**BỘ CÔNG THƯƠNG**

**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP TP. HCM**



**VI ĐIỀU KHIỂN**

**BÀI BÁO CÁO 7**

**NHÓM 3**

Giảng viên : **PHẠM QUANG TRÍ**

Sinh viên :

* Trần Công Hòa 20017691
* Đỗ Tuấn Duy 20061261
* Nguyễn Thị Thu Hạ 20056011

**TP.HCM – 2022**

**Bài tập mức độ 3+:**

1. **Sơ đồ nguyên lí kết nối phần cứng tối thiểu dùng STM32:**

Sơ đồ nguyên lí mức độ 1 và 2:

Diagram, schematic

Description automatically generated

Sơ đồ nguyên lí mức độ 3:

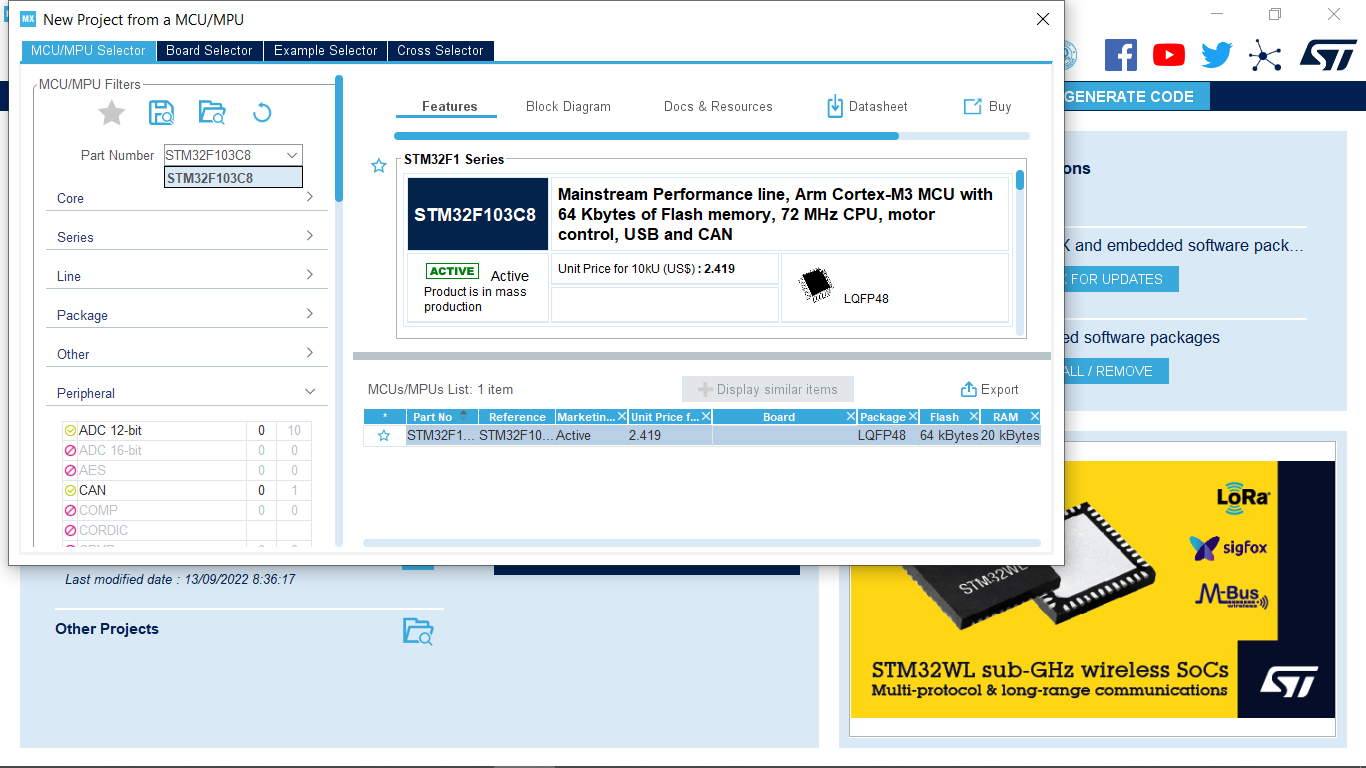
Diagram, schematic

Description automatically generated

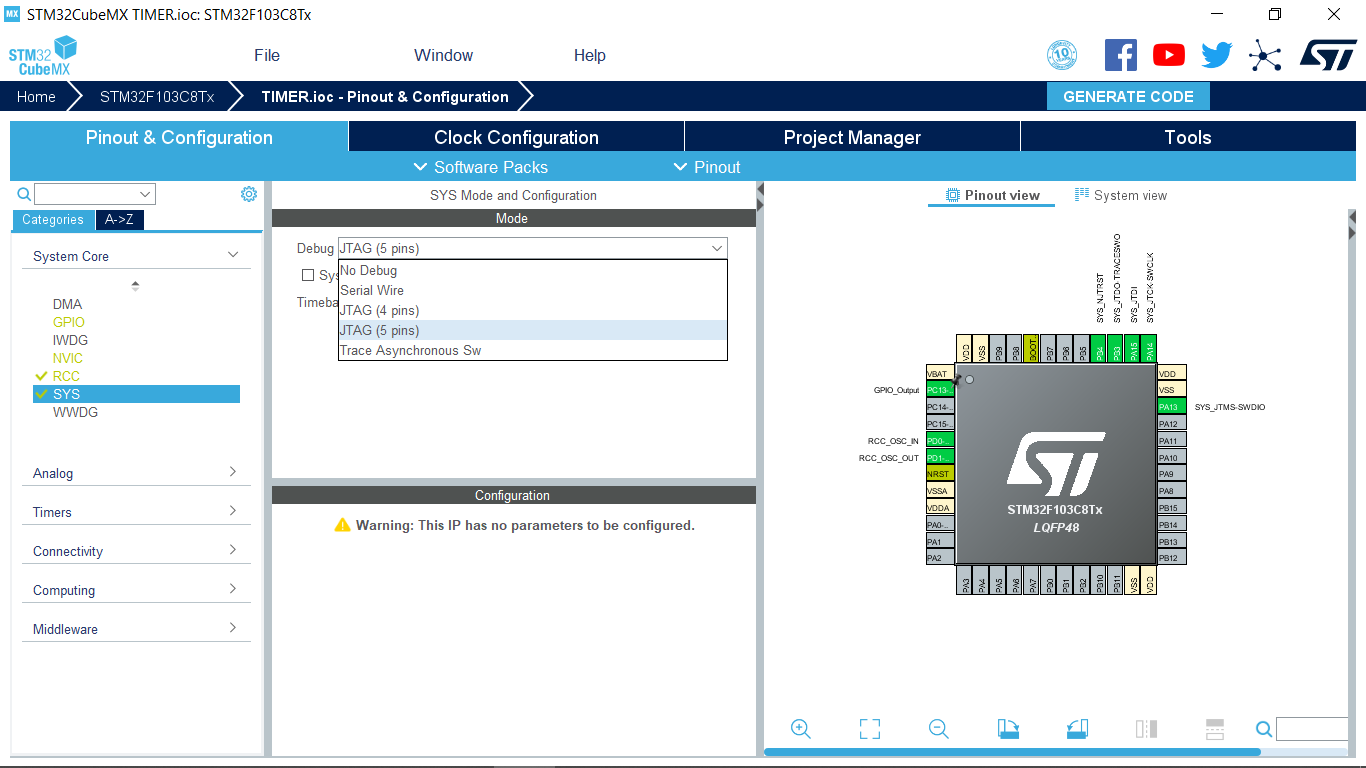
1. **Cách thiết lập chức năng trong phần mềm CubeMX:**

**Cấu hình mức độ 1:**

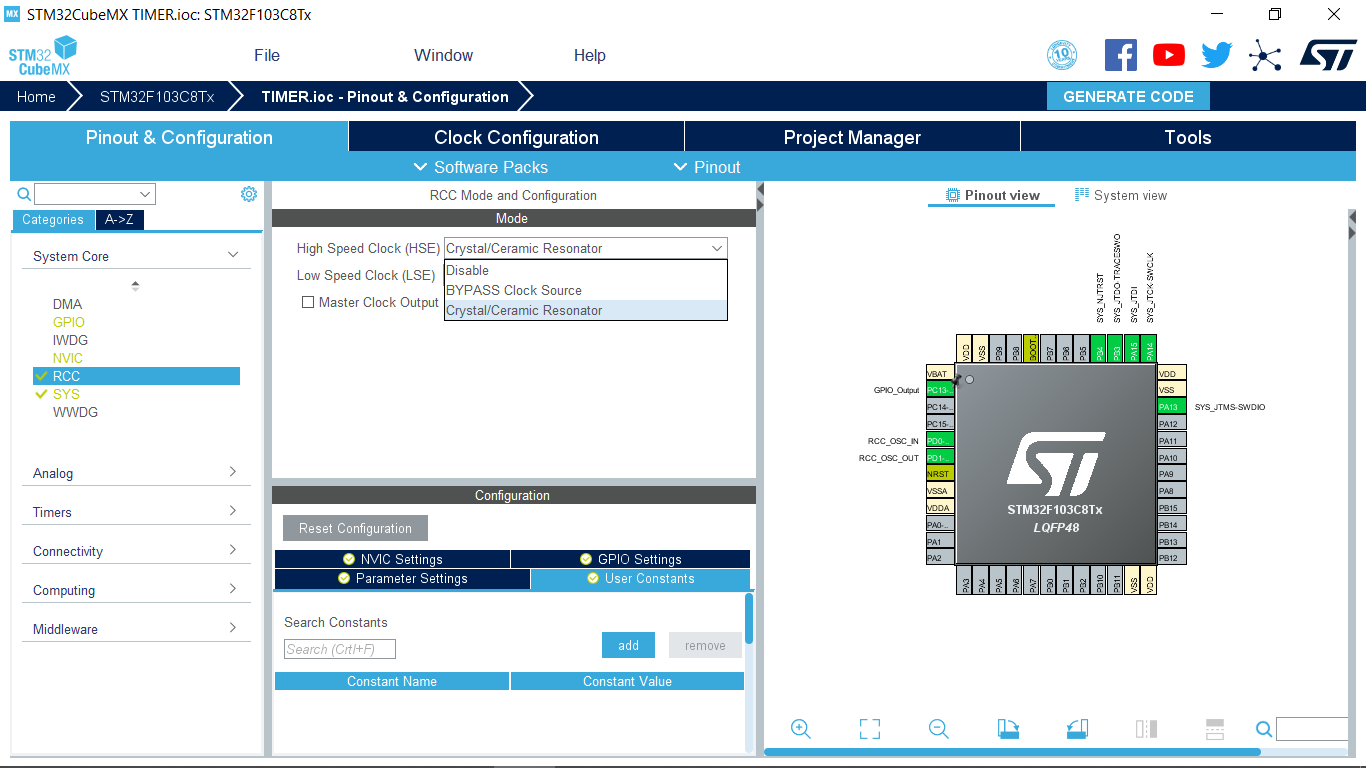
**Bước 1**: Khởi động phần mềm CubeMX, tạo New Project và chọn mã số STM32F103C8.



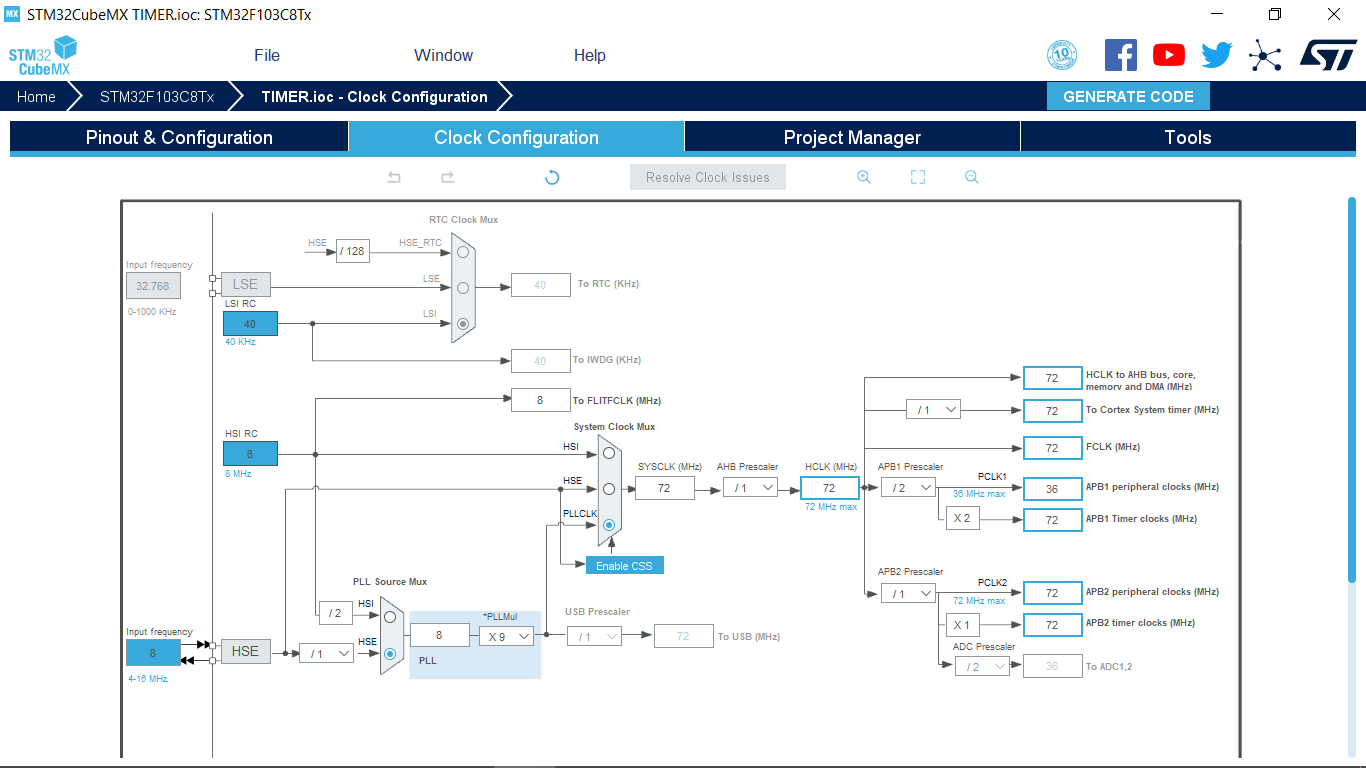
**Bước 2**: Chọn mạch nạp:



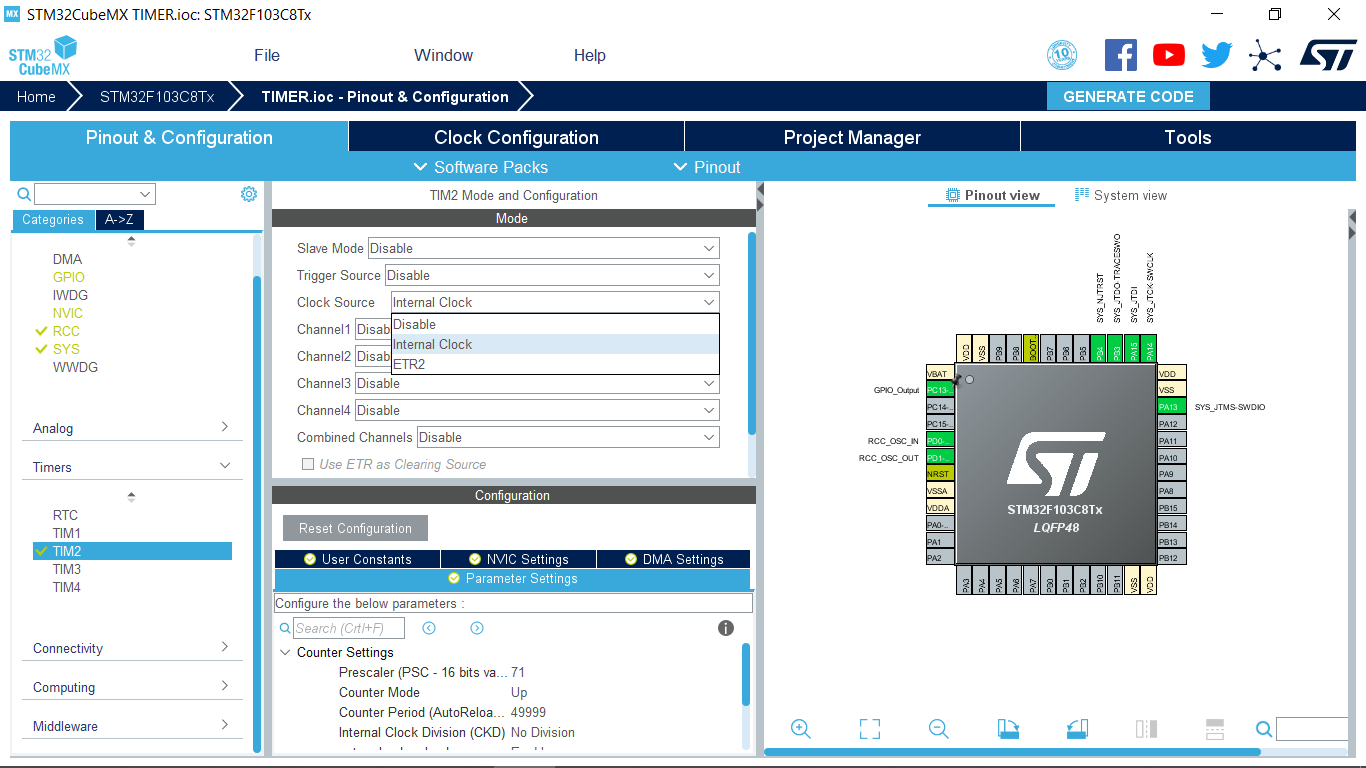
**Bước 3**: Chọn bộ giao động:



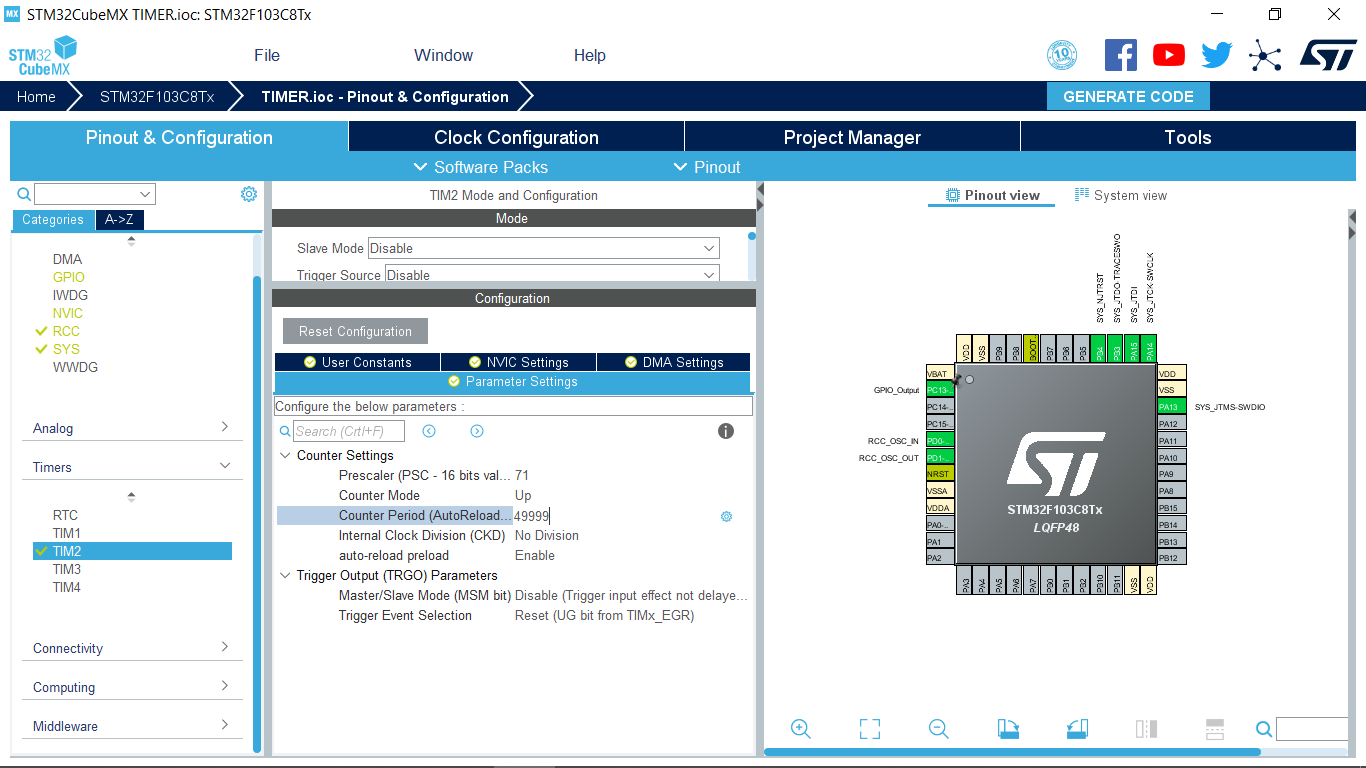
**Bước 4:** Chọn tần số giao động:



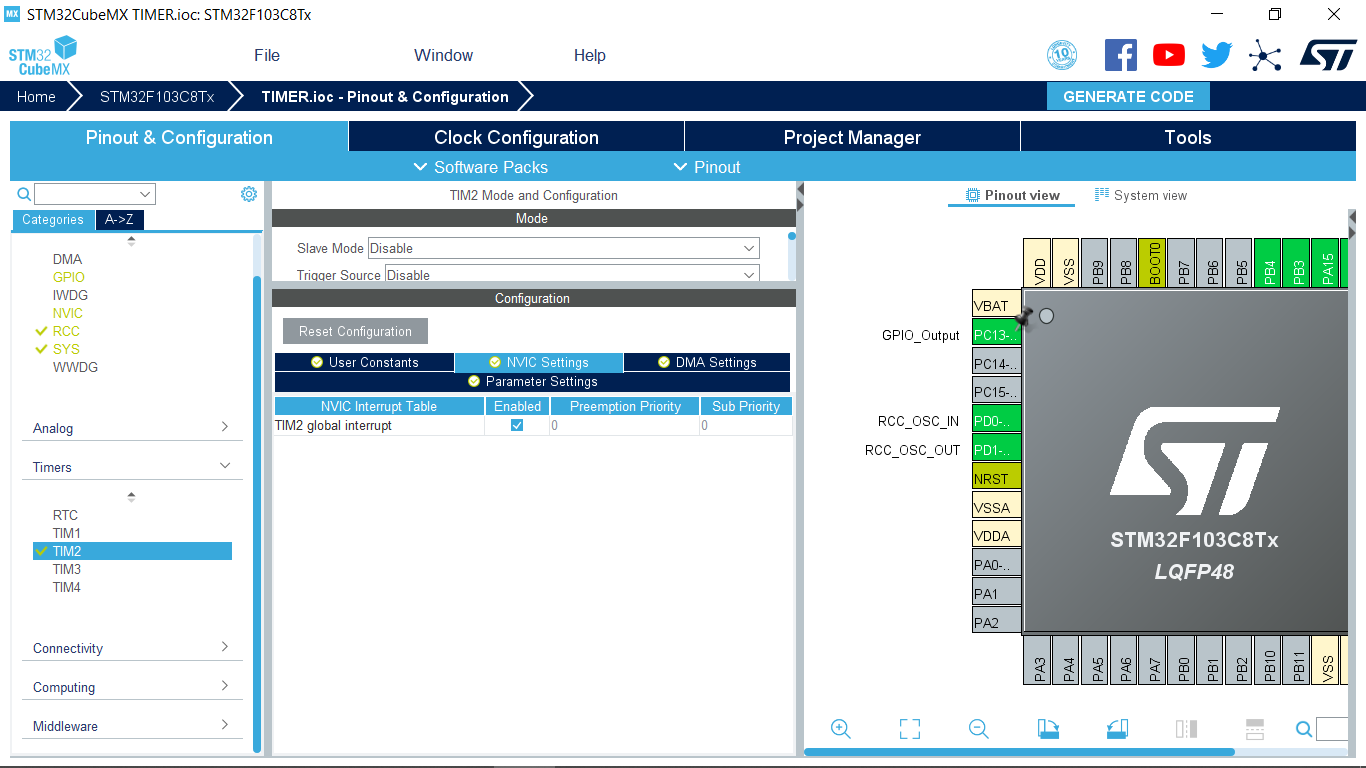
**Bước 5** : Cấu hình Timer2 ở chế độ Internal Clock



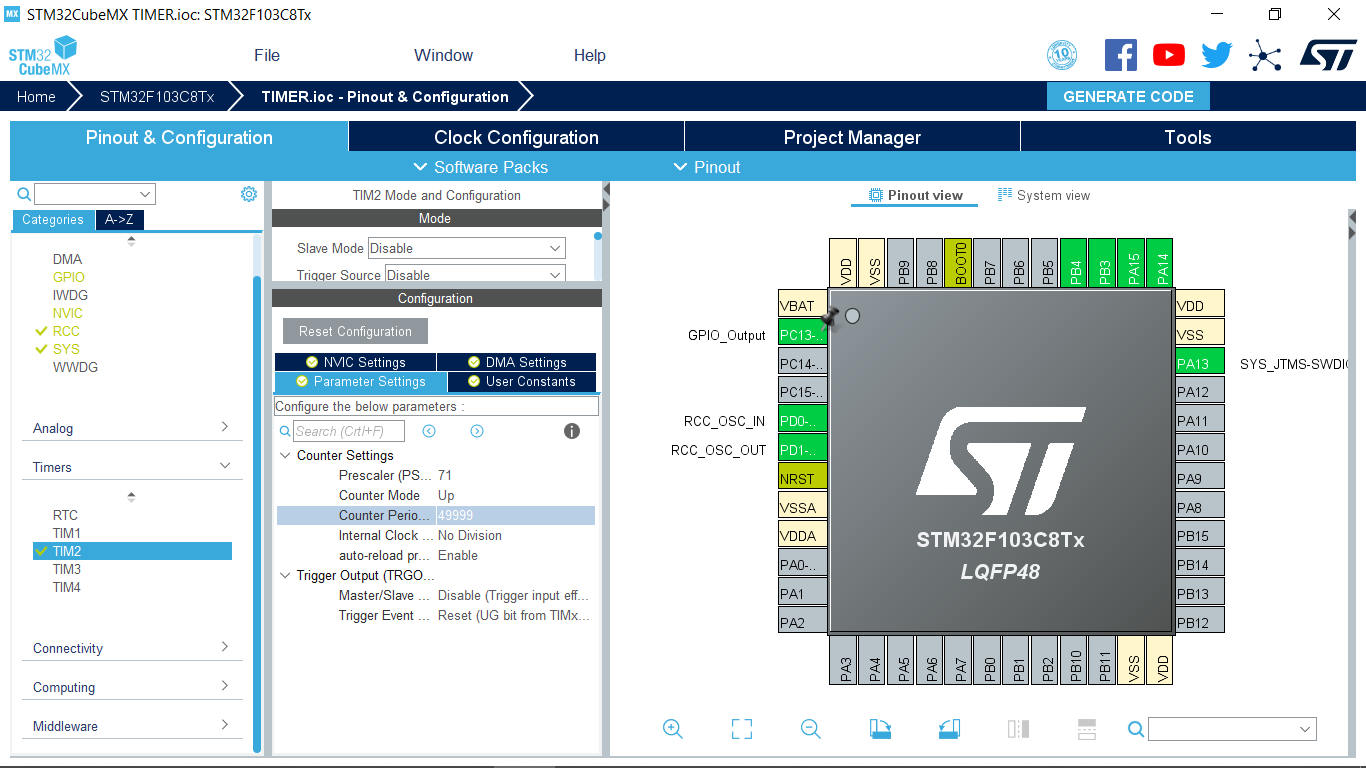
Đặt các thông số



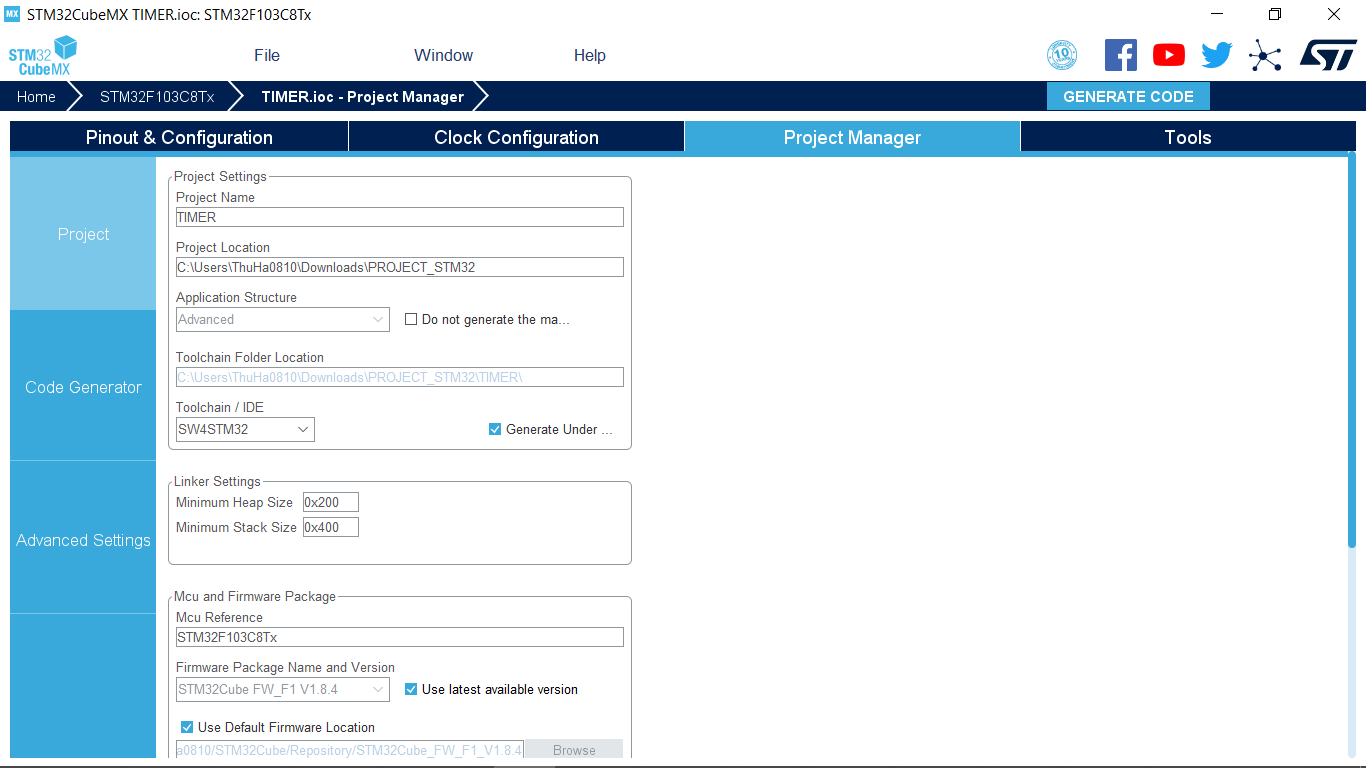
Chọn ngắt



**Bước 6:** Cấu hình chân PC13 là GPIO\_Output



**Bước 7**: Đặt tên cho Project và chọn nơi lưu trữ, chọn phần mềm viết chương trình



**Cấu hình mức độ 2:**

**Bước 1**: Khởi động phần mềm CubeMX, tạo New Project và chọn mã số STM32F103C8T6.

Graphical user interface, text, application

Description automatically generated

**Bước 2:** Chọn mạch nạp:

**Graphical user interface

Description automatically generated**

**Bước 3**: Chọn bộ giao động:

**Graphical user interface, application

Description automatically generated**

**Bước 4:** Chọn tần số giao động:

**Diagram

Description automatically generated**

**Bước 5** : Cấu hình Timer2

Graphical user interface

Description automatically generated

Chọn ngắt

Graphical user interface, application

Description automatically generated

**Bước 6:**  Cấu hình chân PC13 là GPIO\_Output

Graphical user interface

Description automatically generated

**Bước 7:** Đặt tên cho Project và chọn nơi lưu trữ, chọn phần mềm viết chương trình

**Graphical user interface, text, application, email

Description automatically generated**

**Cấu hình mức độ 3:**

**Bước 1**: Khởi động phần mềm CubeMX, tạo New Project và chọn mã số STM32F103C8T6.

Graphical user interface, text, application

Description automatically generated

**Bước 2**: Chọn mạch nạp:

Graphical user interface

Description automatically generated

**Bước 3**: Chọn bộ giao động:

**Graphical user interface

Description automatically generated**

**Bước 4:** Chọn tần số giao động:

Diagram

Description automatically generated

**Bước 5** : Cấu hình Timer2 ở chế độ Counter

Graphical user interface, application

Description automatically generated

Thông số cấu hình

Graphical user interface

Description automatically generated

Chân PA0 nối nút nhấn

Graphical user interface

Description automatically generated

Graphical user interface

Description automatically generated**Bước 6**: Cấu hình UART1

**Bước 7**: Đặt tên cho Project và chọn nơi lưu trữ, chọn phần mềm viết chương trình

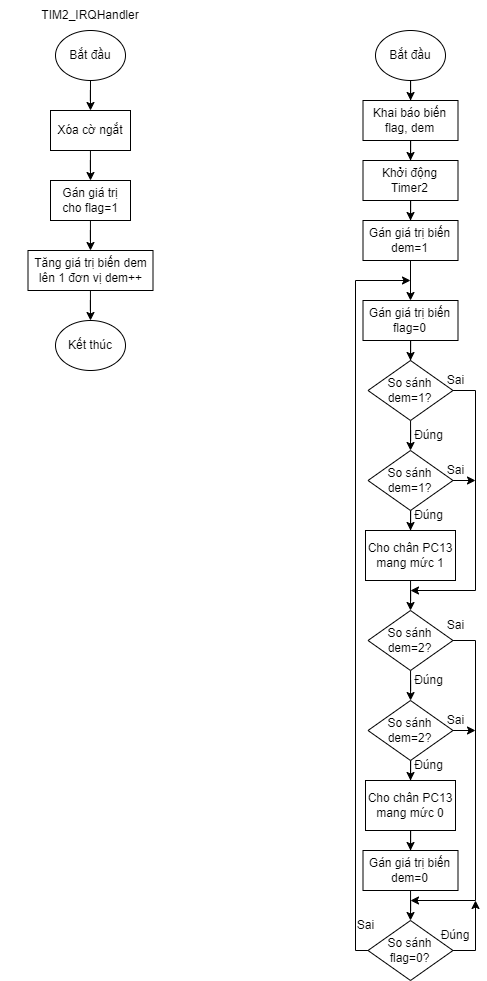
Graphical user interface, text, application

Description automatically generated

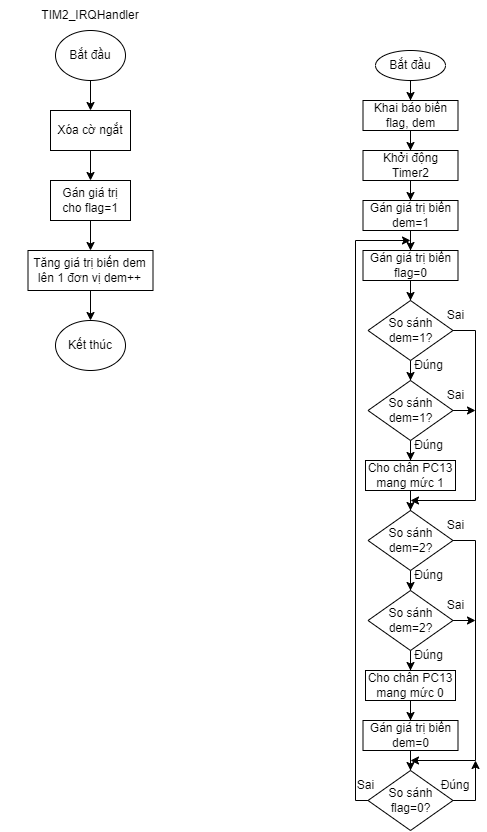
1. **Lưu đồ giải thuật:**

**Mức 1:**

**Chương trình ngắt:**

****

**Chương trình chính:**

****

**Mức 2:**

**Chương trình ngắt:**

**Diagram

Description automatically generated**

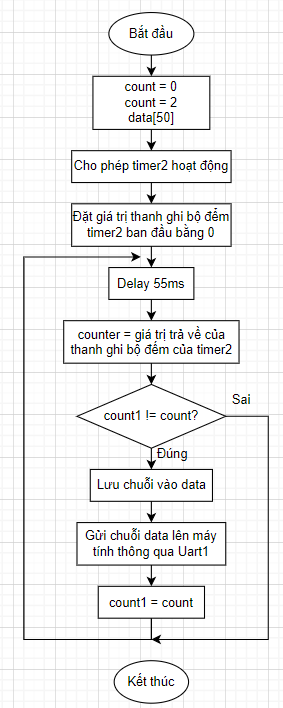
**Chương trình chính:**

**Diagram

Description automatically generated**

**Mức 3:**

**Chương trình chính:**

****

1. **Mã nguồn chương trình:**

**Mức 1:**

**Chương trình ngắt:**

/\* USER CODE BEGIN EV \*/

**extern** uint8\_t flag;

**extern** uint8\_t dem;

/\* USER CODE END EV \*/

...

**void** **TIM2\_IRQHandler**(**void**)

{

/\* USER CODE BEGIN TIM2\_IRQn 0 \*/

/\* USER CODE END TIM2\_IRQn 0 \*/

HAL\_TIM\_IRQHandler(&htim2);

/\* USER CODE BEGIN TIM2\_IRQn 1 \*/

//HAL\_GPIO\_TogglePin(GPIOC, GPIO\_PIN\_13);

flag=1;

dem++;

/\* USER CODE END TIM2\_IRQn 1 \*/

}

**Chương trình chính:**

/\* USER CODE BEGIN PV \*/

uint8\_t flag;

uint8\_t dem;

/\* USER CODE END PV \*/

...

**int** **main**(**void**)

{

/\* USER CODE BEGIN 1 \*/

/\* USER CODE END 1 \*/

/\* MCU Configuration--------------------------------------------------------\*/

/\* Reset of all peripherals, Initializes the Flash interface and the Systick. \*/

HAL\_Init();

/\* USER CODE BEGIN Init \*/

/\* USER CODE END Init \*/

/\* Configure the system clock \*/

SystemClock\_Config();

/\* USER CODE BEGIN SysInit \*/

/\* USER CODE END SysInit \*/

/\* Initialize all configured peripherals \*/

MX\_GPIO\_Init();

MX\_TIM2\_Init();

/\* USER CODE BEGIN 2 \*/

HAL\_TIM\_Base\_Start\_IT(&htim2);

dem=1;

/\* USER CODE END 2 \*/

/\* Infinite loop \*/

/\* USER CODE BEGIN WHILE \*/

**while** (1)

{

/\* USER CODE END WHILE \*/

/\* USER CODE BEGIN 3 \*/

flag=0;

**if**(dem==1)

{

**while**(dem==1)

{

HAL\_GPIO\_WritePin(GPIOC, GPIO\_PIN\_13, 1);

}

}

**if**(dem==2)

{

**while**(dem==2)

{

HAL\_GPIO\_WritePin(GPIOC, GPIO\_PIN\_13, 0);

}

dem=0;

}

//HAL\_GPIO\_TogglePin(GPIOC, GPIO\_PIN\_13);

**while**(flag==0);

//HAL\_Delay(500);

}

/\* USER CODE END 3 \*/

}

**Mức 2:**

**Chương trình chính:**

#include "main.h"

/\* Private includes ----------------------------------------------------------\*/

/\* USER CODE BEGIN Includes \*/

/\* USER CODE END Includes \*/

/\* Private typedef -----------------------------------------------------------\*/

/\* USER CODE BEGIN PTD \*/

/\* USER CODE END PTD \*/

/\* Private define ------------------------------------------------------------\*/

/\* USER CODE BEGIN PD \*/

/\* USER CODE END PD \*/

/\* Private macro -------------------------------------------------------------\*/

/\* USER CODE BEGIN PM \*/

/\* USER CODE END PM \*/

/\* Private variables ---------------------------------------------------------\*/

TIM\_HandleTypeDef htim2;

/\* USER CODE BEGIN PV \*/

uint8\_t my\_flag;

/\* USER CODE END PV \*/

…

int main(void)

{

/\* USER CODE BEGIN 1 \*/

/\* USER CODE END 1 \*/

/\* MCU Configuration--------------------------------------------------------\*/

/\* Reset of all peripherals, Initializes the Flash interface and the Systick. \*/

HAL\_Init();

/\* USER CODE BEGIN Init \*/

/\* USER CODE END Init \*/

/\* Configure the system clock \*/

SystemClock\_Config();

/\* USER CODE BEGIN SysInit \*/

/\* USER CODE END SysInit \*/

/\* Initialize all configured peripherals \*/

MX\_GPIO\_Init();

MX\_TIM2\_Init();

/\* USER CODE BEGIN 2 \*/

HAL\_TIM\_Base\_Start\_IT(&htim2);

/\* USER CODE END 2 \*/

/\* Infinite loop \*/

/\* USER CODE BEGIN WHILE \*/

while (1)

{

/\* USER CODE END WHILE \*/

/\* USER CODE BEGIN 3 \*/

my\_flag = 0;

HAL\_GPIO\_TogglePin(GPIOC, GPIO\_PIN\_13);

HAL\_Delay(700);

while(my\_flag == 0);

}

/\* USER CODE END 3 \*/

}

**Chương trình ngắt:**

#include "main.h"

#include "stm32f1xx\_it.h"

/\* Private includes ----------------------------------------------------------\*/

/\* USER CODE BEGIN Includes \*/

/\* USER CODE END Includes \*/

/\* Private typedef -----------------------------------------------------------\*/

/\* USER CODE BEGIN TD \*/

/\* USER CODE END TD \*/

/\* Private define ------------------------------------------------------------\*/

/\* USER CODE BEGIN PD \*/

/\* USER CODE END PD \*/

/\* Private macro -------------------------------------------------------------\*/

/\* USER CODE BEGIN PM \*/

/\* USER CODE END PM \*/

/\* Private variables ---------------------------------------------------------\*/

/\* USER CODE BEGIN PV \*/

extern uint8\_t my\_flag;

/\* USER CODE END PV \*/

…

void TIM2\_IRQHandler(void)

{

/\* USER CODE BEGIN TIM2\_IRQn 0 \*/

/\* USER CODE END TIM2\_IRQn 0 \*/

HAL\_TIM\_IRQHandler(&htim2);

/\* USER CODE BEGIN TIM2\_IRQn 1 \*/

my\_flag = 1;

/\* USER CODE END TIM2\_IRQn 1 \*/

}

**Mức 3:**

**Chương trình chính:**

/\* USER CODE END Header \*/

/\* Includes ------------------------------------------------------------------\*/

#include "main.h"

/\* Private includes ----------------------------------------------------------\*/

/\* USER CODE BEGIN Includes \*/

#include <stdio.h>

/\* USER CODE END Includes \*/

/\* Private typedef -----------------------------------------------------------\*/

/\* USER CODE BEGIN PTD \*/

/\* USER CODE END PTD \*/

/\* Private define ------------------------------------------------------------\*/

/\* USER CODE BEGIN PD \*/

/\* USER CODE END PD \*/

/\* Private macro -------------------------------------------------------------\*/

/\* USER CODE BEGIN PM \*/

/\* USER CODE END PM \*/

/\* Private variables ---------------------------------------------------------\*/

TIM\_HandleTypeDef htim2;

UART\_HandleTypeDef huart1;

/\* USER CODE BEGIN PV \*/

uint8\_t count = 0;

uint8\_t count1 = 2;

uint8\_t data[50];

/\* USER CODE END PV \*/

…

/\* Private function prototypes -----------------------------------------------\*/

void SystemClock\_Config(void);

static void MX\_GPIO\_Init(void);

static void MX\_TIM2\_Init(void);

static void MX\_USART1\_UART\_Init(void);

/\* USER CODE BEGIN PFP \*/

/\* USER CODE END PFP \*/

/\* Private user code ---------------------------------------------------------\*/

/\* USER CODE BEGIN 0 \*/

/\* USER CODE END 0 \*/

/\*\*

\* @brief The application entry point.

\* @retval int

\*/

int main(void)

{

/\* USER CODE BEGIN 1 \*/

/\* USER CODE END 1 \*/

/\* MCU Configuration--------------------------------------------------------\*/

/\* Reset of all peripherals, Initializes the Flash interface and the Systick. \*/

HAL\_Init();

/\* USER CODE BEGIN Init \*/

/\* USER CODE END Init \*/

/\* Configure the system clock \*/

SystemClock\_Config();

/\* USER CODE BEGIN SysInit \*/

/\* USER CODE END SysInit \*/

/\* Initialize all configured peripherals \*/

MX\_GPIO\_Init();

MX\_TIM2\_Init();

MX\_USART1\_UART\_Init();

/\* USER CODE BEGIN 2 \*/

HAL\_TIM\_Base\_Start(&htim2);

\_\_HAL\_TIM\_SET\_COUNTER(&htim2, 0);

/\* USER CODE END 2 \*/

/\* Infinite loop \*/

/\* USER CODE BEGIN WHILE \*/

while (1)

{

/\* USER CODE END WHILE \*/

/\* USER CODE BEGIN 3 \*/

HAL\_Delay(55);

count = \_\_HAL\_TIM\_GET\_COUNTER(&htim2);

if(count1 != count)

{

sprintf(&data[0], "Dem: %d\r\n", count);

HAL\_UART\_Transmit(&huart1, &data[0], sizeof(data), 10);

count1 = count;

}

}

/\* USER CODE END 3 \*/

}

1. **Video minh chứng:**

Link youtube: