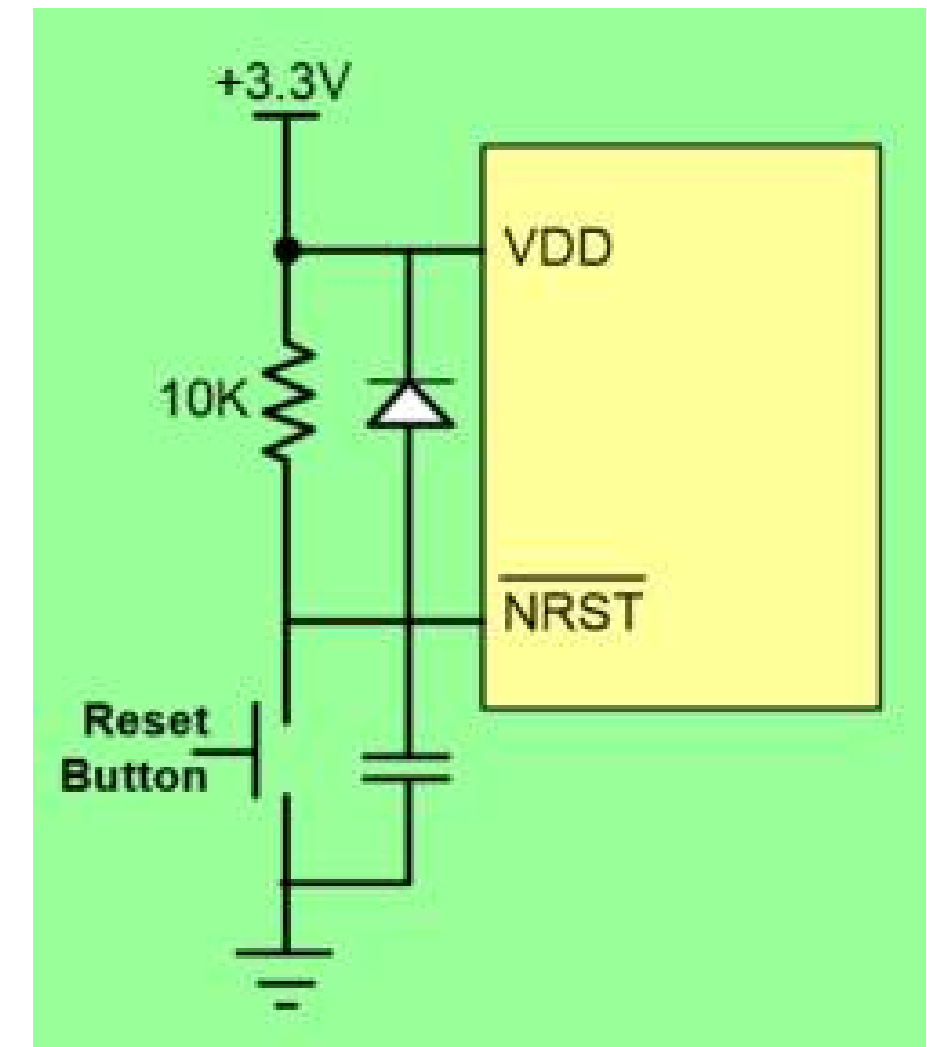
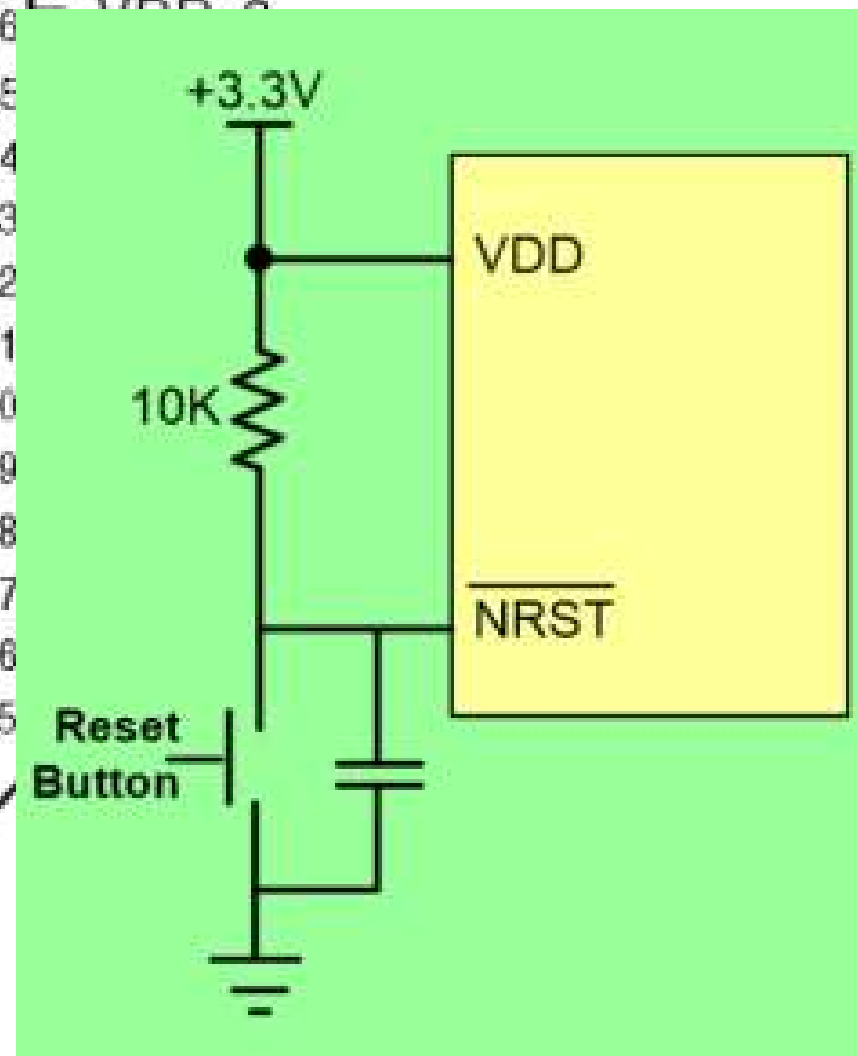
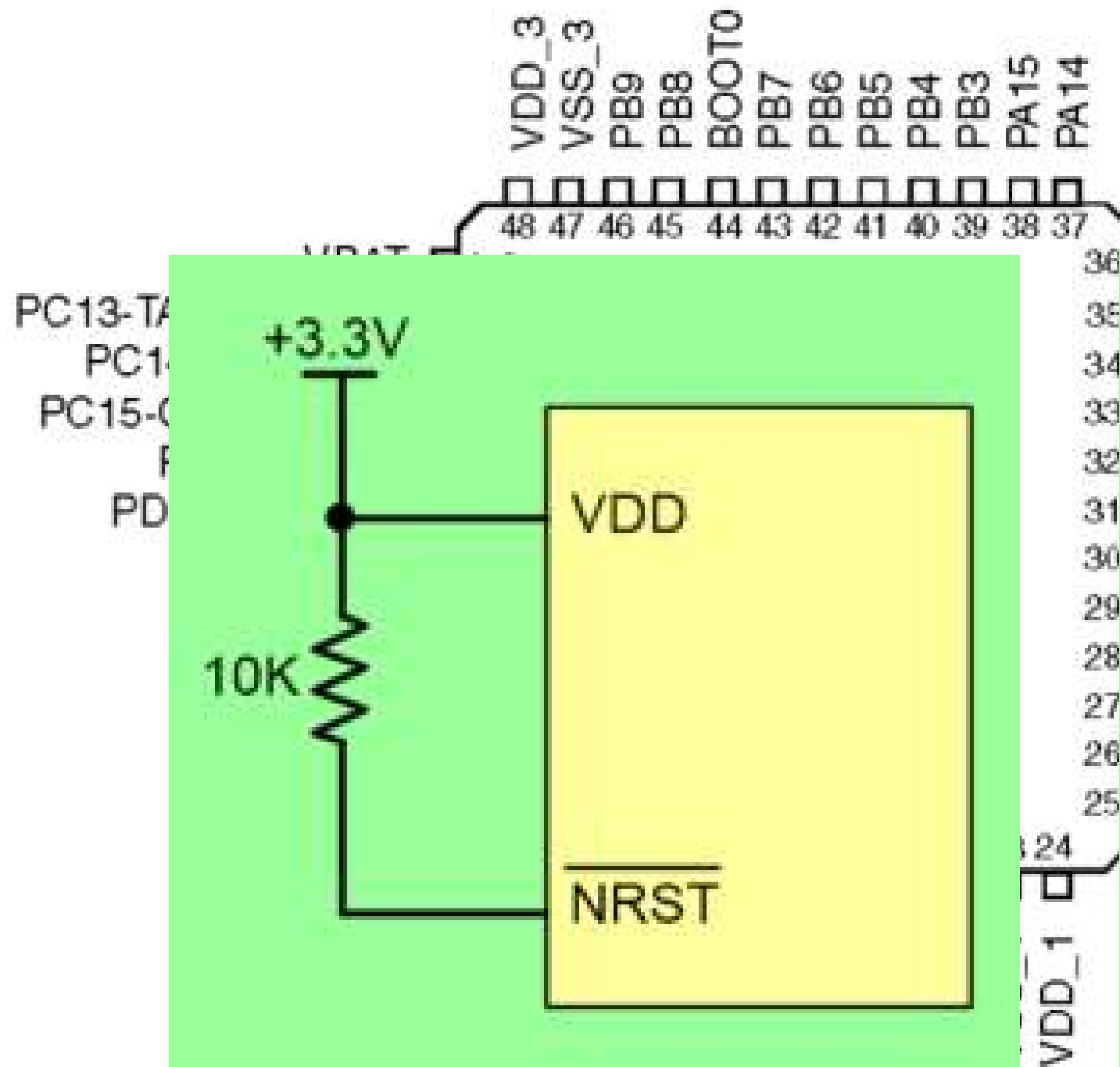


Power Supply Pins

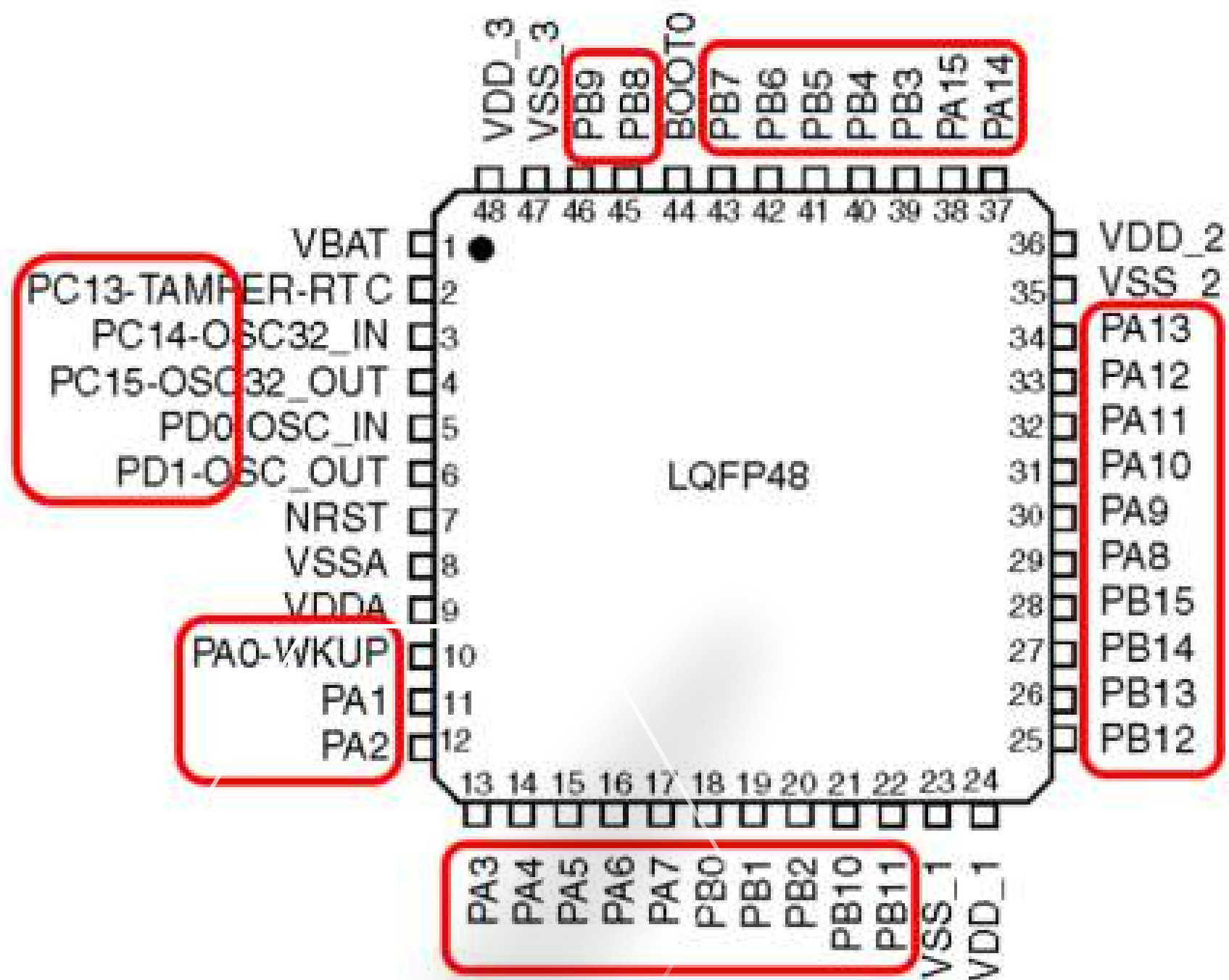


STM32F1 - PINOUT

Reset



STM32F1 - PINOUT

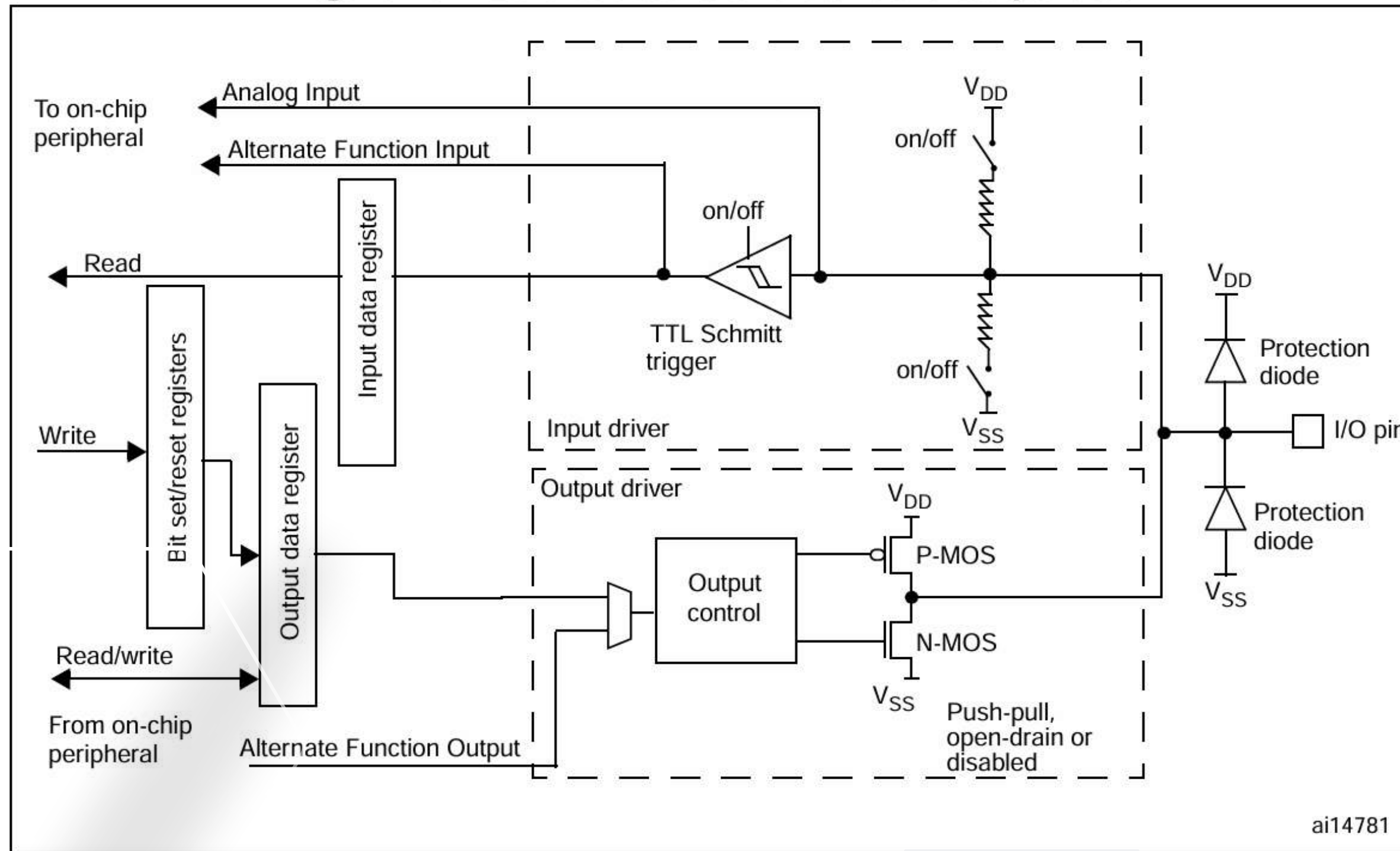


Power Supply Pins

- GPIOA (PAn)
- GPIOB (PBn)
- GPIOC (PCn)
- GPIOD

STM32F1 - PINOUT

Figure 13. Basic structure of a standard I/O port bit



STM32F1 - GPIO Mode

1

Mode Input

- **Input floating:** ngõ vào thả nổi
- **Input pull-up:** ngõ vào có trở kéo lên VCC
- **Input pull-down:** ngõ vào có trở xuống GND
- **Analog:** sử dụng ADC, DAC

2

Mode Output

- **Out open-drain:** ngõ ra cực máng thả nổi
- **Output push-pull:** ngõ ra đẩy kéo

3

Mode AFIO

- **Alternate function push-pull:** vừa là ngõ vào vừa là ngõ ra, không có trở kéo lên và kéo xuống ở input, output giống Out push-pull, remap.
- **Alternate function open-drain:** giống Alternate function push-pull nhưng cực máng để hở

STM32F1 - GPIO Library

```
GPIO_InitTypeDef      X;  
(Khai báo biến X thuộc kiểu GPIO_InitTypeDef)
```

```
X.GPIO_Pin = A;  
(Lệnh chọn chân cấu hình)
```

A:

```
GPIO_Pin_0  
GPIO_Pin_1  
...  
GPIO_Pin_15  
GPIO_Pin_All
```

A:

Chọn chân 0

Chọn chân 1

...

Chọn chân 15

Chọn tất cả các chân 0-15

Chú ý: Nếu muốn chọn nhiều chân thì ta sử dụng lệnh OR "|" giữa các lựa chọn

Ví dụ:

```
X.GPIO_Pin = GPIO_Pin_0|GPIO_Pin_3;
```

STM32F1 - GPIO Library

X.GPIO_Mode = B;
(Lệnh cấu hình chế độ hoạt động)

B:

GPIO_Mode_AIN
GPIO_Mode_IN_FLOATING
GPIO_Mode_IPD
GPIO_Mode_IPU
GPIO_Mode_Out_OD
GPIO_Mode_Out_PP
GPIO_Mode_AF_OD
GPIO_Mode_AF_PP

B:

Cấu hình chế độ hoạt động
Ngõ vào tương tự
Ngõ vào thả nổi
Ngõ vào kéo xuống
Ngõ vào kéo lên
Ngõ ra cực thu hở
Ngõ ra đẩy kéo
Chức năng thay thế cực thu hở
Chức năng thay thế đẩy kéo

X.GPIO_Speed = C;
(Lệnh cấu hình tốc độ)

C:

GPIO_Speed_10MHz
GPIO_Speed_2MHz
GPIO_Speed_50MHz

C:

GPIO_Speed_10MHz
GPIO_Speed_2MHz
GPIO_Speed_50MHz

GPIO_Init(GPIOx, &X);
(Lệnh cấu hình cho GPIOx theo các thông số
được lưu trong biến X)

STM32F1 - GPIO Library

`uint16_t A;`

`A = GPIO_ReadInputDataBit(B, C);`

(Lệnh đọc trạng thái logic chân C của port B rồi lưu vào biến A)

B:

GPIOA

GPIOB

...

C:

GPIO_Pin_0

GPIO_Pin_1

...

GPIO_Pin_15

B: PORT cần đọc dữ liệu

Đọc PORTA

Đọc PORTB

...

C:

Chọn chân 0

Chọn chân 1

...

Chọn chân 15

`GPIO_Write(A, B);`

(Lệnh xuất dữ liệu ra 16 chân của PORT)

A:

GPIOA

GPIOB

...

A: PORT cần xuất dữ liệu

Xuất ra PORTA

Xuất ra PORTB

...

B: Số nguyên 16 bit ứng với trạng thái 16 chân của PORT cần xuất.

STM32F1 - GPIO Library

GPIO_SetBits(A, B) ;
(Lệnh xuất mức '1' ra 1 chân vi điều khiển)

A:

GPIOA

...

GPIOG

B:

GPIO_Pin_0

...

GPIO_Pin_15

GPIO_Pin_All

A: PORT cần xuất

Xuất ra PORTA

...

Xuất ra PORTG

B: Chân cần xuất

Chân số 0

...

Chân số 15

Tất cả các chân của PORT

GPIO_ResetBits(A, B) ;
(Lệnh xuất mức '0' ra 1 chân vi điều khiển)

A:

GPIOA

...

GPIOG

B:

GPIO_Pin_0

...

GPIO_Pin_15

GPIO_Pin_All

A: PORT cần xuất

Xuất ra PORTA

...

Xuất ra PORTG

B: Chân cần xuất

Chân số 0

...

Chân số 15

Tất cả các chân của PORT

STM32F1 - GPIO Library

GPIO_SetBits(A, B) ;
(Lệnh xuất mức '1' ra 1 chân vi điều khiển)

A:

GPIOA

...

GPIOG

B:

GPIO_Pin_0

...

GPIO_Pin_15

GPIO_Pin_All

A: PORT cần xuất

Xuất ra PORTA

...

Xuất ra PORTG

B: Chân cần xuất

Chân số 0

...

Chân số 15

Tất cả các chân của PORT

GPIO_ResetBits(A, B) ;
(Lệnh xuất mức '0' ra 1 chân vi điều khiển)

A:

GPIOA

...

GPIOG

B:

GPIO_Pin_0

...

GPIO_Pin_15

GPIO_Pin_All

A: PORT cần xuất

Xuất ra PORTA

...

Xuất ra PORTG

B: Chân cần xuất

Chân số 0

...

Chân số 15

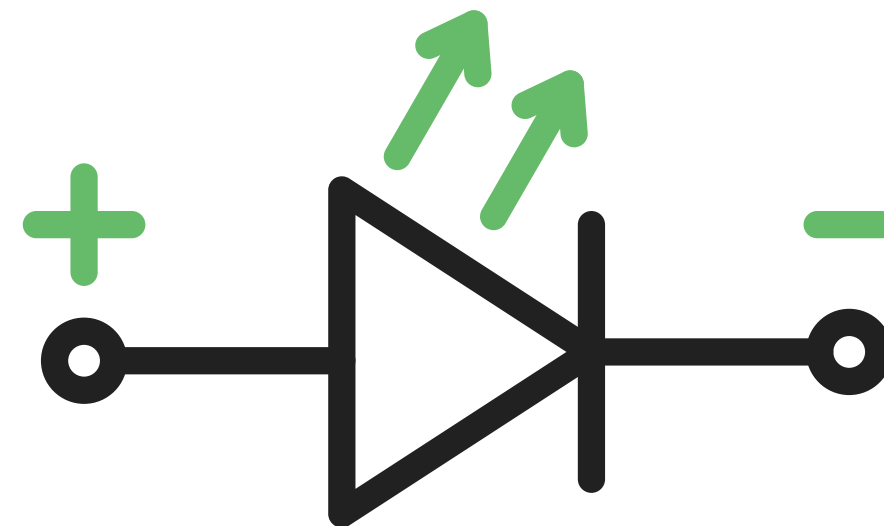
Tất cả các chân của PORT

STM32F1 - LED



LED

- Là viết tắt của Light Emitting Diode
- Có khả năng phát quang
- Bóng đèn led hay còn gọi là đi-ốt phát quang



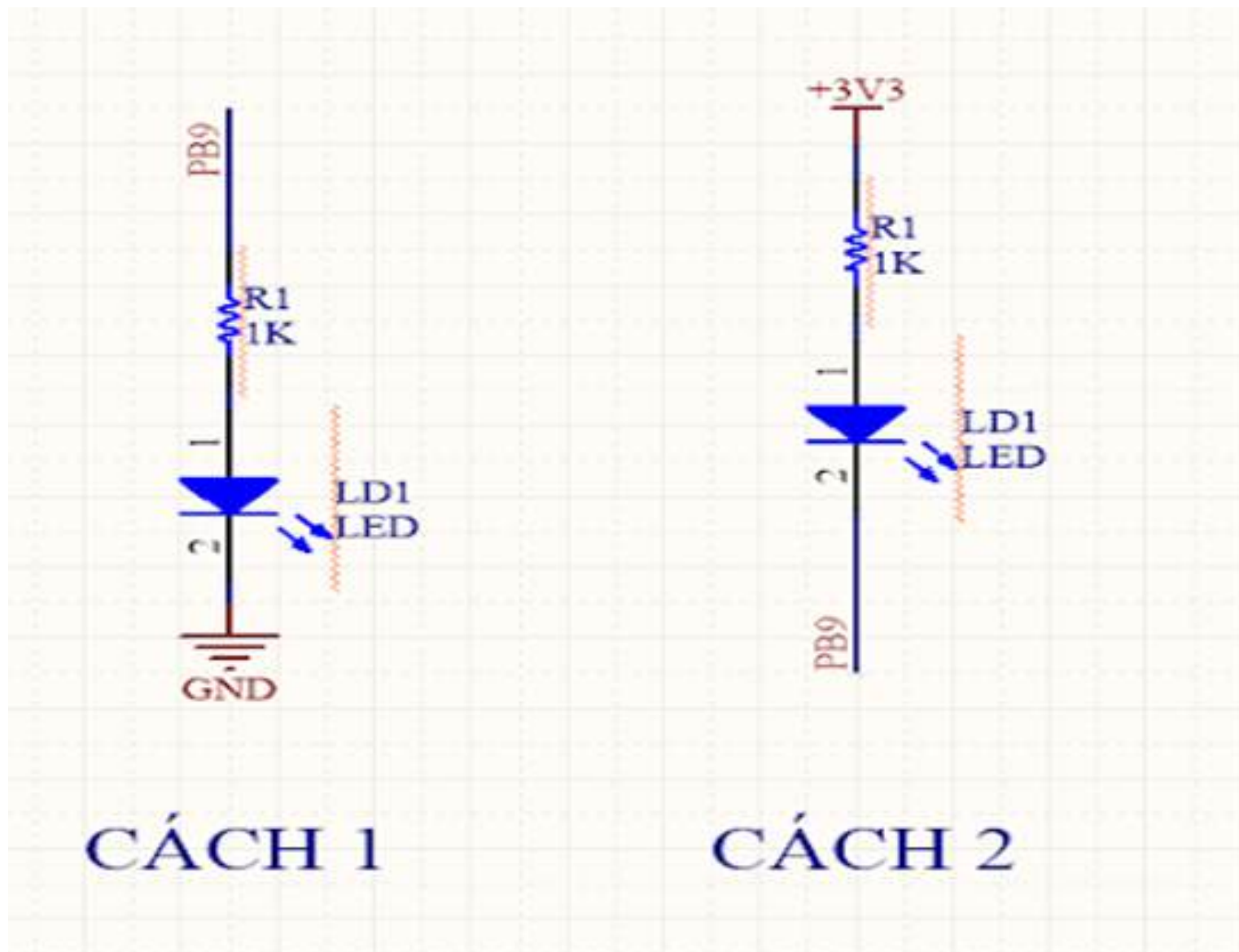
Cấu tạo LED đơn với 2 cực **Anode** và **Cathode**:

- **Anode**: cực dương (chân dài, bản cực nhỏ)
- **Cathode**: cực âm (chân ngắn, bản cực lớn)

STM32F1 - LED



LED



Cách 1	Cách 2
Mức logic 0: LED OFF Mức logic 1: LED ON	Mức logic 0: LED ON Mức logic 1: LED OFF

STM32F1 - GPIO Registers

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CNF7[1:0]		MODE7[1:0]		CNF6[1:0]		MODE6[1:0]		CNF5[1:0]		MODE5[1:0]		CNF4[1:0]		MODE4[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNF3[1:0]		MODE3[1:0]		CNF2[1:0]		MODE2[1:0]		CNF1[1:0]		MODE1[1:0]		CNF0[1:0]		MODE0[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

CONFIGURATION MODE		CNF1	CNF0	MODE1	MODE0	PxODR Register
General purpose output	Push-pull	0	0	01 : 10 MHz 10 : 2 MHz 11 : 50 MHz		0 or 1
	Open-drain		1			0 or 1
Alternate Function output	Push-pull	1	0			X
	Open-drain		1			X
Input	Analog input	0	0	0	X	
	Input floating		1		X	
	Input pull-down	1	0		0	
	Input pull-up				1	

GPIOx_CRL

STM32F1 - GPIO Registers

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CNF15[1:0]		MODE15[1:0]		CNF14[1:0]		MODE14[1:0]		CNF13[1:0]		MODE13[1:0]		CNF12[1:0]		MODE12[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNF11[1:0]		MODE11[1:0]		CNF10[1:0]		MODE10[1:0]		CNF9[1:0]		MODE9[1:0]		CNF8[1:0]		MODE8[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

CONFIGURATION MODE		CNF1	CNF0	MODE1	MODE0	PxODR Register
General purpose output	Push-pull	0	0	01 : 10 MHz 10 : 2 MHz 11 : 50 MHz		0 or 1
	Open-drain		1			0 or 1
Alternate Function output	Push-pull	1	0			X
	Open-drain		1			X
Input	Analog input	0	0	0		X
	Input floating		1			X
	Input pull-down	1	0			0
	Input pull-up					1

GPIOx_CRH

STM32F1 - GPIO Registers

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved										TIM11 EN	TIM10 EN	TIM9 EN	Reserved		
										rw	rw	rw			
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADC3 EN	USART 1EN	TIM8 EN	SPI1 EN	TIM1 EN	ADC2 EN	ADC1 EN	IOPG EN	IOPF EN	IOPE EN	IOPD EN	IOPC EN	IOPB EN	IOPA EN	Res.	AFIO EN
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw

RCC_APB2ENR

Bit-Wise Logical Operators

AND			OR	EX-OR	Inverter
A	B	A&B	A B	A^B	Y=~B
0	0	0	0	0	1
0	1	0	1	1	0
1	0	0	1	1	
1	1	1	1	0	

1110 1111	1110 1111	
& 0000 0001	0000 0001	~ 1110 1011
-----	-----	-----
0000 0001	1110 1111	0001 0100

Shift Operations in C

- `data >> number of bits to be shifted right`
- `data << number of bits to be shifted left`

1110 0000 >> 3

0001 1100

0000 0001 <<2

0000 0100

Setting a bit in a Byte to 1

- We can use | operator to set a bit of a byte to 1

XXXX XXXX		XXXX XXXX
0001 0000	OR	1 << 4
-----		-----
xxx1 xxxx		xxx1 xxxx

```
GPIOA->ODR |= (1<<4); //set bit 4 (5th bit) of GPIOA->ODR
```

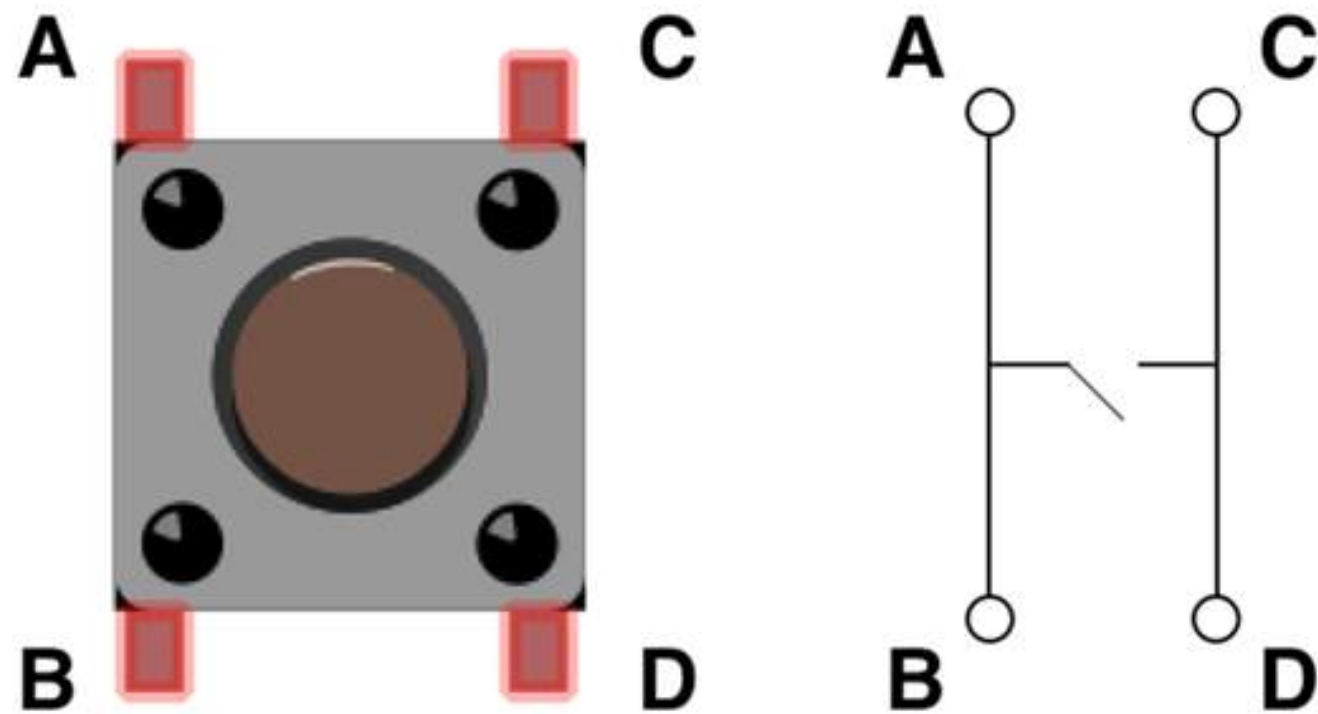
Clearing a bit in a Byte to 0

- We can use `|` operator to set a bit of a byte to 1

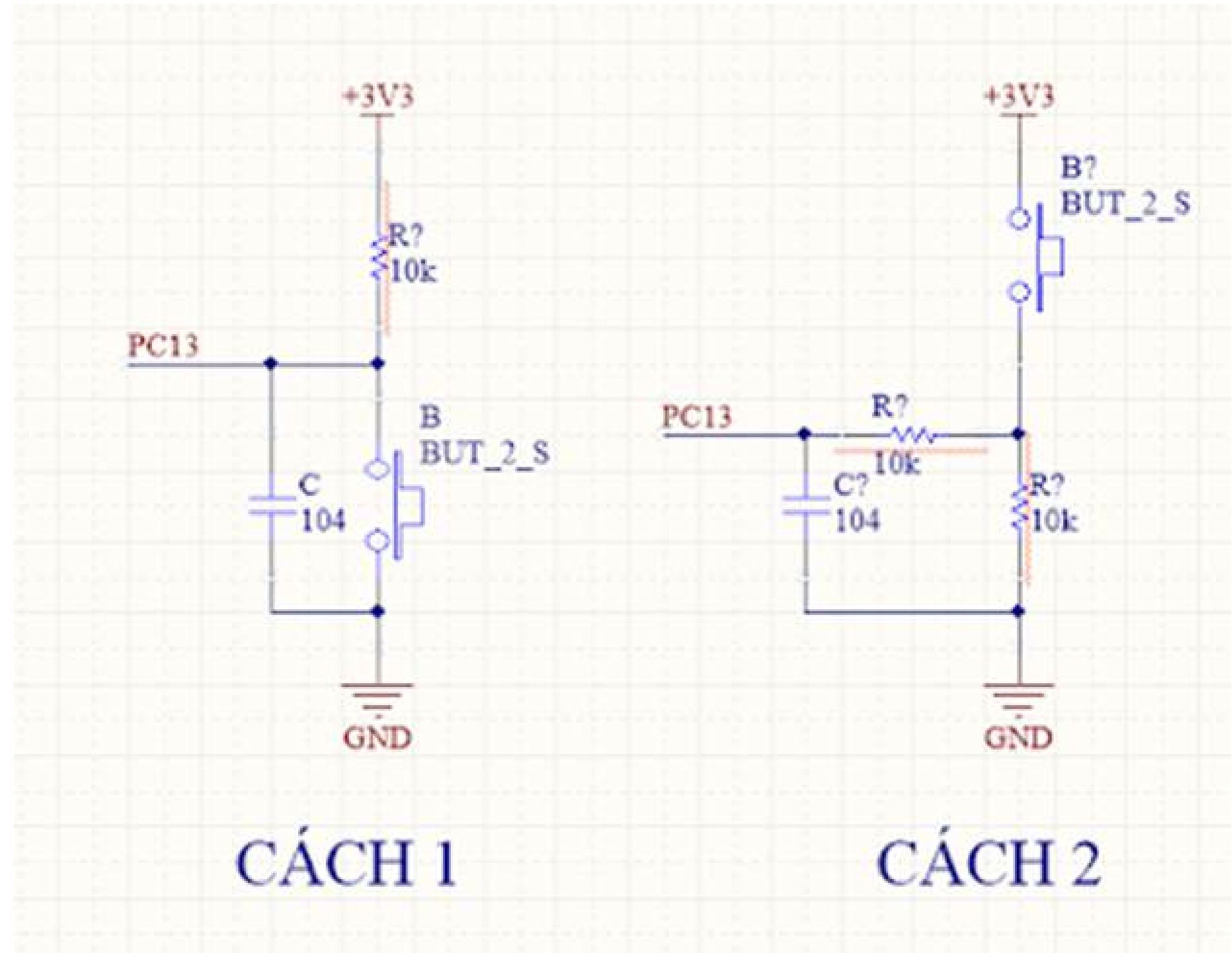
	XXXX	XXXX			XXXX	XXXX
&	1110	1111	OR	&	$\sim(1 \ll 4)$	
	-----				-----	
	xxx0	xxxx			xxx0	xxxx

```
GPIOB->ODR |=  $\sim(1 \ll 4)$ ; //clear bit 4 (5th bit) of GPIOB->ODR
```

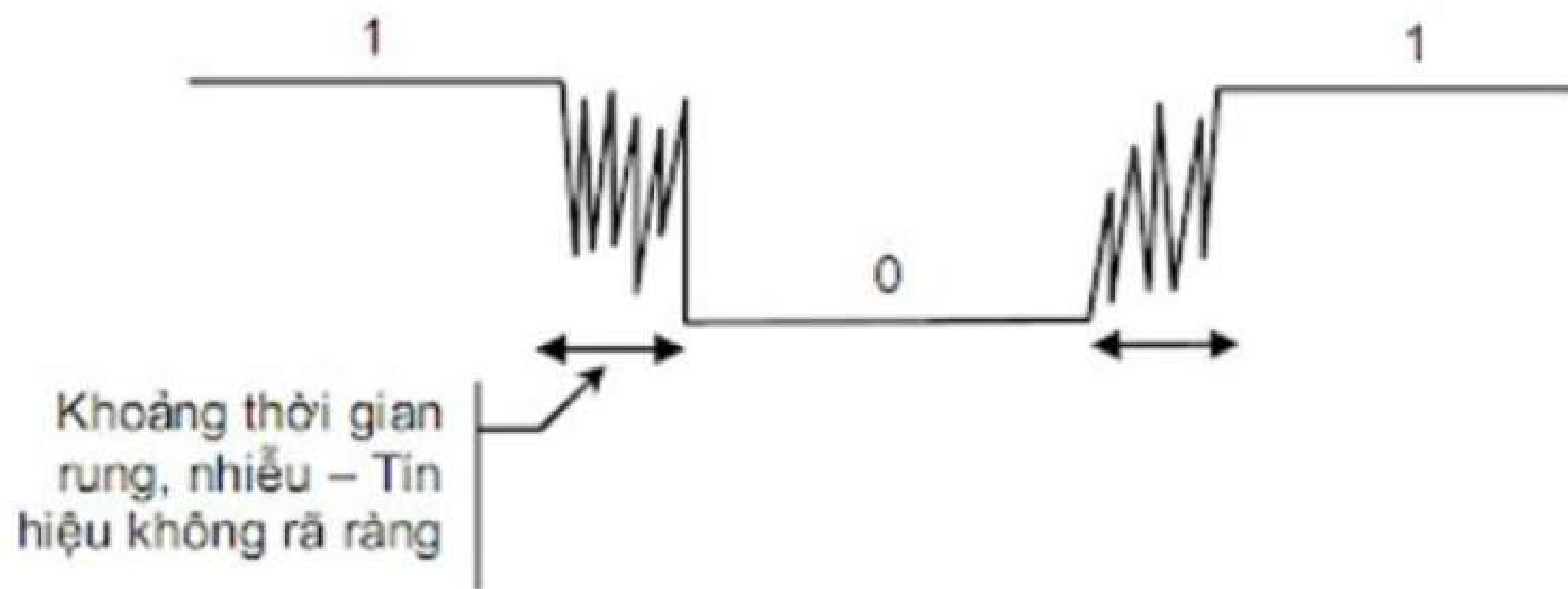
STM32F1 - Button



Cách 1	Cách 2
Tích cực mức 1 Logic 0: nhấn nút Logic 1: nhả nút	Tích cực mức 0 Logic 0: nhả nút Logic 1: nhấn nút



STM32F1 - Button



Tín hiệu rung nhiễu khi nhấn nút



KHỦ NHIỄU???

STM32F1 - GPIO Registers

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IDR15	IDR14	IDR13	IDR12	IDR11	IDR10	IDR9	IDR8	IDR7	IDR6	IDR5	IDR4	IDR3	IDR2	IDR1	IDR0
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

```
uint16_t A;
```

```
A = P->IDR
```

(Lệnh đọc trạng thái logic 16 chân của 1 PORT rồi lưu vào biến A)

GPIOx_IDR