

Tecnicatura Superior en Telecomunicaciones



Electronica Microcontrolada

Shields v1.0

Ejercicio #1

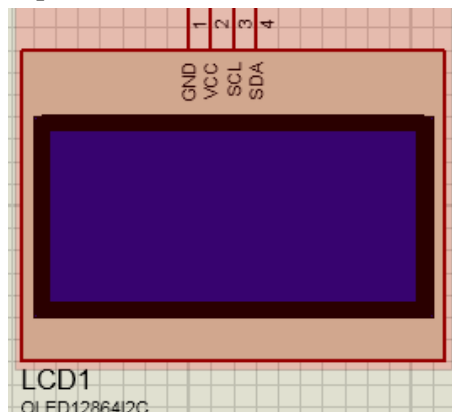
d) Que es el controlador ssd1306(i2c). Existe alguna shield para controlar una pantalla oled 128x64?; si es así, implemente una practica donde muestre el mensaje “Es fácil el desarrollo con shields”.

Respuesta:

Lo que hace el SSD1306 es comunicar con el microcontrolador para obtener los datos y enviarlos a la pantalla OLED para que dibuje esos datos.

La comunicación entre el SSD1306 y el microcontrolador, ya sea un Arduino o un ESP8266, se realiza mediante SPI o I2C. . Generalmente, la comunicación SPI es más rápida que la comunicación I2C. Por el contrario, la comunicación SPI requiere de más pines que la comunicación I2C.

La pantalla OLED una pantalla que utiliza la interfaz I2C. En este tipo de pantallas encontraremos 4 pines:



GND: pin de tierra.

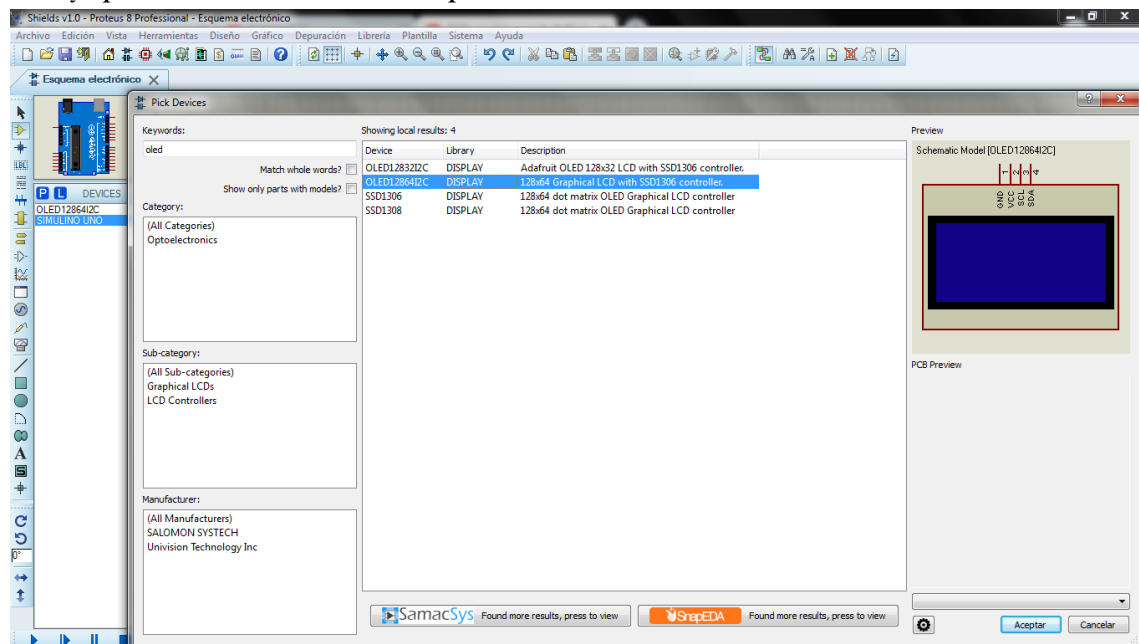
VCC: es el pin de alimentación. Se puede alimentar la pantalla entre 1,8V y 6V.

SCL: es el pin de la señal de reloj de la interfaz I2C.

SDA: es el pin de la señal de datos de la interfaz I2C.

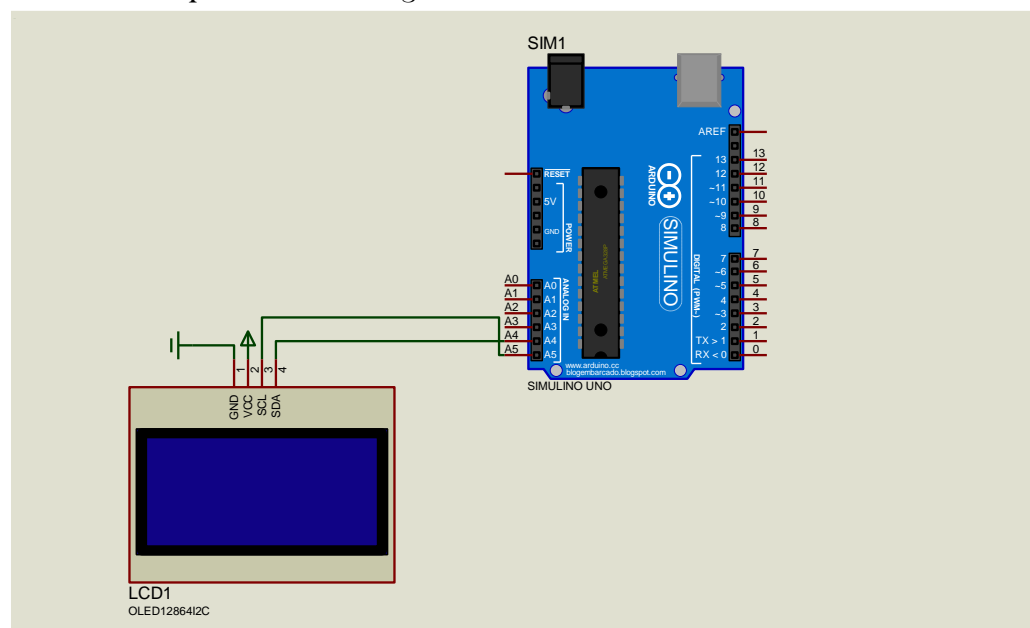
Ahora voy a explicar y desarrollar el uso de la pantalla OLED12864I2C y su posterior simulacion en Proteus:

Primero deberiamos ir a la ventadad de “Modo de Componente” y buscar la palabra OLED seleccionando la que tiene 128 pixeles de ancho por 64 pixeles de alto y que se comunica sobre el protocolo I2C:

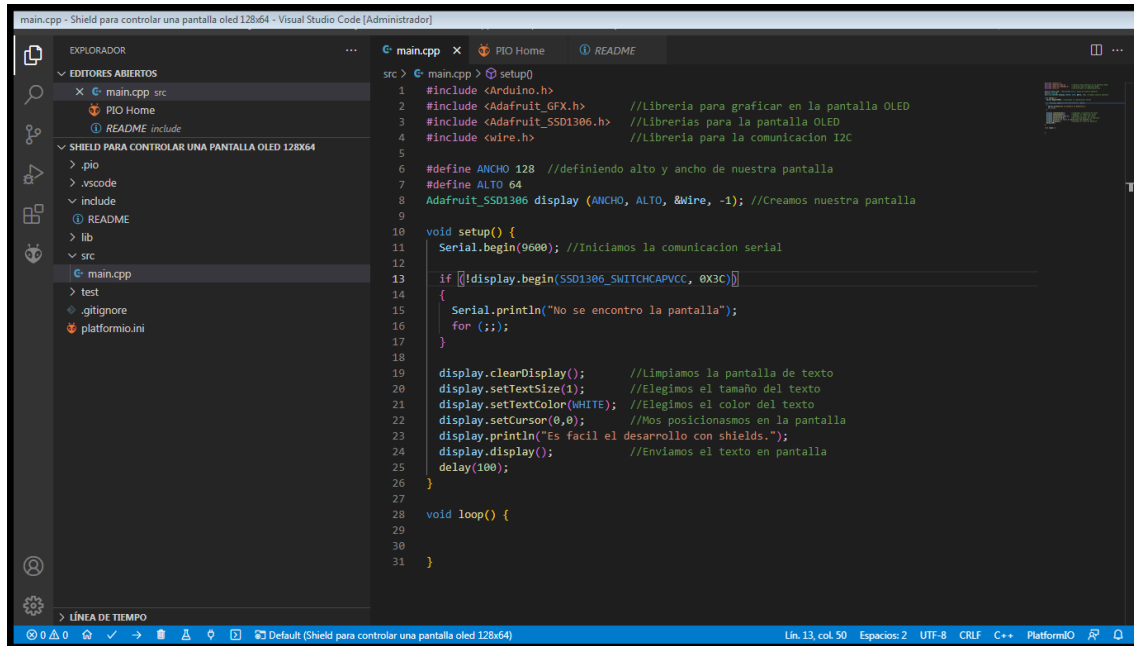


Conectamos lo pines respectivios a nuestro arduino donde encontremos el SDA en A5, y SCL en A4

Lo cual nos quedaria de la siguiente manera:



Luego armaríamos nuestro código en Visual Studio Code, donde mostraría un texto en pantalla, concretamente “Es fácil el desarrollo con shields”



```
1 #include <Arduino.h>
2 #include <Adafruit_GFX.h> //Libreria para graficar en la pantalla OLED
3 #include <Adafruit_SSD1306.h> //Librerias para la pantalla OLED
4 #include <Wire.h> //Libreria para la comunicacion I2C
5
6 #define ANCHO 128 //definiendo alto y ancho de nuestra pantalla
7 #define ALTO 64
8 Adafruit_SSD1306 display (ANCHO, ALTO, &Wire, -1); //Creamos nuestra pantalla
9
10 void setup() {
11   Serial.begin(9600); //Iniciamos la comunicacion serial
12
13   if (!display.begin(SSD1306_SWITCHCAPVCC, 0x3C))
14   {
15     Serial.println("No se encontro la pantalla");
16     for (;;);
17   }
18
19   display.clearDisplay(); //Limpiamos la pantalla de texto
20   display.setTextSize(1); //Elegimos el tamaño del texto
21   display.setTextColor(WHITE); //Elegimos el color del texto
22   display.setCursor(0,0); //Nos posicionamos en la pantalla
23   display.println("Es facil el desarrollo con shields.");
24   display.display(); //Enviamos el texto en pantalla
25   delay(100);
26 }
27
28 void loop() {
29
30 }
31 }
```

Cada recordar que es necesario el uso de 2 librerías que nos permitirá el uso/manejo de la pantalla OLED:

- Adafruit_SSD1306.h: Que es la propia para poder controlar este tipo de pantallas que están utilizando el microcontrolador SSD1306.
- Adafruit_GFX.h: Esta es opcional pero nos facilitara el manejo a la hora de imprimir algunas figuras geométricas.
- Wire.h: Es la librería para la comunicación I2C.

Una vez instaladas la librerías, podríamos trabajar en nuestra pantalla OLED. Definiríamos el alto y ancho de nuestra pantalla; y también crearemos nuestro objeto dándole un nombre a nuestra pantalla que en mi caso es display con 4 parámetros ANCHO (que es el alto de la pantalla), ALTO (que es el alto de la pantalla), &Wire (Indicamos el tipo de comunicación que tendremos en nuestra pantalla) y -1 (para indicar el pin de reset, siempre y cuando nuestra pantalla en físico tenga para resetear).

```
#define ANCHO 128 //definiendo alto y ancho de nuestra pantalla
#define ALTO 64
Adafruit_SSD1306 display (ANCHO, ALTO, &Wire, -1); //Creamos nuestra pantalla
```

Ahora pasaremos a nuestro Void Setup, aquí iniciaremos nuestra comunicación serial con la velocidad de 9600

```
void setup() {
  Serial.begin(9600); //Iniciamos la comunicacion serial
```

Daremos lugar un IF para verificar si nuestra pantalla está conectada o no, como ya vimos que se maneja sobre el protocolo I2C, tenemos que darle una dirección que es 0X3C; y en el caso de que no la detecte nos va a imprimir que no se encontró la pantalla:

```
if (!display.begin(SSD1306_SWITCHCAPVCC, 0X3C))
{
  Serial.println("No se encontro la pantalla");
  for (;;);
}
```

Ahora si vamos a poder a manejar nuestra pantalla con una serie de comandos propias de Adafruit; primero limpiamos la pantalla, luego elegimos el tamaño del texto que es(1) y corresponde a una altura de 8 pixeles, también elegiremos el color del texto a blanco y por ultimo ubicaremos nuestro pixel donde la primera parte corresponde a los renglones y la segunda a las columnas, imprimiremos un mensaje que dira “Es fácil el desarrollo con shields”. Por ultimo enviaremos el mensaje mediante display. “el nombre de nuestra pantalla”) que en este caso es display, y también tendrá un delay de 100 segundos para evitar pequeños problemas.

```
display.clearDisplay();           //Limpiamos la pantalla de texto
display.setTextSize(1);           //Elegimos el tamaño del texto
display.setTextColor(WHITE);      //Elegimos el color del texto
display.setCursor(0,0);           //Mos posicionamos en la pantalla
display.println("Es facil el desarrollo con shields.");
display.display();                //Enviamos el texto en pantalla
delay(100);
```

¡Lo hemos logrado! Ahora solo debemos exportarlo y ver los resultados en nuestra simulación:

