(Soạn thảo bài đăng)

Đây là phiên bản nâng cấp mới nhất tính từ ngày 16/12/2016. V2.0.1

LCD chip st7565 có kích thước màn hình 128x64 pixel, là loại lcd graphic đa dụng cho nhiều dự án.

Ở nước ta việc mua mới lcd này khá khó do không được ưa chuộng, sở dĩ có chuyện này có lẽ là do thiếu các tài liệu và hướng dẫn liên quan.

Hiện lcd loại này chỉ có thể vọc từ các điện thoại homephone hoặc mua cũ ..

Nếu bạn đang có trong tay lcd này, thì bài viết dưới sẽ giúp bạn sử dụng nó với arduino.

A: CHUẨN BỊ

1. 1 arduino chip atm328 trở lên, ở đây mình chọn arduino uno r3
2. 1 màn lcd homephone st7565 10 chân, hiện bạn cũng có thể mua cũ nó trên các website bán hang.
3. Ic ghép tầng chuyển tiếp (hạ áp) 4050 làm cầu giao tiếp arduino với lcd.
4. Nếu không có ic, bạn cần có 5 điện trở 10k và 5 điện trở 4,7k;
5. Và các đồ dùng cần thiết cho ráp mạch: dây điện, máy hàn…

B: TÌM HIỂU SƠ QUA

Cùng so sánh với loại lcd đang phổ biến nhất KS0108

|  |  |
| --- | --- |
| Ks0108-128x64 | St7565-128x64 |
| Ghép 2 ic ks0108,  Có 2x512=1024 bytes ram cho 2 ic, dùng để lưu dữ liệu điểm ảnh. Có thể đọc và ghi dữ liệu. | Ic điều khiển st7565  Không có bộ nhớ rời( chỉ hiện lên thôi);  Chỉ ghi. Phải lấy 1024 bytes ram từ arduino để làm bộ nhớ đệm. |
| Tương thích với các arduio đời đầu. | Arduino chip atm328 trở lên |
| 18 pin (1 pin tương phản, 3 pin điều khiển, 8 pin dữ liệu song song, 2 pin chọn chip, 2 pin nguồn, 1 pin reset, 1 pin Vee) | 7 pin ( 2 pin nguồn, 5 pin giao tiếp Spi) |
| Điện áp nguồn: 5v  Điện áp mức logic: 0-5v | Điện áp nguồn: 3,5v  Điện áp mức logic: 0-3,3v |
|  |  |
|  |  |

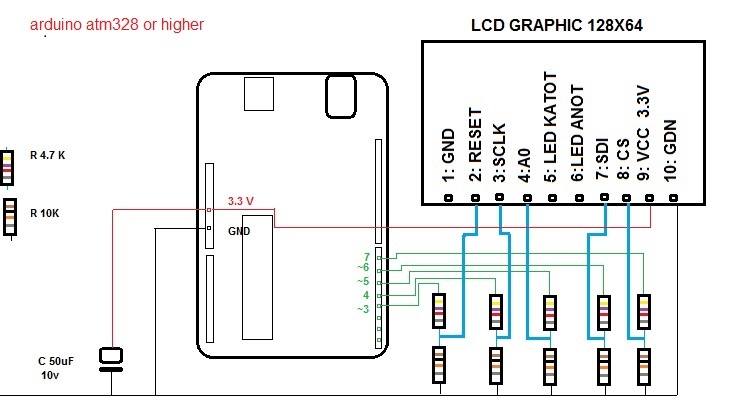
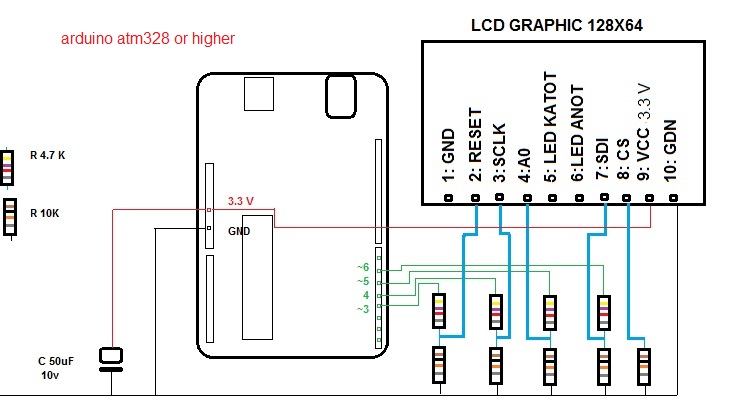
Trọng tâm bài viết là chia sẻ mã nguồn, hãy tìm hiểu thêm về lcd này tại các trang web sau:

<http://www.ladyada.net/learn/lcd/st7565.html>

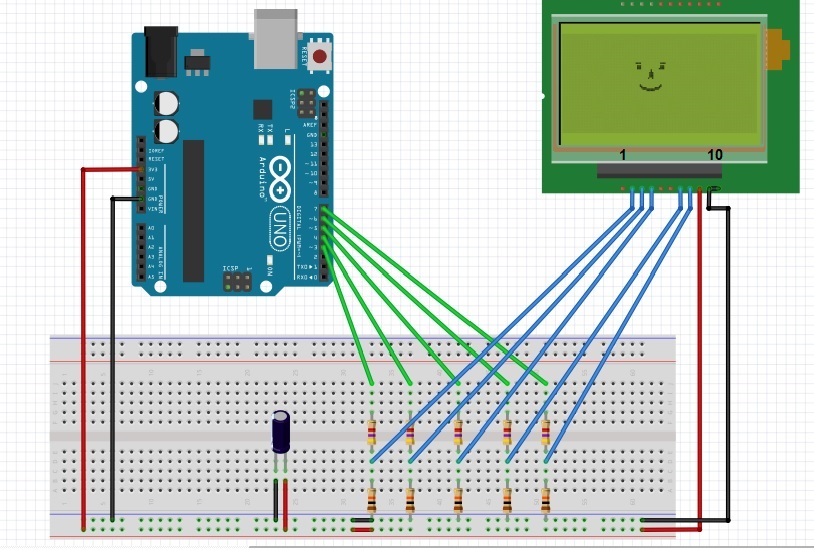
<http://edeca.net/wp/electronics/the-st7565-display-controller/>

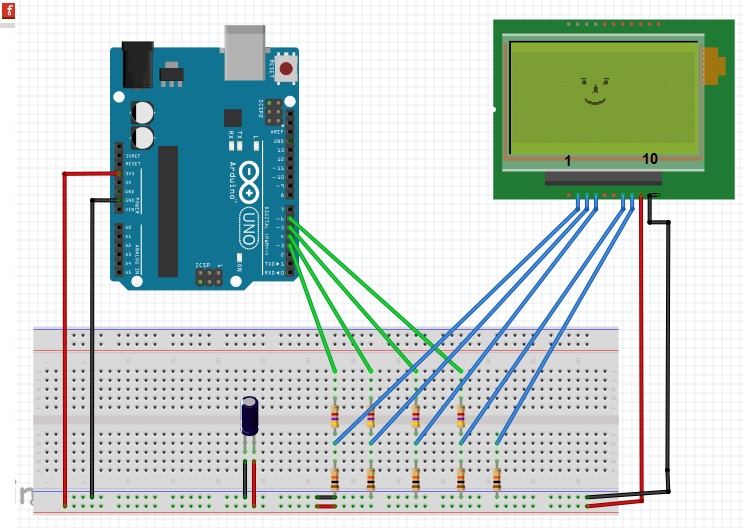
C: KẾT NỐI

Kết nối SPI rút gọn (4 dây) sử dụng điện trở làm cầu phân áp, chú ý nối chân rút gọn CS (Chip Select) xuống nguồn âm GND nhé.

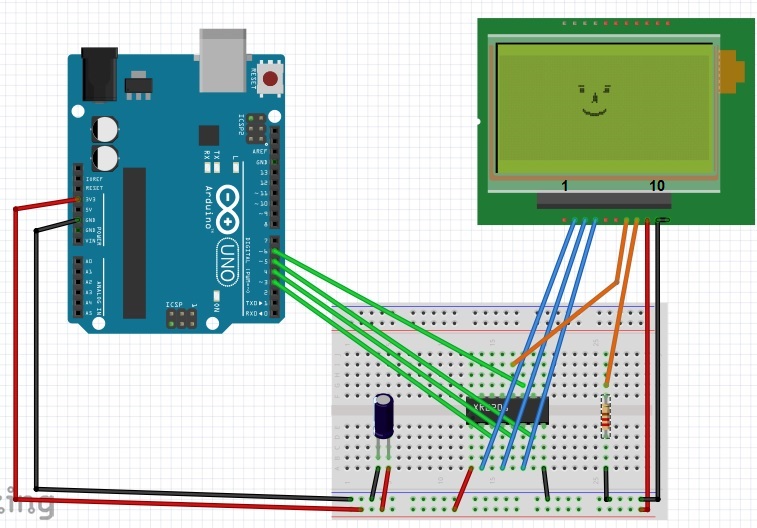


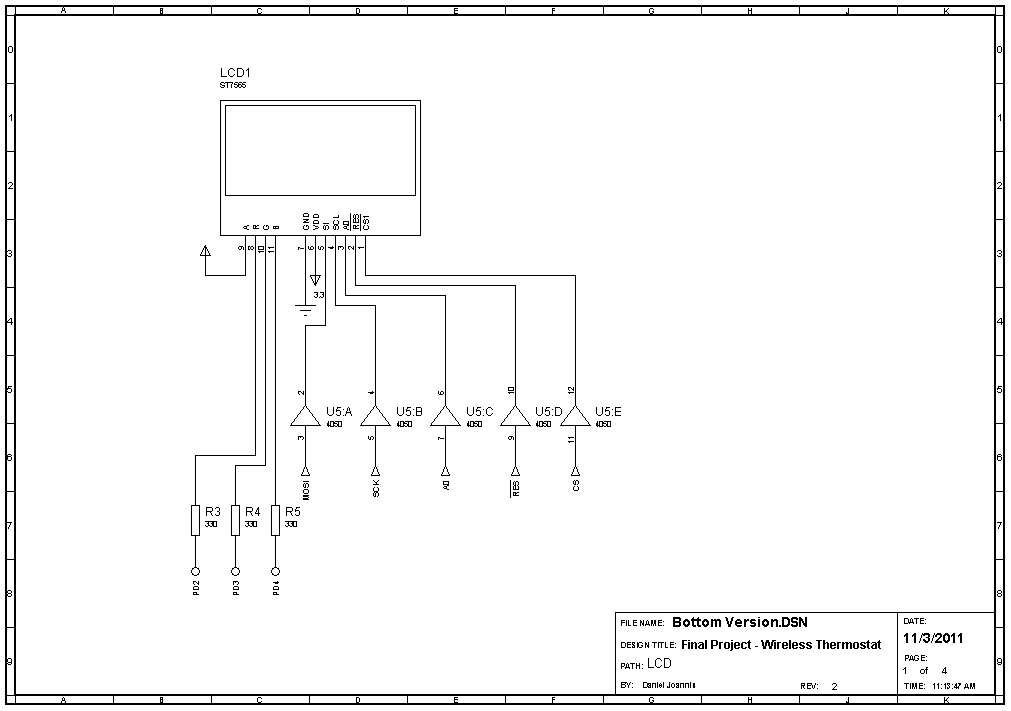
Kết nối SPI đầy đủ:





Nếu dùng ic4050 thì các bạn chú ý nối đúng cực và nguồn cho ic, áp nguồn cho ic 4050 là 5v





C:CÁC QUY ƯỚC CẦN NHỚ

1:Pixel là điểm ảnh nhỏ nhất cấu thành lên “bức ảnh”.

2:Kích thước màn hình là tổng số điểm ảnh mà màn hình có thể hiển thị.

3: Chiều rộng viết trước, chiều cao viết sau (vd:128x64, 320x240..).

4: Trục tọa độ : trục nằm ngang (x), trục đứng (y),

5:Gốc tọa độ là điểm có tọa độ (0,0) nằm ở góc trái trên cùng của màn hình.

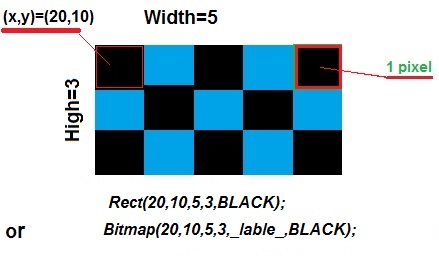
6: Kích thước của ảnh( hình chữ nhật) chính là số pixel của ảnh đó.

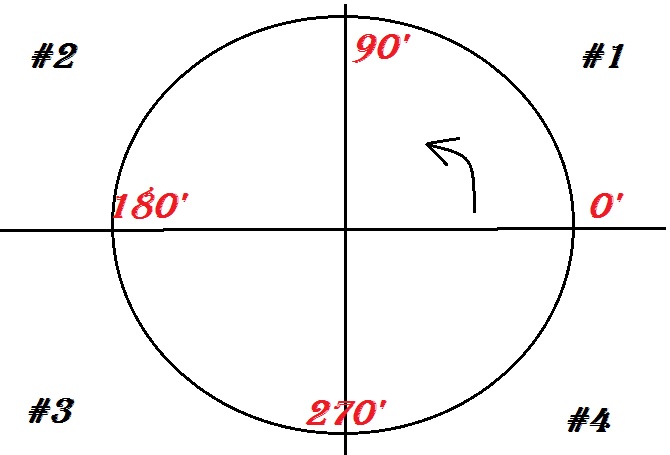
7:Tọa độ của hình vuông, hình chữ nhật là tọa độ của đỉnh nằm ở góc trái trên cùng của hình đó

8:Tọa độ hình tròn chính là tọa độ của tâm

9:Góc quay theo chiều dương thuộc hệ tọa độ Decac có chiều ngược chiều kim đồng hồ







D: SỬ DỤNG CÁC HÀM

Sau khi kết nối hoàn thiện và cài đặt thành công hai thư viện , cùng mở IDE lên và test nhé!

: Cấu trúc khai báo thư viện

<

#include "ST7565\_homephone.h"//a

ST7565 lcd(3,4,5,6);//b

void setup() {

lcd.ON();//c

lcd.SET(22,0,0,0,4); //d

}

void loop(){

} >

Vì đây là thủ tục, bạn phải thực hiện đủ 4 thao tác:

a : Thêm thư viện

b : Thiết lập pin kết nối

c: Bật màn hình

d : Cài đặt cấu hình giao diện

Trong bước b:

|  |  |  |
| --- | --- | --- |
| Số pin | Thứ tự pin | Pin tương ứng muốn dùng |
| Dùng 4 pin | lcd( RST, SCLK, A0, SID) | lcd(3,4,5,6); |
| Dùng 5 pin | lcd( RST, SCLK, A0, SID,CS) | lcd(3,4,5,6,7); |

1:**void ON();**

Gọi hàm này một lần duy nhất để bật màn hình.

2: **void SET(byte contrast, bool negative, bool rotation, bool mirror, byte tyledientro);**

Cài đặt giao diện. Hàm này có thể được gọi nhiều lần với các thông số khác.

\* contrast- độ tương phản- giá trị từ 0->63 (0x00 trở xuống 0x3f)

\* negative- hiển thị âm bản- giá trị 0 hoặc 1

\* rotation- đảo ngược màn hình- giá trị 0 hoặc 1

\* mirror- hiển thi gương- giá trị 0 hoặc 1

\* tyledientro- bằng R1/R2- thiết lập điện áp hoạt động- 0->6( 0x0 đến 0x6)

<//Viết chữ A lên màn hình

#include "ST7565\_homephone.h"

ST7565 lcd(3,4,5,6);

void setup() {

lcd.ON();

lcd.SET(22,0,0,0,4);

}

void loop(){

lcd.Asc\_Char(60,30,'A',BLACK);

lcd.Display();

}

>

3:

Boolean color: chỉ chấp nhận 3 loại sau BLACK (1), WHITE(0),DELETE(0),

void Display();: là hàm cho phép màn hình hiển thị những gì đã vẽ

void Clear(); : xóa tất cả màn hình

// kể từ đây mình sẽ không viết phần thiết lập cấu hình nữa. bạn có thể giữ cấu hình mặc định như ví dụ trên, chúng ta sẽ thay đổi hàm loop

<

void loop(){

lcd.Rect( 60,30,20,20,BLACK);// vẽ hcn A màu đen

lcd.Display();

delay(1000);

lcd.Rect( 60,30,20,20,DELETE);// xóa hcn A

lcd.Display();

delay(1000);

lcd.Rect( 40,30,20,20,BLACK);// vẽ hcn B màu đen

lcd.Display();

delay(1000);

lcd.Rect( 40,30,20,20,WHITE);//vẽ hcn B màu trắng

lcd.Display();

delay(1000);

lcd.Clear();// xóa toàn màn hình

}

>

4: void DrawPixel(int x,int y, bool color);

// vẽ 1 điểm có tọa độ x,y

<

void loop(){

lcd.DrawPixel(60,30,BLACK);//vẽ điểm có tọa độ (60,30)

lcd.Display();

}

>

5: bool GetPixel( int x, int y);

// hàm kiểm tra điểm ở tọa độ x,y có được vẽ( tô ) hay không

//trả về 1 ( true) nếu đúng, ngược lại trả về false

6: void DrawLine(int x0,int y0,int x1,int y1, bool color);

// vẽ đoạn thẳng

// x0,y0: tọa độ điểm thứ nhất

// x1,y1: tọa độ điểm thứ hai

<

void loop(){

lcd.DrawLine(0,0,60,30,BLACK);

lcd.Display();

}

>

7:void Rect(int x,int y, int w, int h, bool color);

// vẽ hình tứ chữ nhật

//x,y: tọa độ đỉnh góc trái trên cùng

//w,h: chiều dài theo trục ngang và trục dọc(kích thước)

<

void loop(){

lcd.Rect(40,30,30,20,BLACK);

lcd.Display();

}

>

8: void FillRect(int x,int y, int w, int h, bool color);

//tô màu hình chữ nhật

<

void loop(){

lcd.FillRect(40,30,40,20,BLACK);

lcd.Display();

}

>

9: void Corner( int x,int y, int w, int h,int r, bool color);

// hình chữ nhật kèm hiệu ứng bo góc

// x,y: tọa độ cúa đỉnh bên trái ,trên cùng

//w,h: chiều ngang, chiều cao;

//r : bán kính đường tròn bo góc

<

void loop(){

lcd.Corner( 30,20,60,30,8,BLACK);

lcd.display();

}

>

10: void FillCorner( int x,int y,int w,int h, int r, bool color);

// vẽ hcn Corner tô màu

<

void loop(){

lcd.FillCorner( 40,20, 40,30,5,BLACK);

lcd.Display();

}

>

11: void Tri (int x1,int y1,int x2,int y2,int x3,int y3, bool color);

//vẽ tam giác có 3 đỉnh A(x1,y1) B(x2,y2), C(x3,y3)

<

void loop(){

lcd.Tri(60,10,30,40,90,40,BLACK);

lcd.Display();

}

>

12: void FillTri(int x1,int y1,int x2,int y2 ,int x3,int y3, bool color);

// vẽ tam giác tô màu

<

void loop(){

lcd.FillTri( 30,20,90,40,60,10,BLACK);

lcd.Display();

}

>

13: void Circle(int x0,int y0,int r, bool color);

// x0,y0: tọa độ tâm

// r: bán kính

<

void loop(){

lcd.Circle( 60,30,20,BLACK);

lcd.Display();

}

>

14: void FillCircle(int x0,int y0, int r, bool color);

// vẽ hình tròn tô màu

<

void loop(){

lcd.FillCircle( 60,30,20,BLACK);

lcd.Display(); }

>

15: void Elip( int x, int y, int xRadius, int yRadius, bool color);

//vẽ elip

// x,y: tâm ellipse

// xRadius,yRadius: bán kính trục đứng và trục ngang

<

void loop(){

lcd.Elip( 60,30,20,10,BLACK);

lcd.Display();

}

>

16: void FillElip(int x0, int y0 , int xRadius, int yRadius, bool color);

//vẽ hình elip tô màu

// x0,y0: tâm ellipse

// xRadius, yRadius: bán kính trục x và y

<

void loop(){

lcd.FillElip( 63,31, 50,20,BLACK);

lcd.Display();

}

>

17: void Asc\_Char(int x1, int y1,unsigned char c, bool color);

//viết 1 kí tự trong bảng ASCII, biến truyền vào có kiếu (unsigned char)

// x1,y1: tọa độ con trỏ

///unsigned char có miền giá trị (0->255);

<

void loop(){

//c1:

char text ='A';

text='A';

lcd.Asc\_Char(10,10,text,BLACK);

lcd.Display();

//c2:

lcd.Asc\_Char(20,10,'B',BLACK);

lcd.Display();

//c3:

lcd.Asc\_Char(30,10,67,BLACK);

lcd.Display();

}

>



18: void Asc\_String(int x1, int y1,unsigned char c[] , bool color);

//x1,y1: tọa độ con trỏ của chữ cái đầu tiên

// c: mảng kí tự dạng chuỗi kí tự

<

void loop(){

//c1:

lcd.Asc\_String(10,15,Asc(" Cach 1"), BLACK);

//c2:

const static unsigned char text[] PROGMEM =" Cach 2";

lcd.Asc\_String(10,35,text, BLACK);

lcd.display();

}

>

19: void Uni\_Char(int x1, int y1, char16\_t c, bool color);

//cài bộ gõ UNIKEY ở chế độ "unicode c string" , nhấn "Đóng"rồi gõ như bình thường nhé

// viết 1 kí tự chữ Việt,

// x,y: tọa độ con trỏ căn lề

// char16\_t : kiểu char 16bit, nhập vào chữ chuẩn mã hóa Unicode

// chú ý tiền tố : u

<

void loop(){

//c1:

lcd.Uni\_Char(63,10,u'A',BLACK);//A

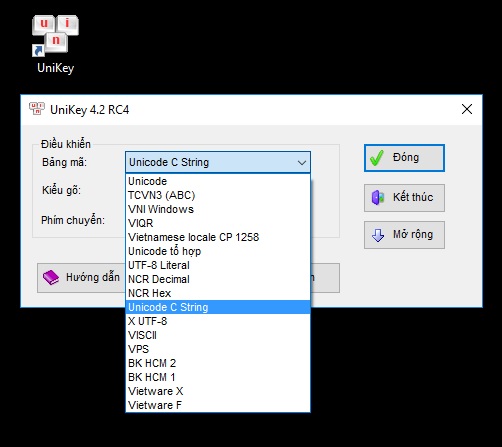
//c2:

const static char16\_t text=u'\x1ED8';//Ộ

lcd.Uni\_Char(63,31,text,BLACK);

lcd.display();

}

>

20: void Uni\_String(int x, int y, char16\_t c[] , bool color);

// viết 1 chuỗi chữ Việt,

// x,y: tọa độ con trỏ căn lề

// char16\_t : kiểu char 16bit, nhập vào chữ chuẩn mã hóa Unicode

//PROGMEM: lưu hằng vào Flash

// chú ý tiền tố : u" "

<

void loop(){

//c1:

lcd.Uni\_String( 30, 15,Uni( u"\x111\x1ECF"), BLACK);//đỏ

//c2:

const static char16\_t text[] PROGMEM =u"\x111\x65n";//đen

lcd.Uni\_String( 30, 35,text, BLACK);

lcd.Display();

}

>

**//Tất cả các cách khai báo chuỗi trên đều được lưu vào Flash, nên bạn có thể tang độ dài chuỗi mà không cần lo thiếu Ram, yên tâm nhé.**

21:

//void Number\_Long(int x, int y,long a,byte select\_font, bool color);

//void Number\_Ulong(int x, int y,unsigned long a,byte select\_font, bool color);

//void Number\_Float(int x, int y, float a,byte n, byte select\_font,bool color);

//viết chữ số

//x,y :tọa độ con trỏ

//a: tham số truyền vào với kiểu biến phù hợp

// select\_font: chọn font

//ASCII\_NUMBER (hoặc bất kì số nào)

//CASIO\_NUMBER (hoặc số 1)

//STYLE\_NUMBER (hoặc số 2)

// n: Nmax=10 (số chữ số muốn hiển thị sau dấu phảy( chỉ dùng cho kiểu float))

<

int k=0;// int , long...

byte l=0;// byte, unsigned int, unsigned long...

float m=0.123;

void loop(){

lcd.Number\_Long(10,10,k,STYLE\_NUMBER,BLACK);

lcd.Number\_Ulong(10,28,l,CASIO\_NUMBER,BLACK);

lcd.Number\_Float(10, 50,m,3,ASCII\_NUMBER,BLACK);

lcd.Display();

k++;

l++;

m++;

lcd.Clear();

}

>



22: void Bitmap(int x, int y, unsigned int w, unsigned int h,const uint8\_t \*bitmap , bool color);

// vẽ ảnh bitmap

//x,y: tọa độ, w,h: kích thước của bitmap đó,

//\*bitmap: tên địa chỉ của file bitmap

<

#ifdef \_\_AVR\_\_

#include <avr/io.h>

#include <avr/pgmspace.h>

#endif

/\*\* bitmap logo 16x 16

\*/

const static unsigned char \_\_attribute\_\_ ((progmem)) logo16\_glcd\_bmp[]= {

0x30, 0xf0, 0xf0, 0xf0, 0xf0, 0x30, 0xf8, 0xbe, 0x9f, 0xff, 0xf8, 0xc0, 0xc0, 0xc0, 0x80, 0x00,

0x20, 0x3c, 0x3f, 0x3f, 0x1f, 0x19, 0x1f, 0x7b, 0xfb, 0xfe, 0xfe, 0x07, 0x07, 0x07, 0x03, 0x00, };

void loop(){

lcd.drawbitmap( 60,30,16,16,logo16\_glcd\_bmp,BLACK);

lcd.display();

}

>

23: void Plus\_Bitmap(int x0, int y0, unsigned int w, unsigned int h,const uint8\_t \*bitmap , int goc, bool mirror, bool color);

// xoay ảnh bitmap:

//x0,y0: tọa độ, w,h: kích thước của bitmap đó,

//\*bitmap: tên địa chỉ của file bitmap

// goc: chỉ chấp nhận 4 xoay góc: 0-90-180-270;

// mirror: MIRROR / NO\_MIRROR hoặc 1/0 hoặc true/false...

24: int Keep\_Angle(int goc);

// giữ cho góc đang tăng \_luôn thuộc khoảng 0 -> 360

<

void setup() {

Serial.begin(9600); // mở monitor để xem

}

void loop(){

int keep;

for( int goc=-720; goc<3600; goc++){

Serial.print( goc);

Serial.print("\_");

keep=lcd.Keep\_Angle(goc);// giữ góc

Serial.println( keep);

delay(10);

}

}>

25:

// void Find\_XY\_Elip(int x0, int y0, int a, int b,int goc\_alpha) ;

//TÌM TẬP CÁC ĐIỂM CÓ QUỸ ĐẠO THUỘC ELLIPSE

//int X\_Elip();

//int Y\_Elip();

// TRẢ VỀ TỌA ĐỘ (X,Y) CỦA ĐIỂM THUỘC ĐƯỜNG ELLIPSE

// goc\_alpha: góc ( độ) tạo bởi điểm đó với trục hoành ( xem hình minh họa nhé);

// góc nhỏ alpha min=-32768, alpha max=32767

<

void loop(){

byte x,y;

byte x0=60, y0=30, a=30,b=20;

for( int goc=0; goc<360; goc++){

lcd.Find\_XY\_Elip(x0,y0,a,b,goc);

x=lcd.X\_Elip();

y=lcd.Y\_Elip();

lcd.DrawPixel( x,y,BLACK);

lcd.Display();

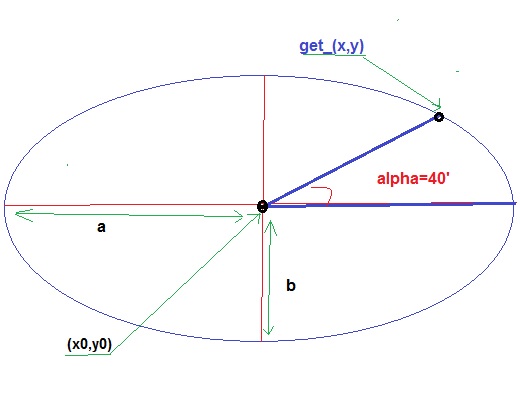
delay(100);

}

lcd.Clear();

}

>



26:

// void Find\_XY\_Sphere(int x0, int y0, int a, int b, int goc\_alpha, int goc\_beta);

//int X\_Sphere();

///int Y\_Sphere();

// tìm tọa độ của điểm A(x,y) có tập quãy đạo là một mặt cầu

// mặt phẳng nằm ngang qua tâm cắt nửa hình cầu là 1 ellipse có tâm x0,y0, bán kính a,b

//trên mặt phẳng nằm ngang, hình chiếu vuông góc của điểm A là A' hớp với trục hoành góc alpha

//trên mặt phẳng đứng qua tâm chứa A, góc hợp bởi điểm A và A' là góc beta

// alpha : 0->360 (min -32768 ->32767)

// beta: 0->360 (min -32768 -> 32767);

<

void loop(){

byte x0=63,y0=31,a=40,b=10;

byte x,y;

for(int beta=0; beta<360; beta+=10){

for( int alpha=0; alpha<360; alpha++){

lcd.Find\_XY\_Sphere(x0,y0,a,b,alpha,beta);

x=lcd.X\_Sphere();

y=lcd.Y\_Sphere();

lcd.DrawPixel( x,y,BLACK);

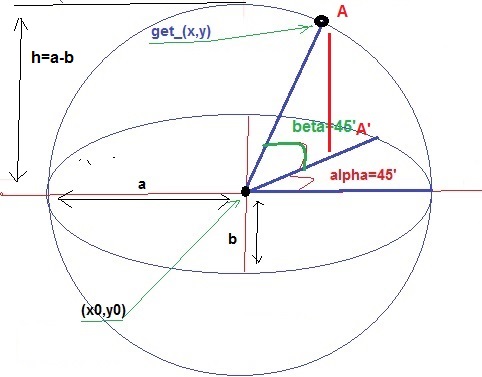
lcd.Display();

}

}

}

>



27: void AllPixel( byte color);

// ép hiển thị toàn bộ điểm ảnh

< void loop(){

lcd.AllPixel(BLACK);

lcd.display();

}

>

26:

// thư viện có hỗ trợ macro mở rộng, bạn có thể khai báo hàm theo 2 kiểu sau đây:

/\*

allpixel AllPixel

drawline DrawLine

drawpixel DrawPixel

getpixel GetPixel

fillcircle FillCircle

circle Circle

corner Corner

fillcorner FillCorner

rect Rect

fillrect FillRect

tri Tri

filltri FillTri

elip Elip

fillelip FillElip

bitmap Bitmap

plus\_bitmap Plus\_Bitmap

asc\_char Asc\_Char

asc\_string Asc\_String

uni\_char Uni\_Char

uni\_string Uni\_String

runstatus RunStatus

number\_long Number\_Long

number\_ulong Number\_Ulong

number\_float Number\_Float

keep\_angle Keep\_Angle

find\_xy\_elip Find\_XY\_Elip

x\_elip X\_Elip

y\_elip Y\_Elip

find\_xy\_sphere Find\_XY\_Sphere

x\_sphere X\_Sphere

y\_sphere Y\_Sphere

analog Analog

pullup\_4 Pullup\_4

clear Clear

display Display

asc Asc

uni Uni

\*/

<

void loop() {

lcd.Circle(30,30,20,BLACK);

lcd.circle(90,30,20,BLACK);

lcd.display();

}>

27: byte Pullup\_4(byte right\_pin, byte up\_pin, byte left\_pin, byte down\_pin);

//hàm có chức năng trả về chỉ số của nút tương ứng khi chúng được nhấn

//=0 nếu không có nút nào được nhấn

//=1 right

//=2 up

//=3 left

//=4 down

//1\*2=20 right và up

//1\*3=30 right và left

//1\*4=40 right và down

//2\*3=60 up và left

//2\*4=80 up và down

//3\*4=120 left và down

/\*

\* tính theo chiều dương của 4 góc phần tư

\_\_\_\_\_\_\_[2]

\_\_\_[3]\_\_\_\_\_[1]

\_\_\_\_\_\_\_[4]

\*/

< // code đầy đủ

#include "ST7565\_homephone.h"

ST7565 lcd(3,4,5,6);

//cài đặt chân input là 4 chân analog

//chú ý: 4 nút nối ở chế độ PULL Up ( tìm hiểu cách nối trước khi tiếp tục);

#define right\_b A3

#define up\_b A2

#define left\_b A1

#define down\_b A0

void setup() {

Serial.begin(9600);

lcd.SET(23,0,0,0,4);

pinMode(left\_b,INPUT\_PULLUP);

pinMode(down\_b,INPUT\_PULLUP);

pinMode(right\_b,INPUT\_PULLUP);

pinMode(up\_b,INPUT\_PULLUP);

}

void loop(){

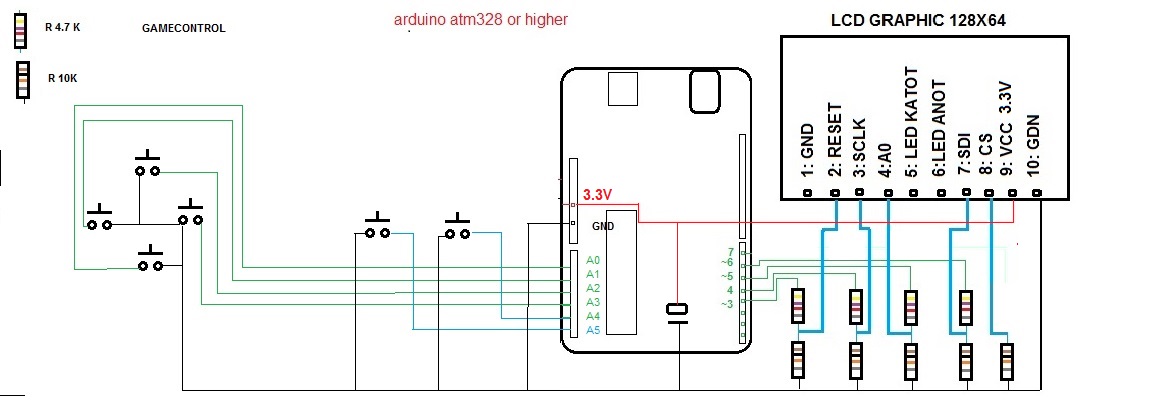
byte value;

value=lcd.Pullup\_4(right\_b, up\_b, left\_b, down\_b);

Serial.println(value);// mở cửa sổ monitor để xem

}

>



15/12/2016 hỗ trợ thay đổi cỡ chữ.Tuyệt cú mèo.

* void Asc\_Char(int16\_t x1, int16\_t y1,unsigned char c PROGMEM,byte he\_so\_phong\_to, bool color);
* void Asc\_String(int16\_t x1, int16\_t y1,PGM\_CHAR s , byte he\_so\_phong\_to, bool color);
* void Uni\_Char(int16\_t x1, int16\_t y1, char16\_t c, byte he\_so\_phong\_to, bool color);
* void Uni\_String(int16\_t x1, int16\_t y1,PGM\_CHAR16T s , byte he\_so\_phong\_to, bool color);
* void Number\_Long(int16\_t x, int16\_t y,long a,byte select\_font, byte he\_so\_phong\_to, bool color);
* void Number\_Ulong(int16\_t x, int16\_t y,unsigned long a,byte select\_font, byte he\_so\_phong\_to, bool color);
* void Number\_Float(int16\_t x, int16\_t y, float a,byte n, byte select\_font, byte he\_so\_phong\_to, bool color);

<

#include "ST7565\_homephone.h"

ST7565 lcd(3,4,5,6);

void setup() {

lcd.ON();

lcd.SET(23,0,0,0,4);

}

void loop(){

lcd.Asc\_Char(10,10,'A',2,BLACK);

lcd.Asc\_Char(22,10,'B',3,BLACK);

lcd.Asc\_Char(40,10,'C',4, BLACK);

lcd.Asc\_Char(67,10,'D',5, BLACK);

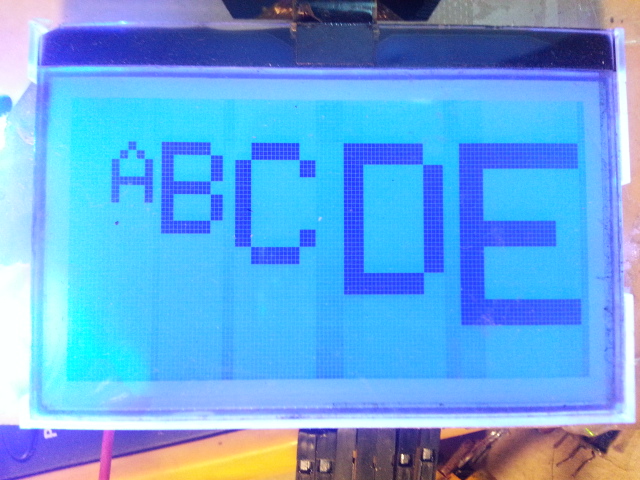
lcd.Asc\_Char(96,10,'E',6, BLACK);

lcd.Display();

}

>

Một vài hình ảnh

:

Với hệ số phóng to lớn hơn hoặc bằng 1.

E: Tải thư viện

Hãy tải và thêm vào kho thư viện 2 tệp sau:

1: avr.zip (thư viện hỗ trợ)

2: st7565\_homephone.zip (thư viện chính)

Video hướng dẫn.

Video trailer.

(Thái Sơn.)