**HƯỚNG DẪN SỬ DỤNG THƯ VIỆN STM32F103C8T6**

1. **Tổng quan:**

* Thư viện còn thiếu phần giải thích nội dung một số hàm.
* Các module cần được test lại do vẫn còn sửa chữa nội dung.
* Có vấn đề ở module analog.

1. **Tình trạng hoàn thành thư viện:**
2. **Module GPIO:**

+ pinMode, digitalToggle, digitalRead, digitalWrite.

*\* có thể thêm hàm đọc trạng thái của cả 1 GPIO( 16 chân).*

1. **ADC**

+ analogInit, analogRead, ADC\_DMA.

* analogDeinit

*\* analogInit cấu hình toàn bộ các chân ADC, cần sửa lại theo từng chân.*

*\* bổ sung các mode ở dong 77 file stm32f10x\_gpio.h*

1. **PWM:**

+ PWMInit, AnalogWrite.

\*  *chưa test trên từng chân.*

1. **Ngắt ngoài:**

+ attachInterrupt.

- detachInterrupt

\*  *chưa test trên từng chân.*

*\* Hoàn thành ngắt ngoài cho các chân pin 0-4, chưa hoàn thành pin 5-15*

1. **Timer:**

+ Timer2Init, millis.

* micros.

*\* chưa xong các timer 0,1,3,4.*

1. **USART1(Serial)**

+ begin, end, ngắt nhận USART, print, println, write, read.

* Available, peek, flush, availableForWrite,

\* *không hiểu tác dụng của 1 số hàm trên, cần hoàn thiện các module USART khác*

1. **I2C1:**

+ begin, end, , beginTransmission,endTransmission, requestFrom, write, available, read, Slave ngắt yêu cầu gởi đi(onrequest), Slave ngắt yêu cầu gởi đến

* setClock

\* chưa xong các module I2C2,3…

1. **SPI1**

+ Begin, transfer, setDataMode, setClockDivider, read, attachInterrupt, detachInterrupt.

* setBitOrder, setClockDivider.

\* Chưa xong các module còn lại.

1. **Hướng dẫn sử dụng thư viện:**
2. **GPIO:**

\* Thư viện GPIO chứa các hàm dùng để điều khiển I/O cho các chân vđk. Việc điều khiển bao gồm:

* PinMode: cấu hình chế độ vào ra cho các chân.
* digitalRead: Đọc trạng thái 1 chân digital.
* digitalWrite: xuất trạng thái 1 chân ( mức 1 hoặc 0).
* digitalToggle: đảo trạng thái 1 chân (từ 1 qua 0 hoặc ngược lại).

1. **PinMode:** cấu hình chế độ vào ra cho các chân.

**\* Cú pháp: void** **pinMode**(GPIO\_TypeDef \*GPIOx,uint8\_t PINx,uint8\_t GPIO\_MODE\_)

* GPIOx: GPIOA, GPIOB, GPIOC.
* PINx: số thứ tự chân của GPIO cần cấu hình: 0~15 tùy theo GPIO.
* GPIO\_MODE\_: Chế độ hoạt động cần thiết lập.

Các chế độ của GPIO hiện tại:

**#define** INPUT 0x04 // chế độ nhận digital.

**#define** OUTPUT 0x10 // chế độ xuất digital.

**#define** INPUT\_PULLUP 0x48 // chế độ nhận digital kéo lên Vcc.

**#define** INPUT\_PULLDOWN 0x28 // chế độ nhận digital kéo xuống GND.

**#define** AF\_PP 0x18 // có thế nhập/ xuất nhưng ko có trở kéo lên/xuống nếu nhận digital, khi xuất thì theo kiểu Push/Pull.

**#define** AF\_OD 0x1C // có thế nhập/ xuất nhưng ko có trở kéo lên/xuống nếu nhận digital, khi xuất thì theo kiểu Open-Drain.

\* Giải thích thêm về PP và OD:

* Output open-drain: cấu hình chân I/O là ngõ ra, khi output control = 0 thì N-MOS sẽ dẫn, chân I/O sẽ nối VSS, còn khi output control = 1 thì P-MOS và N-MOS đều không dẫn, chân I/O được để nổi.
* Output push-pull: cấu hình chân I/O là ngõ ra, khi output control = 0 thì N-MOS sẽ dẫn, chân I/O sẽ nối VSS, còn khi output control = 1 thì P-MOS dẫn, chân I/O được nối VDD.

1. **digitalRead:** Đọc trạng thái 1 chân digital.

**\* Cú pháp:** uint8\_t **digitalRead**(GPIO\_TypeDef\* GPIOx, uint8\_t PINx)

* GPIOx: GPIOA, GPIOB, GPIOC.
* PINx: số thứ tự chân của GPIO cần cấu hình: 0~15 tùy theo GPIO.
* Giá trị trả về (uint8\_t): 0 nếu pin ở mức thấp, 1 nếu pin ở mức cao.

1. **digitalWrite:** xuất trạng thái 1 chân ( mức 1 hoặc 0) với các chân ở mode Output.

**\* Cú pháp:**

**void** **digitalWrite**( GPIO\_TypeDef\* GPIOx, uint8\_t PINx, uint8\_t GPIO\_STATE\_);

* GPIOx: GPIOA, GPIOB, GPIOC.
* PINx: số thứ tự chân của GPIO cần cấu hình: 0~15 tùy theo GPIO.
* GPIO\_STATE\_: trạng thái của chân digital: HIGH/LOW. Hoặc có thể 1/0.
* Giá trị trả về: không có trả về.

1. **digitalToggle:** đảo trạng thái 1 chân với các chân ở mode Output.

**\* Cú pháp:**

**void** **digitalToggle**( GPIO\_TypeDef\* GPIOx, uint8\_t PINx)

* GPIOx: GPIOA, GPIOB, GPIOC.
* PINx: số thứ tự chân của GPIO cần cấu hình: 0~15 tùy theo GPIO.
* Giá trị trả về: không có trả về.

1. **ADC:**

\* Thư viện ADC chứa các hàm dùng để điều khiển I/O cho các chân vđk. Việc điều khiển bao gồm:

* analogInit: cấu hình tất cả các chân analog.
* analogRead: Đọc trạng thái 1 chân analog.
* ADC\_DMA\_Init: cài đặt chế độ đọc analog sử dụng DMA.
  1. **analogInit:** cấu hình tất cả các chân analog #cần thay đổi.

**\* Cú pháp: void** **analogInit**()

* Không có tham số cũng như kiểu trả về.
  1. **analogRead:** Đọc trạng thái 1 chân analog.

**\* Cú pháp:** uint16\_t **analogRead**(uint8\_t channel)

* channel: