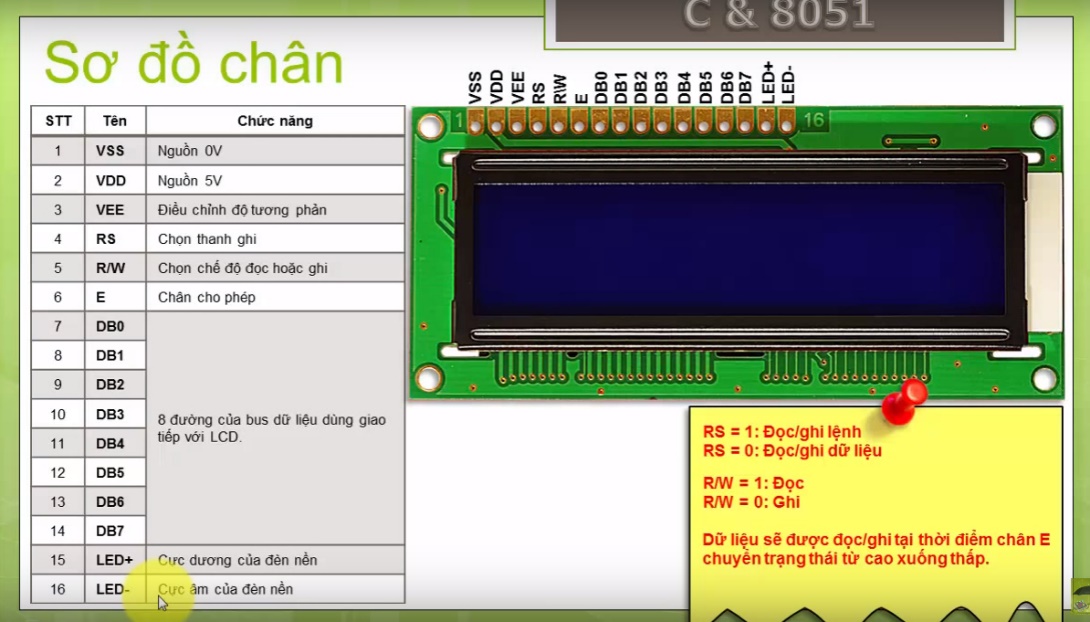
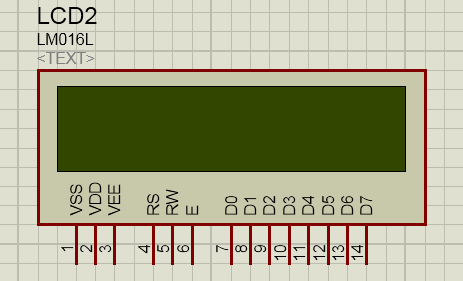
## LCD1602

### 1, Giới thiệu về LCD1602

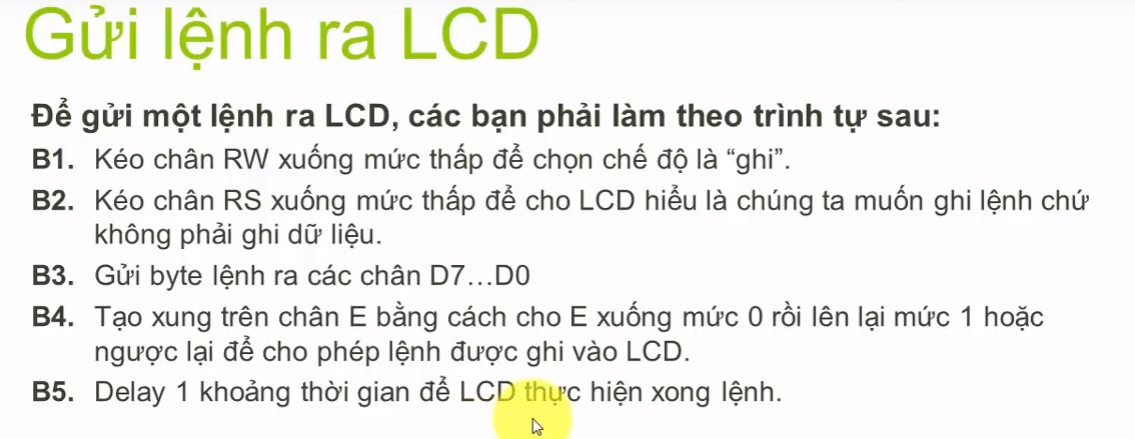
Chúng ta đến với một phần rất quan trọng: Hiển thị thông tin ra LCD. LCD là một màn hình nhiều điểm ảnh, có thể coi là một Led ma trận dạng lớn, tuy nhiên chúng ta không cần đi vào điều khiển từng Pixel hay là từng chấm nhỏ như trong Phần 1 mà chúng ta sẽ điều khiển qua lệnh, con trỏ… để hiển thị thông tin một cách dễ dàng hơn. Có nhiều loại LCD, trong bài này chúng ta dùng loại đơn giản 16x2. Trước tiên chúng ta tìm hiểu về cầu tạo của nó.



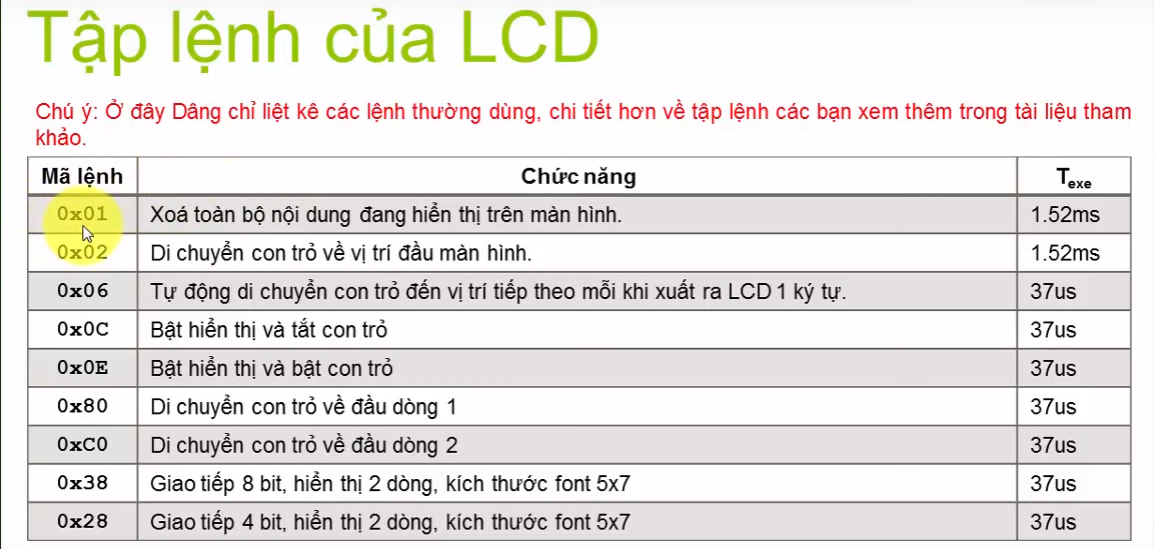
Dưới đây là hình mô phỏng, các em tìm nó với từ khóa “lcd 16x2”.



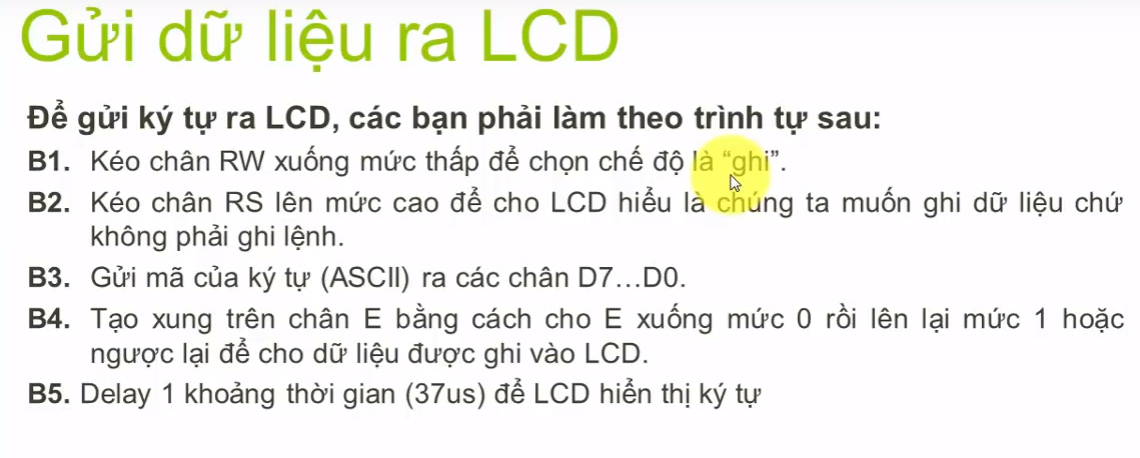
Tiếp theo chúng ta học về cách kết nối, lập trình để sử dụng LCD. Trước tiên chúng ta phải gửi lệnh vào cho LCD biết nó phải làm gì. Cách gửi lệnh như sau:

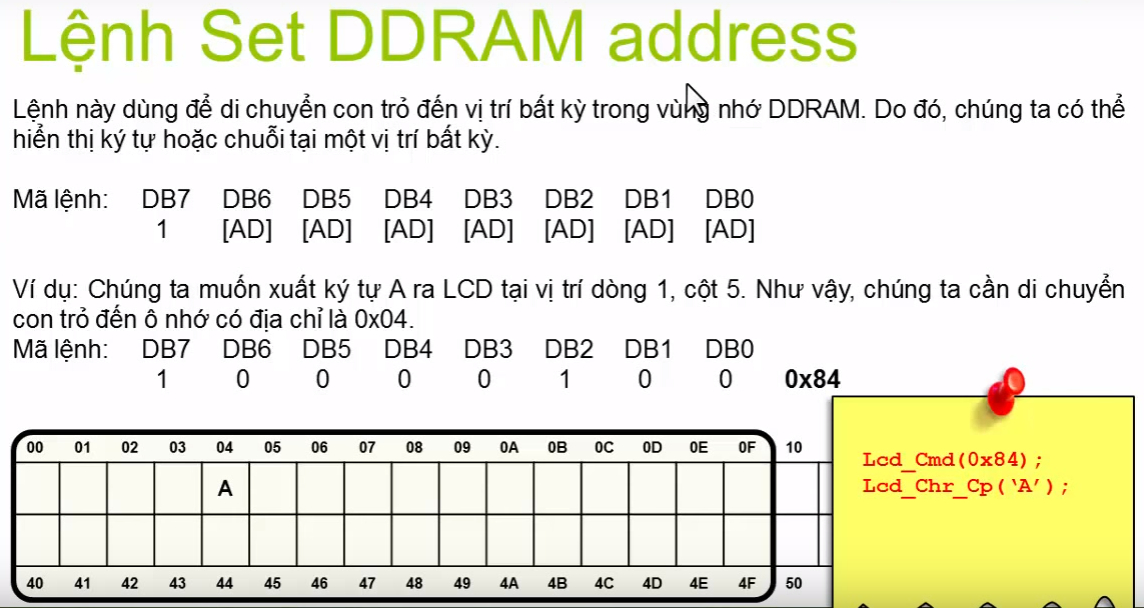


Các em có thể xem một số lệnh sau đây. Những lệnh này gọi là lệnh khởi tạo LCD.

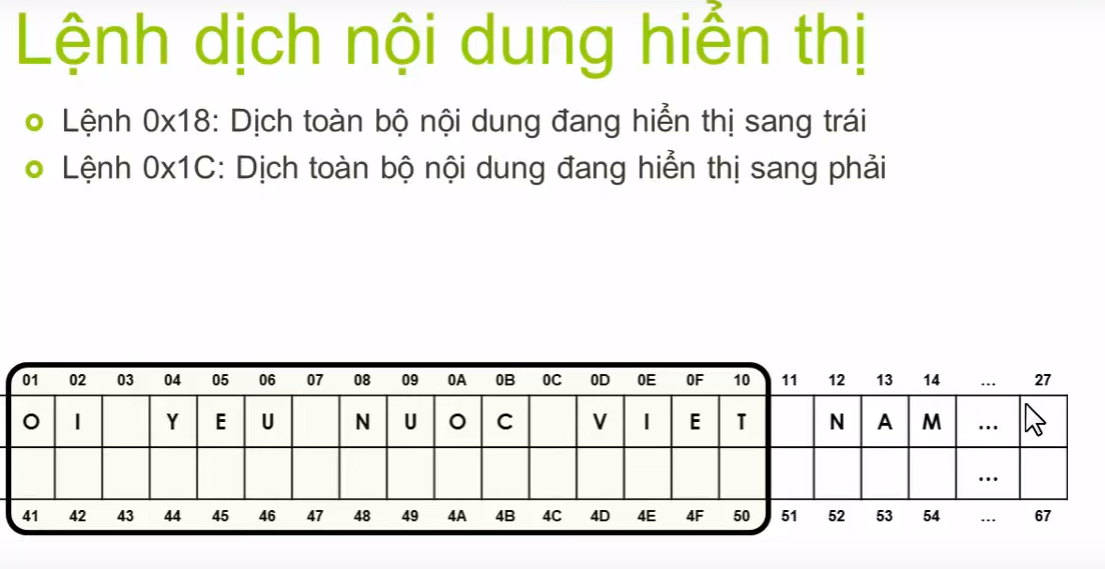


Sau khi gửi các lệnh vào LCD (qua các chân D0 – D7) để nó biết sẽ phải làm gì (bật màn hình, hiển thị hay tắt con trỏ, di chuyển con trỏ …), chúng ta sẽ phải gửi lệnh chứa các kí tự mà chúng ta muốn hiển thị. Cách gửi dữ liệu ra cũng tương tự như cách gửi lệnh, các em có thể xem trong hình sau:

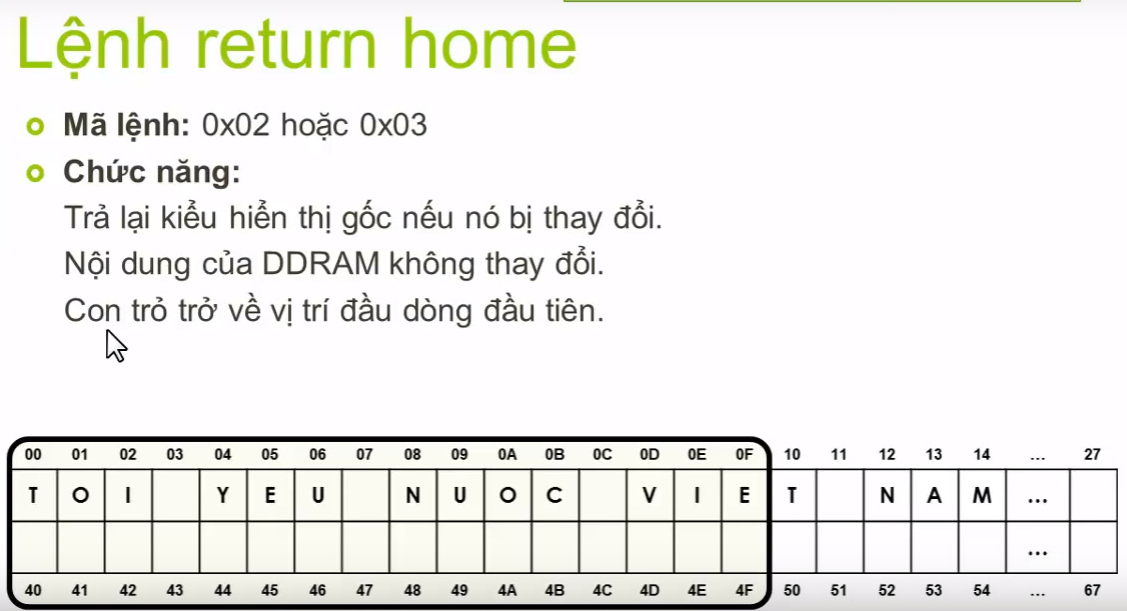




Sau khi hiển thị một kí tự tại một vị trí có sẵn, chúng ta có thể tiến đến bước xa hơn, dịch chuyển kí tự mà chúng ta muốn (dịch trái, dịch phải) với câu lệnh như hình dưới.



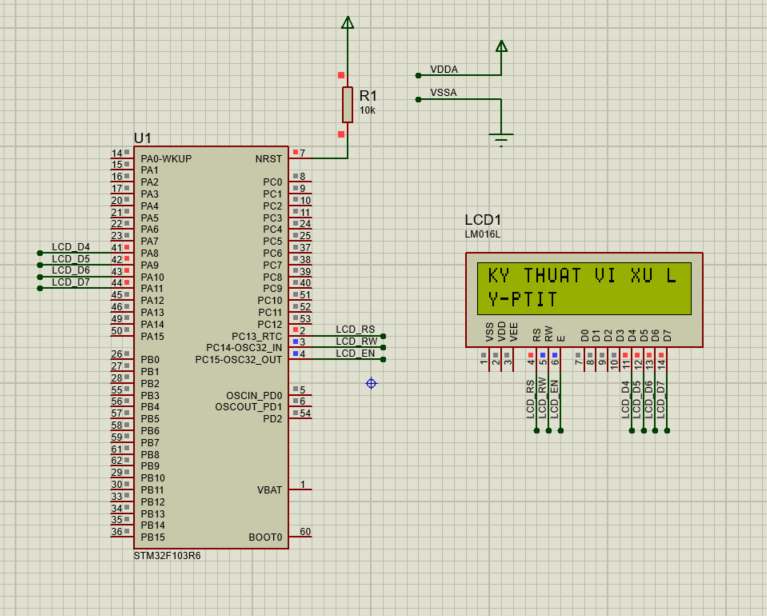
Cuối cùng, khi dòng chữ đang chạy, các em muốn dừng không cho chạy nữa, trở lại vị trí như ban đầu, có thể dùng lệnh sau:



### 2, Hướng dẫn thực hành với KIT STM32F103C8T6

#### Bài 1: Gửu kí tự lên màn hình LCD1602

***a, Mô phỏng trên proteus***



**Sơ đồ nối chân:**

LCD\_RS PC13

LCD\_RW PC14

LCD\_EN PC15

LCD\_D4 PA8

LCD\_D5 PA9

LCD\_D6 PA10

LCD\_D7 PA11

**\*Lưu ý: Các em add thư viện cần thiết vào project như delay.h, lcd.h**

***b, code***

**File main.c:**

#include "stm32f10x\_gpio.h"

#include "stm32f10x\_rcc.h"

#include "lcd16x2.h"

#include "delay.h"

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

**{**

// Delay initialization

DelayInit**();**

// LCD initialization

LCD\_Init**();**

**while(**1**){**

LCD\_Gotoxy**(**0**,**0**);**

LCD\_Puts**(**"KY THUAT VI XU"**);**

LCD\_Gotoxy**(**0**,**1**);**

LCD\_Puts**(**"LY - PTIT"**);**

DelayMs**(**500**);**

**}**

**}**

**File delay.c**

#include "delay.h"

// For store tick counts in us

static \_\_IO uint32\_t usTicks**;**

// SysTick\_Handler function will be called every 1 us

void SysTick\_Handler**()**

**{**

**if** **(**usTicks **!=** 0**)**

**{**

usTicks**--;**

**}**

**}**

void DelayInit**()**

**{**

// Update SystemCoreClock value

SystemCoreClockUpdate**();**

// Configure the SysTick timer to overflow every 1 us

SysTick\_Config**(**SystemCoreClock **/** 1000000**);**

**}**

void DelayUs**(**uint32\_t us**)**

**{**

// Reload us value

usTicks **=** us**;**

// Wait until usTick reach zero

**while** **(**usTicks**);**

**}**

void DelayMs**(**uint32\_t ms**)**

**{**

// Wait until ms reach zero

**while** **(**ms**--)**

**{**

// Delay 1ms

DelayUs**(**1000**);**

**}**

**}**

**File delay.h**

#ifndef \_\_DELAY\_H

#define \_\_DELAY\_H

#ifdef \_\_cplusplus

extern "C" {

#endif

#include "stm32f10x.h"

void DelayInit(void);

void DelayUs(uint32\_t us);

void DelayMs(uint32\_t ms);

#ifdef \_\_cplusplus

}

#endif

#endif

**File LCD.c**

#include "lcd16x2.h"

#define LCD\_RS GPIO\_Pin\_13

#define LCD\_RW GPIO\_Pin\_14

#define LCD\_EN GPIO\_Pin\_15

#define LCD\_D4 GPIO\_Pin\_8

#define LCD\_D5 GPIO\_Pin\_9

#define LCD\_D6 GPIO\_Pin\_10

#define LCD\_D7 GPIO\_Pin\_11

void GPIO\_LCD\_Config**(**void**)**

**{**

RCC\_APB2PeriphClockCmd**(**RCC\_APB2Periph\_GPIOA**|**RCC\_APB2Periph\_GPIOC**,** ENABLE**);**

GPIO\_InitTypeDef GPIO\_LCD\_InitStruction**;**

/\*Configure GPIO pin Output Level \*/

GPIO\_WriteBit**(**GPIOA**,** LCD\_D4**|**LCD\_D5**|**LCD\_D5**|**LCD\_D7**,** 0**);**

GPIO\_WriteBit**(**GPIOC**,** LCD\_RS**|**LCD\_RW**|**LCD\_EN**,** 0**);**

GPIO\_LCD\_InitStruction**.**GPIO\_Mode **=** GPIO\_Mode\_Out\_PP**;**

GPIO\_LCD\_InitStruction**.**GPIO\_Pin **=** LCD\_D4**|**LCD\_D5**|**LCD\_D6**|**LCD\_D7**;**

GPIO\_LCD\_InitStruction**.**GPIO\_Speed **=** GPIO\_Speed\_10MHz**;**

GPIO\_Init**(**GPIOA**,** **&**GPIO\_LCD\_InitStruction**);**

GPIO\_LCD\_InitStruction**.**GPIO\_Pin **=** LCD\_EN**|**LCD\_RS**|**LCD\_RW**;**

GPIO\_Init**(**GPIOC**,** **&**GPIO\_LCD\_InitStruction**);**

**}**

void LCD\_Enable**(**void**)**

**{**

GPIO\_SetBits**(**GPIOC**,** LCD\_EN**);**

DelayMs**(**1**);**

GPIO\_ResetBits**(**GPIOC**,** LCD\_EN**);**

DelayMs**(**1**);**

**}**

void LCD\_Send4Bit**(**unsigned char Data**)**

**{**

GPIO\_WriteBit**(**GPIOA**,** LCD\_D4**,** Data **&** 0x01**);**

GPIO\_WriteBit**(**GPIOA**,** LCD\_D5**,** **(**Data**>>**1**)&**1**);**

GPIO\_WriteBit**(**GPIOA**,** LCD\_D6**,** **(**Data**>>**2**)&**1**);**

GPIO\_WriteBit**(**GPIOA**,** LCD\_D7**,** **(**Data**>>**3**)&**1**);**

**}**

void LCD\_SendCommand**(**unsigned char command**)**

**{**

LCD\_Send4Bit**(**command **>>** 4**);**

LCD\_Enable**();**

LCD\_Send4Bit**(**command**);**

LCD\_Enable**();**

**}**

void LCD\_Clear**()**

**{**

LCD\_SendCommand**(**0x01**);**

DelayUs**(**10**);**

**}**

void LCD\_Init**()**

**{**

GPIO\_LCD\_Config**();**

LCD\_Send4Bit**(**0x00**);**

GPIO\_WriteBit**(**GPIOC**,** LCD\_RS**,** 0**);**

LCD\_Send4Bit**(**0x03**);**

LCD\_Enable**();**

LCD\_Enable**();**

LCD\_Enable**();**

LCD\_Send4Bit**(**0x02**);**

LCD\_Enable**();**

LCD\_SendCommand**(**0x28**);** // giao thuc 4 bit, hien thi 2 hang, ki tu 5x8

LCD\_SendCommand**(**0x0C**);** // cho phep hien thi man hinh

LCD\_SendCommand**(**0x06**);** // tang ID, khong dich khung hinh

LCD\_SendCommand**(**0x01**);** // xoa toan bo khung hinh

**}**

void LCD\_Gotoxy**(**unsigned char x**,** unsigned char y**)**

**{**

unsigned char address**;**

**if(**y **==** 0**)**address**=(**0x80**+**x**);**

**else** **if(**y **==** 1**)** address**=(**0xc0**+**x**);**

LCD\_SendCommand**(**address**);**

**}**

void LCD\_PutChar**(**unsigned char Data**)**

**{**

GPIO\_SetBits**(**GPIOC**,** LCD\_RS**);**

LCD\_SendCommand**(**Data**);**

GPIO\_ResetBits**(**GPIOC**,** LCD\_RS**);**

**}**

void LCD\_Puts**(**char **\***s**)**

**{**

**while** **(\***s**)**

**{**

LCD\_PutChar**(\***s**);**

s**++;**

**}**

**}**

**File LCD.h**

#ifndef \_\_LCD\_H

#define \_\_LCD\_H

#ifdef \_\_cplusplus

extern "C" **{**

#endif

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

#include "stm32f10x.h"

#include "delay.h"

void GPIO\_LCD\_Config**(**void**);**

void LCD\_Enable**(**void**);**

void LCD\_Send4Bit**(**unsigned char Data**);**

void LCD\_SendCommand**(**unsigned char command**);**

void LCD\_Clear**();**

void LCD\_Init**();**

void LCD\_Gotoxy**(**unsigned char x**,** unsigned char y**);**

void LCD\_PutChar**(**unsigned char Data**);**

void LCD\_Puts**(**char **\***s**);**

#ifdef \_\_cplusplus

**}**

#endif

#endif /\* \_\_MISC\_H \*/

Add hết các file cần thiết vào và chạy chương trình

## III, Bài tập

**Bài tập 1:** Hiển thị 2 dòng chữ - Số thứ tự nhóm và Họ và tên một bạn trong nhóm.

**Bài tập 2:** Hiển thị Tên 2 bạn ở vị trí chính giữa màn hình, dòng 1 và dòng 2.

**Bài tập 3:** Dịch phải và dịch trái một đoạn text dài hơn 16 kí tự.