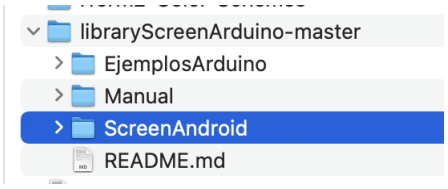
Tutorial Para conectar Arduino a Pantalla Android

step 1: DownLoad library and install

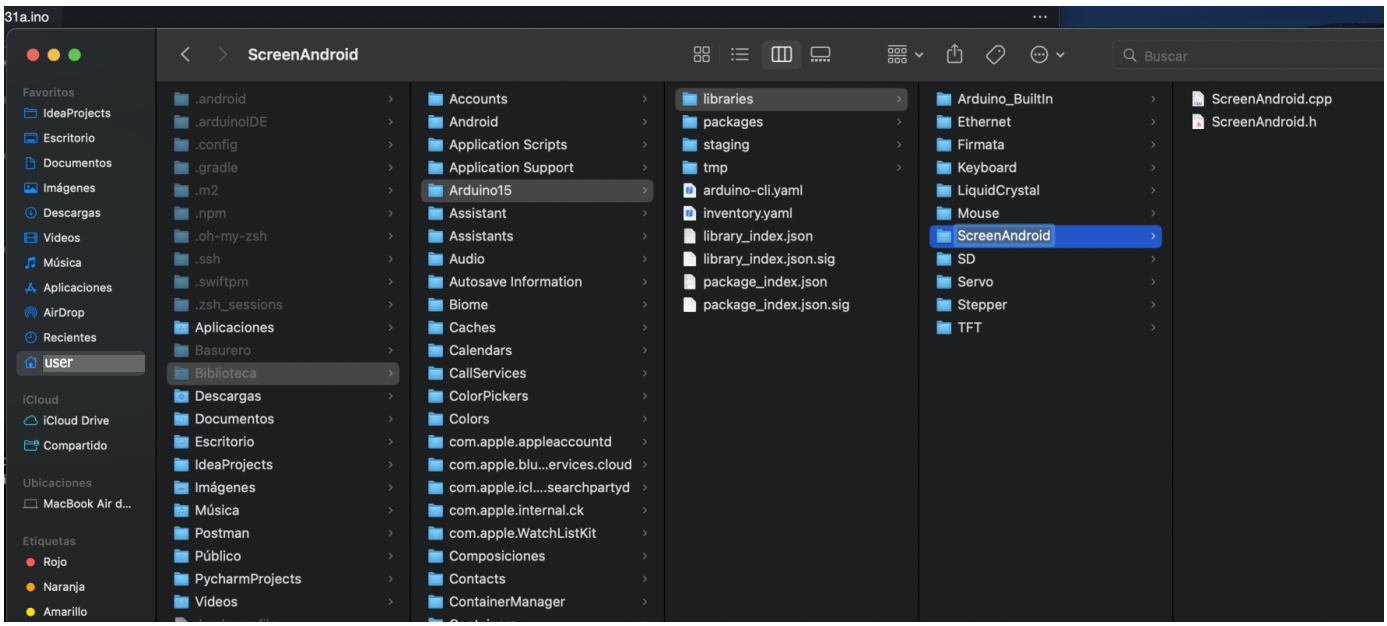
Descarga o clona el repositorio en: https://github.com/johnspice/libraryScreenArduino
Se descargarán 3 carpetas:



MACOS

NOTA: version antigua de ArduinoIDE Version 1.8 o menores

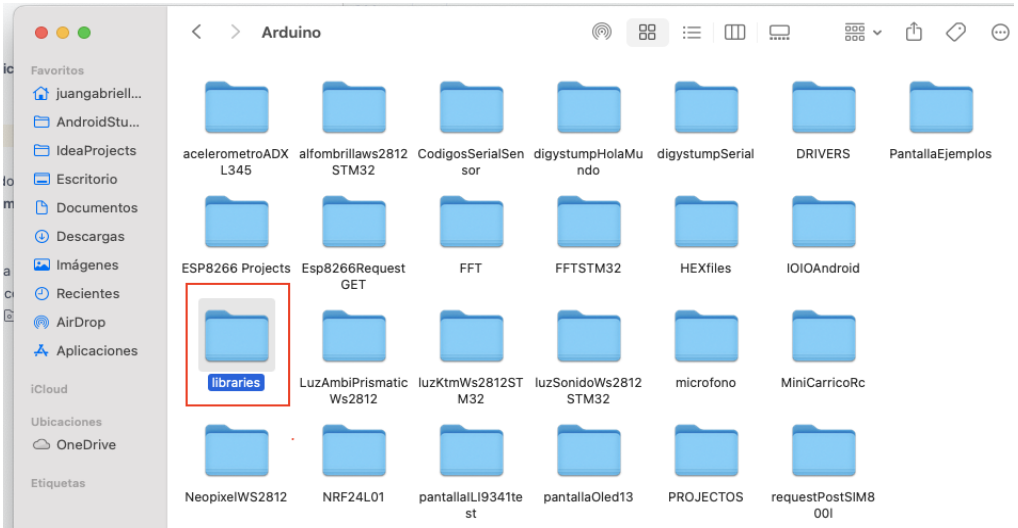
Copy “ScreenAndroid” to “`/Users/yourUser/Library/Arduino15/libraries`”

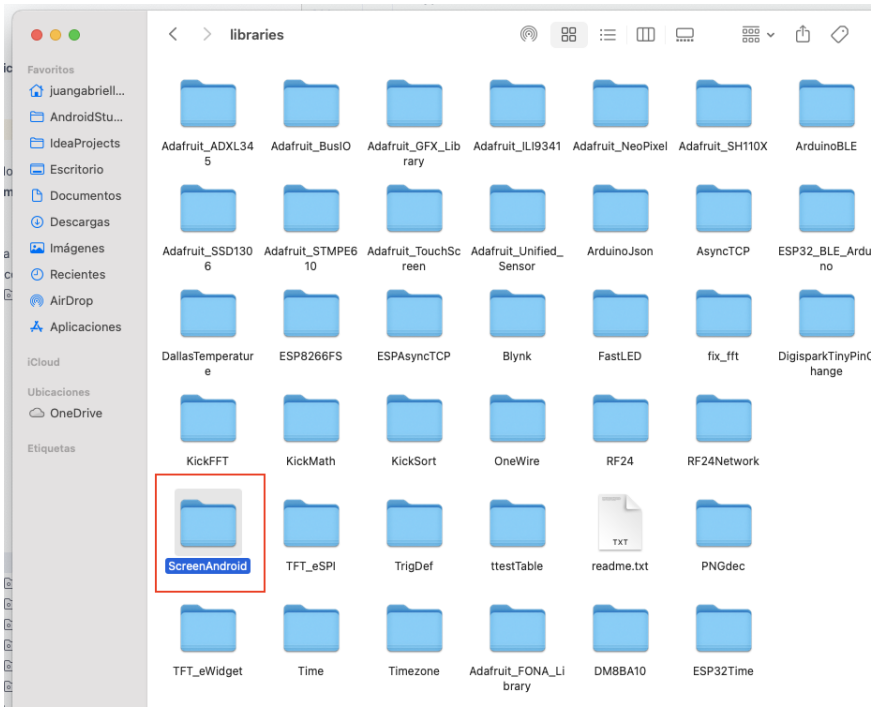


`Note: “Library” may not be visible use command+shift+dot inside finder to show hidden files.You may need to restart the arduino for it to recognize the library.`

Para versiones superiores a 1.8

copy the "ScreenAndroid" folder to Documents/Arduino/libraries





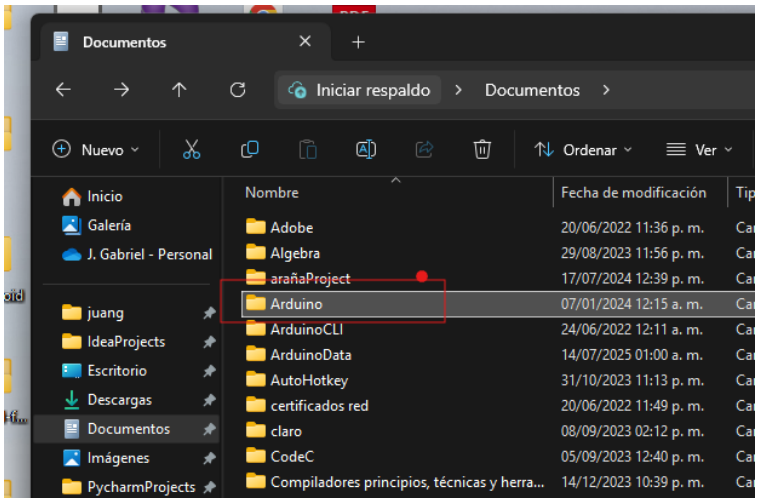
Windows

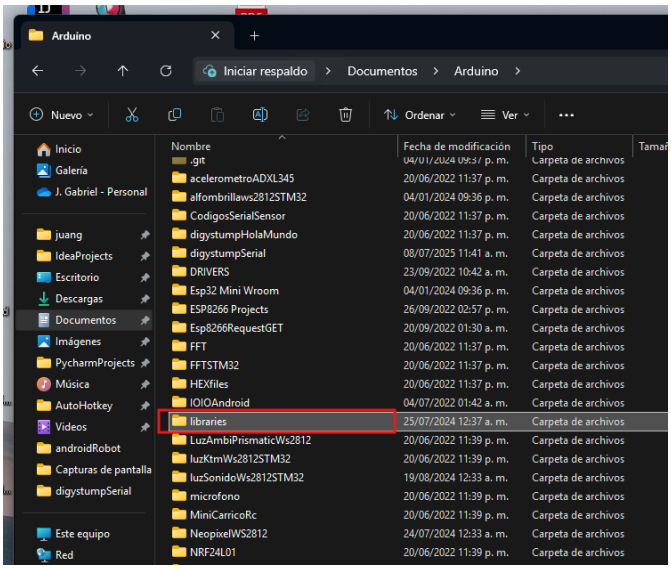
NOTA: version antigua de ArduinoIDE Version 1.8 o menores

Copy into C:\Users\yourUser\Documents\Arduino\libraries

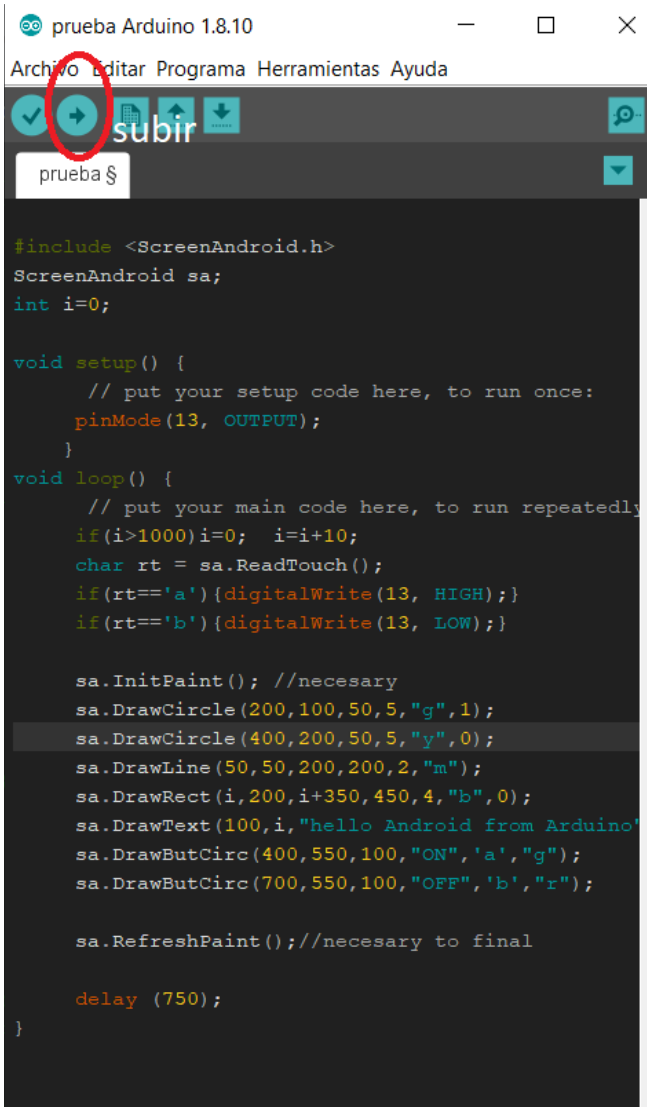
Para versiones superiores a 1.8

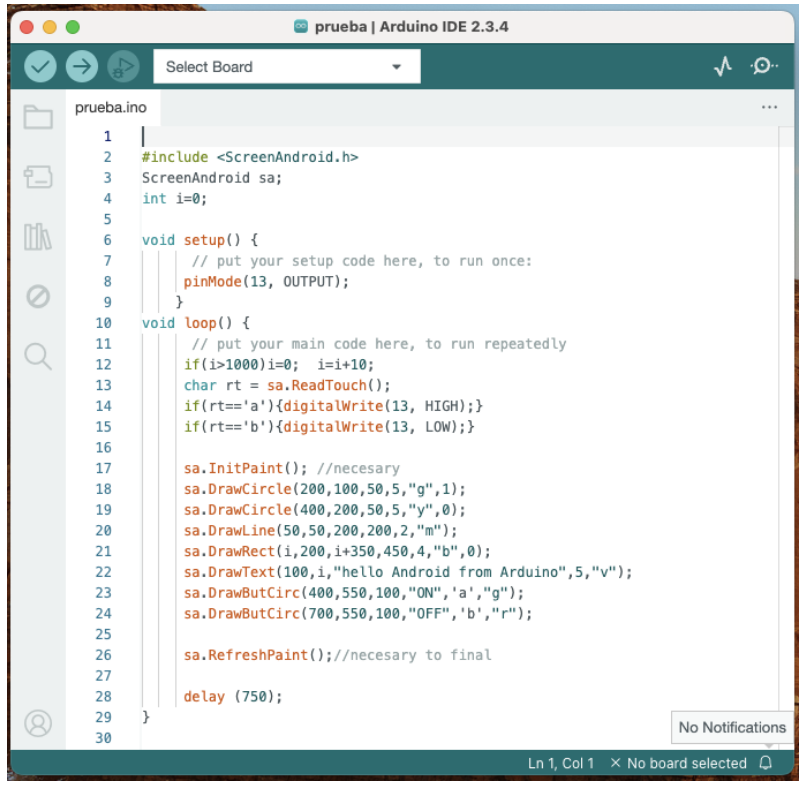
copy the "ScreenAndroid" folder to Documents/Arduino/libraries





Paso 2 Run Test Code

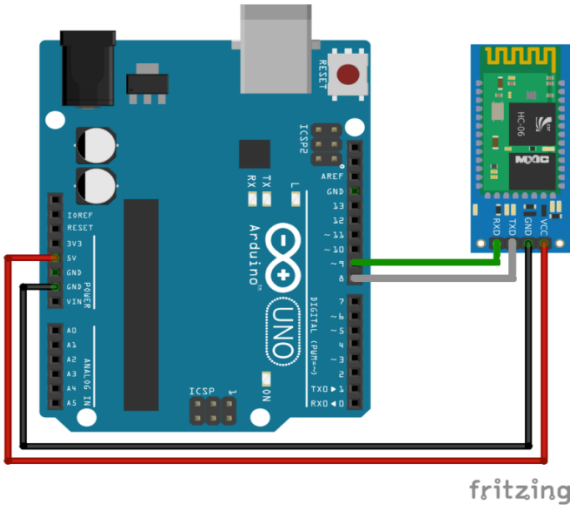




Open arduino and load the "prueba.ino" test code that is inside the "EjemplosArduino" folder. Choose serial port card and go up to the board. (in the next section the code is explained)

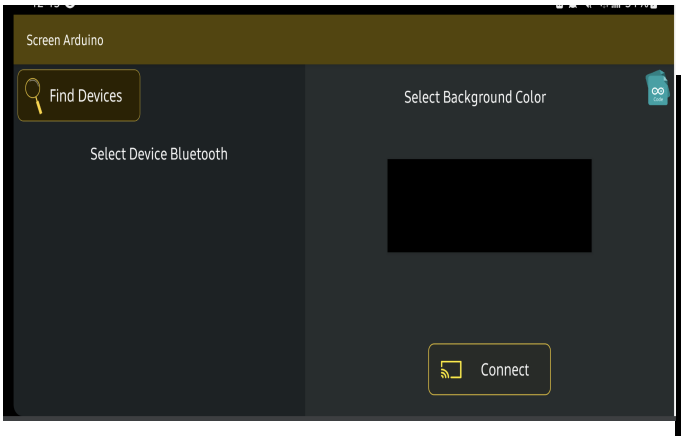
step 3 Conexión Arduino-bluetooth (hc-05 o hc06)

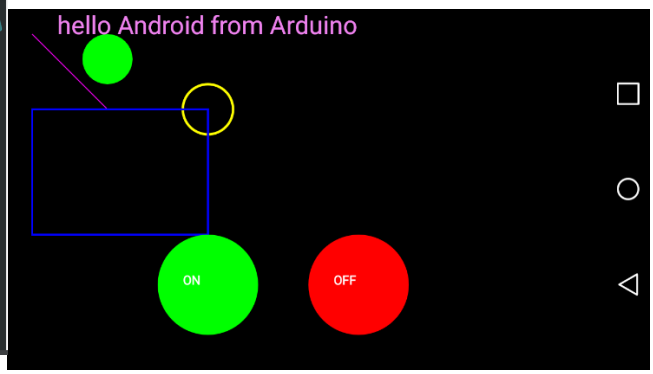
By default the connection is on pins rx=8 and tx= 9, at 9600 bauds. As it's shown in the following. If you need to paint many elements on the screen and have a refresh time of less than 750ms, you should change this value to the maximum to 115200, keep in mind that changing it will make it necessary to change it also in hc05/06 using At commands.



step 4 Conexión hc05/06-Android

-Go to settings in android and pair the bluetooth hc06/05
-open App in android, click find, select bluetooth, seleccionar color de fondo de la pantalla , connect.





Descripción del API ScreenAdroid

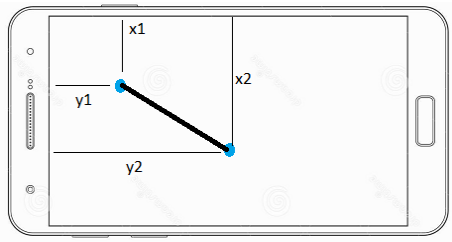
Functions included in the ScreenArduino library.

Before starting to paint it is necessary to call the method
InitPaint(); and at the end you must call the method
RefreshPaint();

If these two methods are no t placed, it will cause errors in android.
The library has only some basic colors that will be called with initials

Draw Line

DrawLine(int x1,int y1, int x2,int y2, int w,string color);



(x1,y1) init Point pixeles
(x2,y2) end Pointf pixeles

w=thick line pixeles
color =line color.
Options Colors :green=g, yellow=y, orange=o,red=r,
violet=v, blue=b, cyan=c, magenta=m, white=w, black=bc, gray=gr
example: DrawLine(50,50,200,200,2,"m"); line color is magenta

Draw Circle

DrawCircle(int x,int y,int r,int w,string color,int fill);

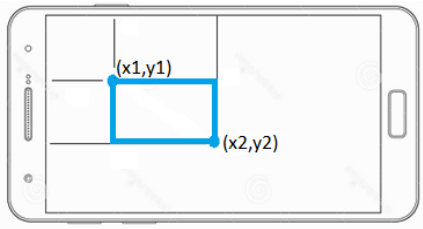
Center (x,y) pixeles
r=radius pixeles

w=line thickness
color=same colors as the line
fill= can be **1** or **0** 1=filled circle, 0=outline only.

Ejemplo: DrawCircle(200,100,50,5,"g",1); filled green circle

Draw rectangle

DrawRect(int x1,int y1,int x2,int y2,int w,int color,int fill);



color = same as line
If fill=1 the rectangle is filled fill=0 only the outline is drawn
Example: DrawRect(100,200,350,450,4,"b",0);

Draw Text

DrawText(int x,int y,string mensaje,int size,string color);

```
Text position (x,y),message=text to print, size text size must be a value
between 1 and 20.
Color=the same as line.
Example; DrawText(100,200,"hello Android from Arduino",5,"v");
```

Draw button

DrawButCirc(int x,int y,int r,string text,char return,string color);

position (x,y) radius=r, text=text button

```
return = is the character that will be sent from android serially when touching the
button on the screen
color = the same as line, the button is always filled.
```

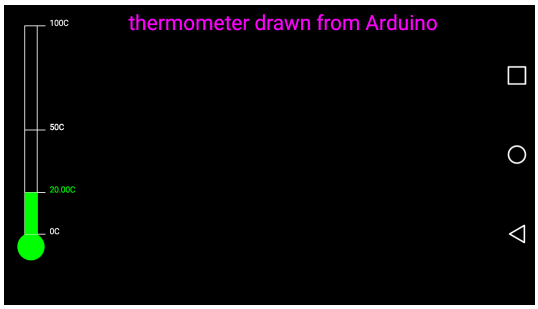
example

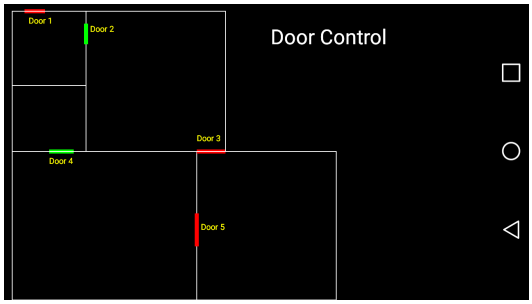
DrawButCirc(700,550,100,"OFF",'b',"r");

```
A circular button will be drawn at the position x=700,y=550 r=100 all in pixels. with
text off Touching the button will send the character 'b' to the arduino via serial,
the button is red.
```

see the test code to see how it is implemented

2 more examples of how to paint a thermometer (example1.ino) and a very simple house plan (example2.ino) are included.






```
If the delay in arduino is greater than 1 second, the reaction speed of the
circular button will suffer a delay when clicked. If you want to draw many
elements with a 100ms refresh, you must change the speed to 115200 bauds in
the "ScreenAndroid.cpp" file that was copied to libraries within arduino.

The library was tested with arduino uno, mega and Nodemcu Wifi Module Esp8266,
although the latter is necessary to change the pins and the test led to the
arduino library.
```