

**Pantalla LCD
84x48
ebook para**

Pantalla LCD 84x48

Nokia 5110



Áreas de aplicación

Educación y enseñanza: uso en escuelas, universidades e instituciones de formación para enseñar los conceptos básicos de electrónica, programación y sistemas integrados. Investigación y desarrollo: Uso en proyectos de investigación y desarrollo para crear prototipos y experimentos en los campos de la electrónica y la informática. Desarrollo de prototipos: Uso en el desarrollo y prueba de nuevos circuitos y dispositivos electrónicos. Proyectos Hobby and Maker: utilizado por entusiastas y aficionados a la electrónica para desarrollar e implementar proyectos de bricolaje.

Conocimientos y habilidades requeridos.

Conocimientos básicos de electrónica e ingeniería eléctrica. Conocimientos de programación, especialmente en el lenguaje de programación C/C++. Capacidad para leer esquemas y diseñar circuitos simples. Experiencia trabajando con componentes electrónicos y soldadura.

Condiciones de operación

El producto sólo puede funcionar con los voltajes especificados en la hoja de datos para evitar daños. Se requiere una fuente de alimentación CC estabilizada para su funcionamiento. Al realizar la conexión a otros componentes y circuitos electrónicos, se deben observar los límites máximos de corriente y voltaje para evitar sobrecargas y daños.

Condiciones ambientales

El producto debe utilizarse en un ambiente limpio y seco para evitar daños causados por la humedad o el polvo. Proteja el producto de la luz solar directa (UV), ya que esto puede afectar negativamente la vida útil de la pantalla.

Uso previsto

El producto está diseñado para su uso en entornos educativos, de investigación y desarrollo. Se utiliza para desarrollar, programar y crear prototipos de proyectos y aplicaciones electrónicos. El producto no pretende ser un producto de consumo terminado, sino más bien una herramienta para usuarios con conocimientos técnicos, incluidos ingenieros, desarrolladores, investigadores y estudiantes.

Uso inadecuado previsible

El producto no es adecuado para uso industrial o aplicaciones relevantes para la seguridad. No se permite el uso del producto en dispositivos médicos o para fines de aviación y viajes espaciales.

desecho

¡No lo deseches con la basura doméstica! Su producto es acorde al europeo. Directiva sobre residuos de aparatos eléctricos y electrónicos que deben eliminarse de forma respetuosa con el medio ambiente. Las valiosas materias primas contenidas en ellos se pueden reciclar. Convertirse en. La aplicación de esta directiva contribuye a la protección del medio ambiente y la salud. Utilice el punto de recogida habilitado por su municipio para devolver y Reciclaje de aparatos eléctricos y electrónicos viejos. N.º registro RAEE: DE 62624346

descarga electrostática

La pantalla es sensible a las descargas electrostáticas (ESD), que pueden dañar o destruir los componentes electrónicos. Tenga en cuenta las siguientes instrucciones de seguridad para evitar riesgos de ESD: Atención: Las cargas electrostáticas en su cuerpo pueden dañar la pantalla. Nota: Conéctese a tierra usando una muñequera antiestática conectada a una superficie con conexión a tierra o tocando una superficie metálica con conexión a tierra antes de manipular la pantalla. Atención: Utilice bolsas y tapetes antiestáticos para proteger la pantalla. Nota: Coloque la pantalla sobre una alfombra de trabajo antiestática y guárdela en bolsas antiestáticas cuando no esté en uso. Nota: Un lugar de trabajo limpio y conectado a tierra minimiza el riesgo de ESD. Acción: Mantenga su lugar de trabajo limpio y libre de materiales que puedan generar cargas electrostáticas. Asegúrese de que todas las superficies utilizadas estén conectadas a tierra.

instrucciones de seguridad

Aunque la pantalla cumple con los requisitos de la Directiva RoHS (2011/65/UE) y no contiene sustancias peligrosas en cantidades superiores a los límites permitidos, aún pueden existir riesgos químicos residuales. Tenga en cuenta las siguientes instrucciones de seguridad: Atención: La parte posterior de la pantalla y la placa de circuito pueden liberar residuos químicos de fabricación o durante el funcionamiento. Nota: Use guantes protectores cuando manipule o instale la pantalla durante mucho tiempo para evitar irritación de la piel. Precaución: Los componentes electrónicos pueden emitir pequeñas cantidades de compuestos orgánicos volátiles (COV), especialmente si la pantalla es nueva. Nota: asegúrese de trabajar en un área

bien ventilada para minimizar la concentración de vapores en el aire. Precaución: No utilice productos químicos ni disolventes agresivos para limpiar la pantalla, ya que pueden dañar la capa protectora o los componentes electrónicos. Nota: Utilice un paño de limpieza antiestático o un limpiador especial para componentes electrónicos para limpiar cuidadosamente la pantalla. Aunque la pantalla cumple con los requisitos de la Directiva RoHS (2011/65/UE) y no contiene sustancias peligrosas en cantidades superiores a los límites permitidos, aún pueden existir riesgos químicos residuales. Tenga en cuenta las siguientes instrucciones de seguridad: Atención: La parte posterior de la pantalla y la placa de circuito pueden liberar residuos químicos de fabricación o durante el funcionamiento. Nota: Use guantes protectores cuando manipule o instale la pantalla durante mucho tiempo para evitar irritación de la piel. Precaución: Los componentes electrónicos pueden emitir pequeñas cantidades de compuestos orgánicos volátiles (COV), especialmente si la pantalla es nueva. Nota: asegúrese de trabajar en un área bien ventilada para minimizar la concentración de vapores en el aire. Precaución: No utilice productos químicos ni disolventes agresivos para limpiar la pantalla, ya que pueden dañar la capa protectora o los componentes electrónicos. Nota: Utilice un paño de limpieza antiestático o un limpiador especial para componentes electrónicos para limpiar cuidadosamente la pantalla. La pantalla contiene componentes electrónicos sensibles y una capa superior. Un manejo inadecuado o una presión excesiva pueden causar daños a la pantalla o lesiones. Observe las siguientes instrucciones de seguridad para evitar riesgos mecánicos: Atención: La tapa del display es frágil y puede romperse si se manipula incorrectamente. Nota: Evite aplicar una presión fuerte o doblar la pantalla. Manipule la pantalla con cuidado y sólo por la placa de circuito para evitar roturas. Precaución: Las caídas o los impactos pueden agrietar la superficie de la pantalla y dañar los componentes electrónicos de la parte posterior. Nota: Evite dejar caer la pantalla y protéjala de impactos. Utilice una superficie suave cuando trabaje para evitar rayones. Atención: Si la pantalla se rompe, los trozos de vidrio afilados pueden provocar lesiones. Nota: Si la pantalla se rompe, manipule los fragmentos con cuidado y use guantes protectores para evitar cortes. Deseche las piezas de vidrio de forma segura. Nota: Una fijación incorrecta puede provocar tensión mecánica y rotura de la pantalla. Acción: Coloque la pantalla de forma segura y sin presión excesiva. Utilice soportes o carcasas adecuados para montar la pantalla de forma estable. Precaución: Los métodos de limpieza inadecuados pueden rayar o dañar la superficie. Nota: Utilice únicamente paños suaves y antiestáticos para limpiar la pantalla. Evite agentes de limpieza agresivos y fricciones fuertes. La pantalla funciona con voltajes y corrientes eléctricas que, si se usan incorrectamente, pueden provocar descargas eléctricas, cortocircuitos o incendios. Tenga en cuenta las siguientes instrucciones de seguridad: Atención: Utilice el producto únicamente con los voltajes especificados. Nota: Los límites de rendimiento del producto se pueden encontrar en la hoja de datos asociada. Nota: Las fuentes de voltaje inadecuadas pueden dañar la pantalla o provocar situaciones peligrosas. Acción: Utilice únicamente fuentes de alimentación o baterías probadas y adecuadas para alimentar sus circuitos. Asegúrese de que la fuente de voltaje cumpla con los requisitos de la pantalla. Precaución: Evite cortocircuitos entre los conectores y componentes del producto. Nota: Asegúrese de que ningún objeto conductor toque o puente la placa de circuito. Utilice herramientas aisladas y preste atención a la disposición de las conexiones. Precaución: No realice ningún trabajo en el producto cuando esté conectado a una fuente de alimentación. Nota: Desconecte el producto de la alimentación antes de realizar cambios en el circuito o conectar o quitar componentes. Nota: busque señales de daños eléctricos, como humo, olores inusuales o decoloración. Acción: Si ocurren tales signos, apague la alimentación inmediatamente e inspeccione minuciosamente el circuito para detectar errores. La pantalla puede generar calor durante el funcionamiento, lo que podría provocar sobrecalentamiento, quemaduras o incendios si se manipula incorrectamente. Tenga en cuenta las siguientes instrucciones de seguridad: Atención: Algunos componentes de la pantalla pueden calentarse durante el funcionamiento o en caso de error. Medida: Después de apagarla, deje que la pantalla se enfríe lo suficiente antes de tocar directamente los componentes individuales de la parte posterior. Evite el contacto directo con componentes calientes. Precaución: La sobrecarga puede provocar un calentamiento excesivo de los componentes electrónicos. Nota: Asegúrese de que la fuente de alimentación y voltaje cumpla con las especificaciones de la pantalla y no cause sobrecarga.

Pantalla LCD 84x48

Índice

Introducción	3
Especificaciones	4
El Pinout	5
Instalación del IDE Arduino	6
Configuración de la Raspberry Pi y Python	10
Conexión de la pantalla con ATmega328p	11
Biblioteca para Arduino IDE	13
Ejemplo de croquis	15
Conexión de la pantalla con la Raspberry Pi	26
Habilitación de la interfaz SPI	28
Bibliotecas y herramientas para Python	29
Script en Python	30

Introducción

La pantalla LCD de 84x48 es un dispositivo pequeño y fácil de usar que consume poca energía y puede mostrar tanto texto como mapas de bits. El chip controlador de la pantalla se basa en un controlador LCD CMOS PCD8544 de bajo consumo fabricado por Philips.

El chip PCD8544 está diseñado para controlar una pantalla gráfica de 84x48 píxeles. Se conecta a los microcontroladores a través de una interfaz de bus serie similar a SPI.

El chip controla la fuente de alimentación de la pantalla LCD, así como la regulación de tensión de los niveles lógicos, lo que se traduce en un bajo consumo de energía adecuado para aplicaciones sensibles a la energía y aplicaciones alimentadas por batería. Cuando la pantalla no recibe ninguna información (estado de reposo), consume tan sólo 7 mA.

La tensión de funcionamiento de la pantalla está en el rango de 2,7 a 3,3V. Para utilizar la pantalla con un microcontrolador de 5V, se debe utilizar un dispositivo Logic Level Converter, de lo contrario la pantalla puede dañarse.

La pantalla tiene una retroiluminación LED integrada. La tensión de funcionamiento de los LEDs de retroiluminación es de 3V, se puede conectar a los 5V pero hay que utilizar una resistencia adecuada para proteger los (4) LEDs de retroiluminación de daños.

Pantalla LCD
84x48

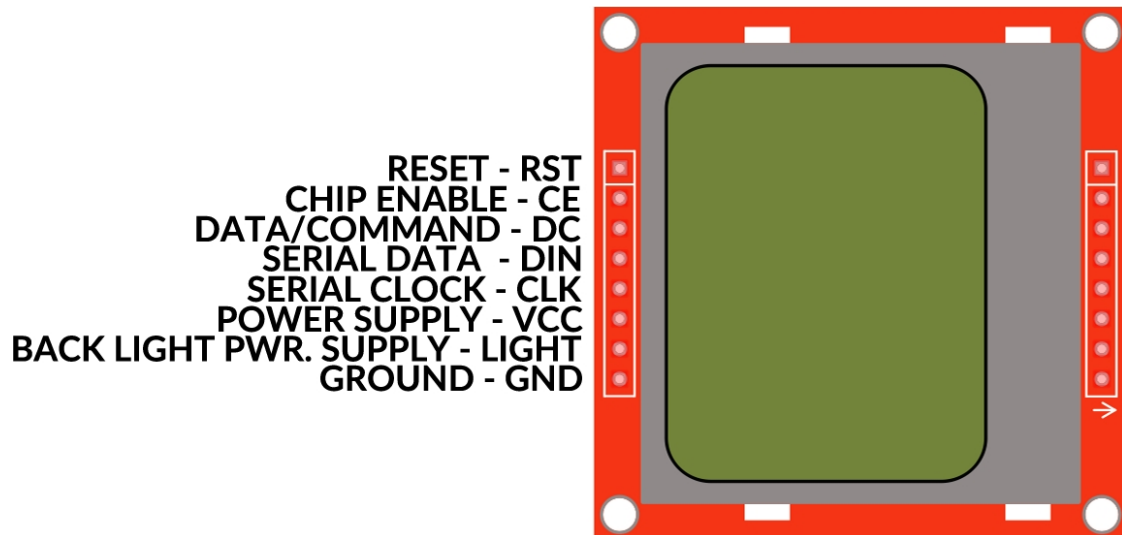
Especificaciones

Tensión de funcionamiento	2,7 a 3,3 V
Corriente de funcionamiento	50 mA máx.
Temperatura de funcionamiento	-27 a +70°C
Tipo de pantalla	LCD de matriz de puntos
Tamaño de la pantalla	1,5 pulgadas
Resolución	84x48 píxeles
Ángulo de visión	180°
Interfaz	SPI
Dimensiones	45x45x7mm

Pantalla LCD 84x48

El Pinout

El LCD 84x48 tiene 16 pines, ocho iguales en ambos lados. El pinout se muestra en el siguiente diagrama de conexión:



RST - pin es un pin activo bajo que puede reiniciar la pantalla tirando de él hacia abajo.

CE - (Chip enable) es un pin activo bajo que se utiliza para seleccionar el dispositivo cuando hay más de uno conectado en el bus SPI.

DC - (Data/Command) del pin de interfaz SPI que indica al visualizador si los datos que está recibiendo son un comando o datos visualizables.

DIN - es un pin de datos serie para la interfaz SPI.

CLK - es un pin de reloj serie para la interfaz SPI.

VCC - pin suministra energía para el LCD que conectamos a los 3.3V

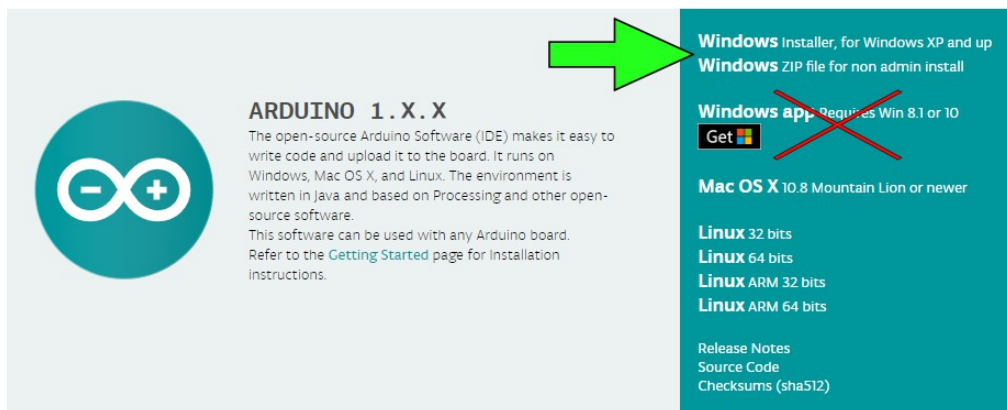
LIGHT - (Retroiluminación) controla la retroiluminación de la pantalla.

Instalación del IDE Arduino

Puede descargar el entorno de desarrollo gratuito Arduino IDE desde el siguiente enlace: <https://www.arduino.cc/en/Main/Software>

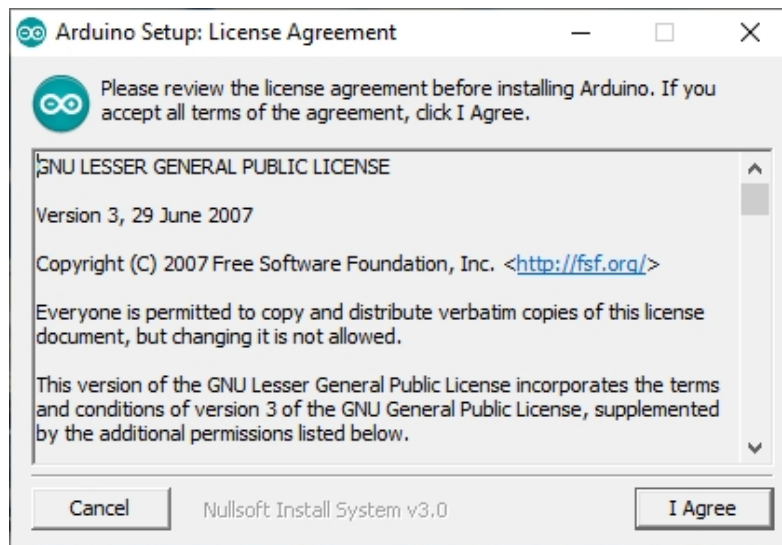
Los usuarios de Windows deben utilizar definitivamente una de las dos primeras opciones de descarga para el IDE de Arduino. La versión "Windows App" de la Windows Store causará problemas de conexión, especialmente cuando se utilizan definiciones de placas de terceros.

Download the Arduino IDE

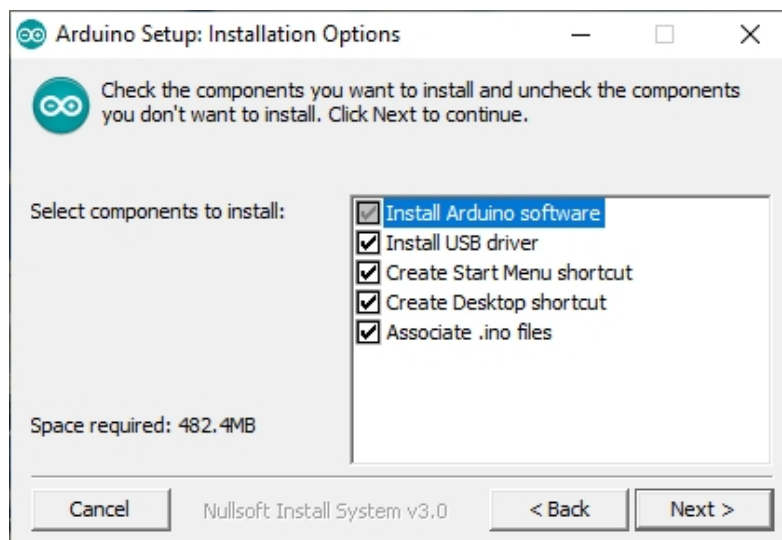


Después de iniciar el archivo de instalación del IDE Arduino "arduino-1.X.X-windows.exe" se deben leer y aceptar las condiciones de licencia del software:

Pantalla LCD 84x48



En el siguiente paso, se pueden seleccionar diferentes opciones para la instalación.



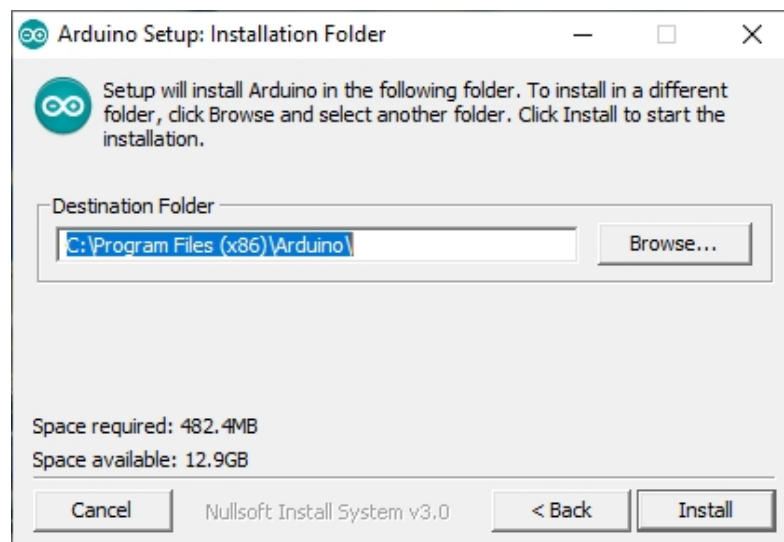
A continuación se ofrece un breve resumen de las distintas opciones, con una breve explicación de cada una de ellas:

Pantalla LCD

84x48

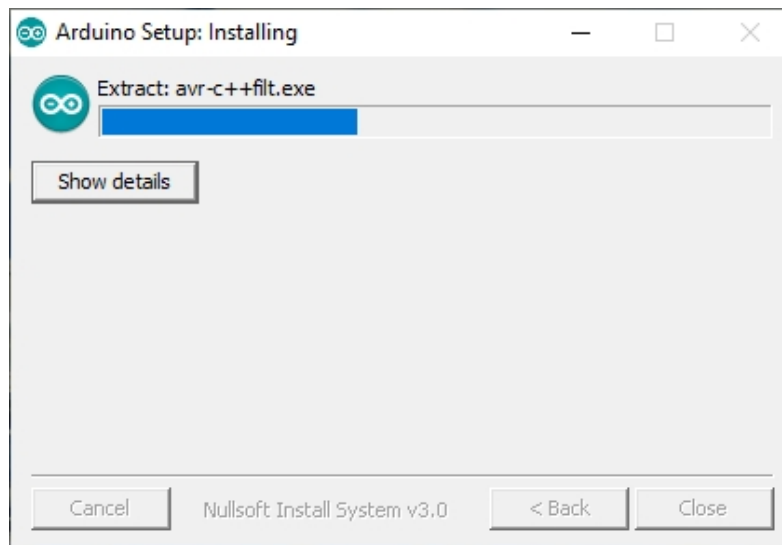
Opción	Erklärung
Instalar el software Arduino	Instala el IDE de Arduino - Esta opción no se puede deseleccionar
Instalar controlador USB	Instala controladores USB para otros microcontroladores. No son necesarios para utilizar el software con el D1 mini, pero le recomendamos encarecidamente que los instale si también utiliza otros microcontroladores.
Crear acceso directo al menú Inicio	Crea un acceso directo en el menú Inicio de Windows (opcional)
Crear acceso directo al escritorio	Crea un acceso directo en la estación de trabajo (Opcional)
Asociar archivos .ino	Crea una extensión de nombre de archivo para los archivos con la extensión .ino y la vincula al IDE de Arduino

Por último, debe especificarse la carpeta de destino. La instalación requiere unos 500 MB de espacio libre en disco.

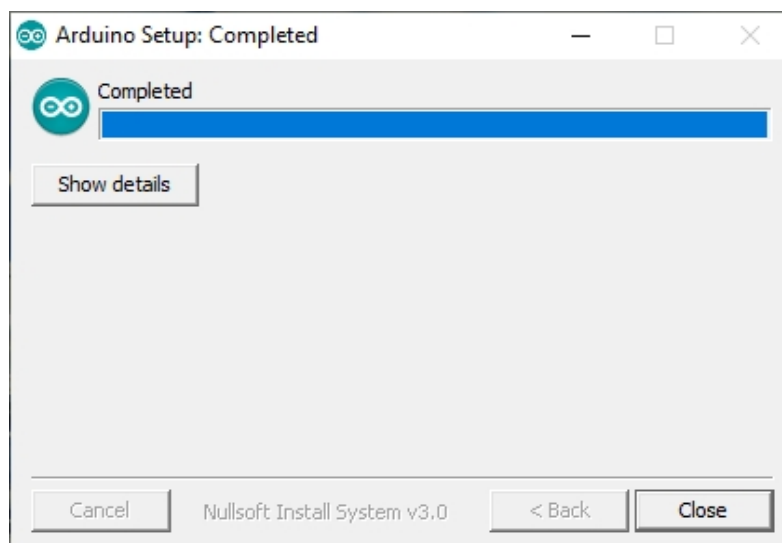


Haga clic en "Instalar" para iniciar la instalación.

Pantalla LCD 84x48



Tras la instalación, el programa de instalación puede cerrarse con el botón "Cerrar":



Configuración de la Raspberry Pi y Python

En el caso de la Raspberry Pi, primero hay que instalar el sistema operativo y, a continuación, configurar todo para que pueda utilizarse en modo headless. El modo Headless permite la conexión remota a la Raspberry Pi sin necesidad de un monitor de pantalla de PC, ratón o teclado. Lo único que se utiliza en este modo es la propia Raspberry Pi, la fuente de alimentación y la conexión a Internet. Todo esto se explica en detalle en el eBook gratuito:

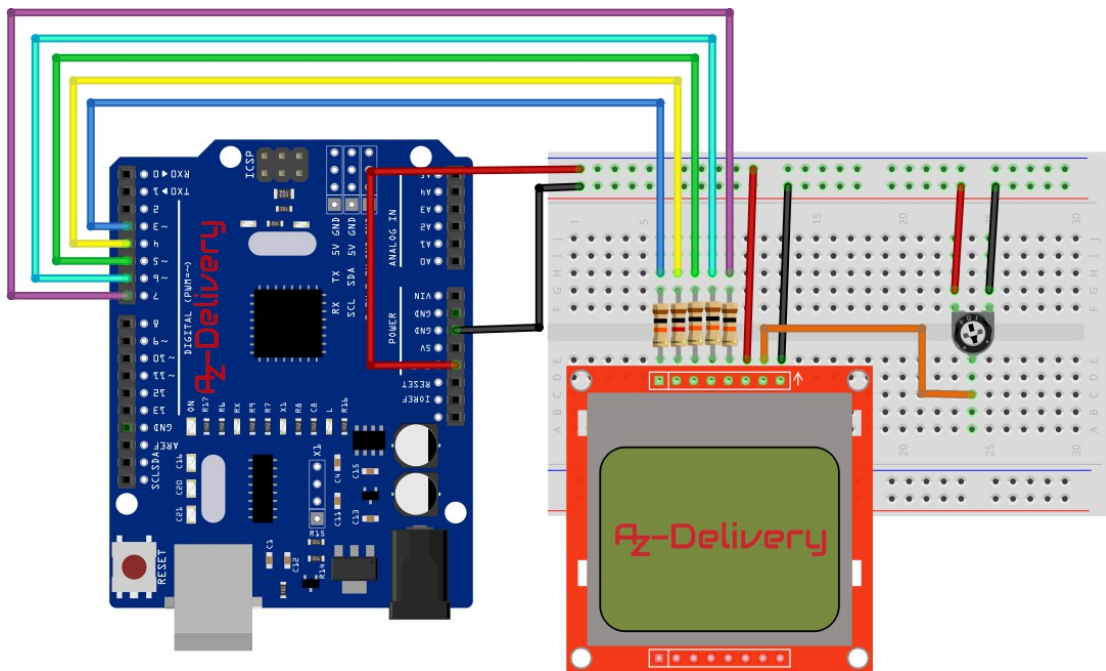
[Guía de inicio rápido de Raspberry Pi](#)

Python está preinstalado en el sistema operativo Raspberry Pi.

Pantalla LCD 84x48

Conexión de la pantalla con ATmega328p

Conecte el módulo con el ATmega328p como se muestra en la siguiente imagen:



Nota: Los pines del display trabajan en el rango de 3V. Para utilizar el módulo con el Atmega328p, debes añadir resistencias de 10k Ohm para RST, DC, DIN, CLK y 1k Ohm para CE. De lo contrario, la introducción de la señal de los pines Atmega328p a los pines de la pantalla causará daños a la pantalla.

Alternativamente, puede utilizar un convertidor de nivel lógico que AZ-Delivery también ofrece.

Advertencia: No conecte 5V a la pantalla

Pantalla LCD 84x48

Pin del microcontrolador	Pin de la pantalla	Color del cable
D3	RST(con 10K ohm)	azul
D4	CE(con 1K ohm)	amarillo
D5	CC(con 10K ohm)	verde
D6	DIN(con 10K ohm)	cian
D7	CLK(con 10K ohm)	morado
3.3V	VCC	rojo
GND	GND	negro

Potenciómetro para el LED de retroiluminación:

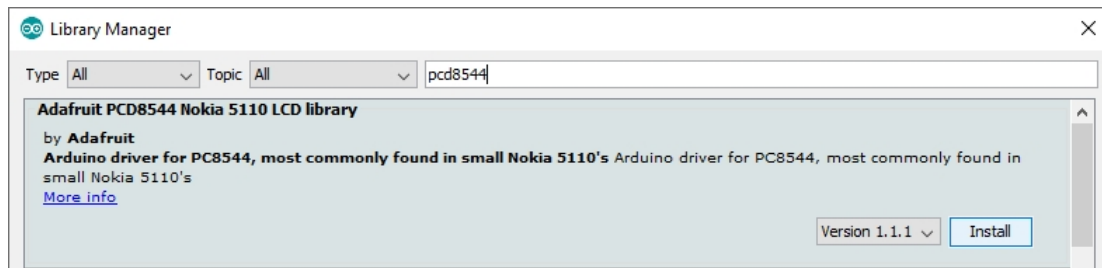
Conecte la salida del potenciómetro (1.5k ohm) al pin "LIGHT" de la pantalla (cable naranja).

Pantalla LCD 84x48

Biblioteca para Arduino IDE

Para utilizar la pantalla con un Atmega328p, se recomienda descargar una librería externa. Las librerías que se van a utilizar se llaman "Adafruit PCD8544 Nokia 5110 LCD library" versión 1.1.1 y "Adafruit GFX library" versión 1.8.2.

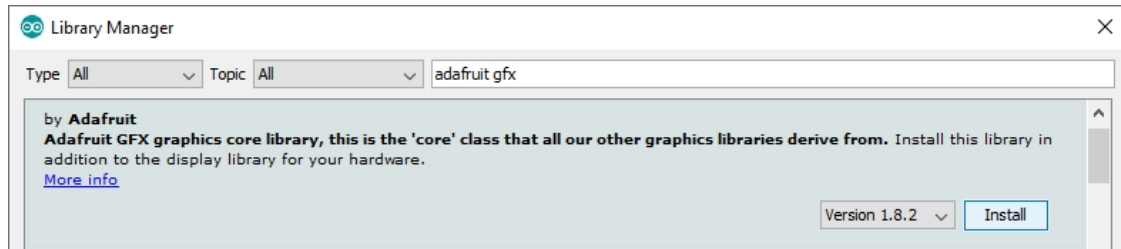
Para descargar e instalar "Adafruit PCD8544 Nokia 5110 LCD library", abra Arduino IDE y vaya a: Herramientas > Gestionar Bibliotecas. Cuando se abra una nueva ventana, escribe "PCD8544" en el cuadro de búsqueda e instálala, como se muestra en la siguiente imagen:



Pantalla LCD

84x48

Para descargar e instalar "Adafruit GFX library", abra Arduino IDE y vaya a: Herramientas > Gestionar Bibliotecas. Cuando se abra una nueva ventana, escribe "Adafruit GFX" en el cuadro de búsqueda e instálala, como se muestra en la siguiente imagen:



Cuando se pulsa el botón Instalar, se abre una nueva ventana, solicitando instalar dependencias, librerías que se utilizan con Adafruit GFX Library. Pulse el botón "Instalar todo" y espere unos instantes a que todo termine. La ventana de solicitud se puede ver en la siguiente imagen:



Pantalla LCD 84x48

Sketch ejemplo

El Sketch utilizado en este ebook es el ejemplo que se proporciona con la librería "Adafruit PCD8544". Para abrirlo puedes ir a Archivo > ejemplos > Adafruit PCD8544 Nokia 5110 > pcdtest:

pcdtest.ino

```
#include <SPI.h>
#include <Adafruit_GFX.h>
#include <Adafruit_PCD8544.h>

// Software SPI (actualizaciones más lentas, opciones de pines más flexibles):
// pin 7 - Salida de reloj serie (SCLK)
// pin 6 - Salida de datos serie (DIN)
// pin 5 - Selección de datos/comando (D/C)
// pin 4 - Selección de chip LCD (CS)
// pin 3 - reset LCD (RST)
Adafruit_PCD8544 display = Adafruit_PCD8544(7, 6, 5, 4, 3);

// Hardware SPI (más rápido, pero debe usar ciertos pines de hardware):
// SCK es el reloj serie del LCD (SCLK) - este es el pin 13 en Arduino Uno
// MOSI es LCD DIN - este es el pin 11 en un Arduino Uno
// pin 5 - Selección de datos/comando (D/C)
// pin 4 - Selección de chip LCD (CS)
// pin 3 - reset LCD (RST)
// Adafruit_PCD8544 display = Adafruit_PCD8544(5, 4, 3);
// Nota con hardware SPI los pines MISO y SS no se usan pero seguirán siendo leídos
// y escrito durante la transferencia SPI. ¡Ten cuidado al compartir estos pines!

#define NUMFLAKES 10
#define XPOS 0
#define YPOS 1
#define DELTAY 2
```

Pantalla LCD 84x48

```
#define LOGO16_GLCD_HEIGHT 16
#define LOGO16_GLCD_WIDTH 16

static const unsigned char PROGMEM logo16_glcd_bmp[] =
{ B00000000, B11000000,
  B00000001, B11000000,
  B00000001, B11000000,
  B00000011, B11100000,
  B11110011, B11100000,
  B11111110, B11111000,
  B01111110, B11111111,
  B00110011, B10011111,
  B00011111, B11111100,
  B00001101, B01110000,
  B00011011, B10100000,
  B00111111, B11100000,
  B00111111, B11110000,
  B01111100, B11110000,
  B01110000, B01110000,
  B00000000, B00110000 };

void testdrawbitmap(const uint8_t *bitmap, uint8_t w, uint8_t h)
{
  uint8_t iconos[NUMFLAKES][3];
  randomSeed(666);    // cualquier semilla

  // inicializar
  for (uint8_t f=0; f< NUMFLAKES; f++) {
    iconos[f][XPOS] = random(display.width());
    iconos[f][YPOS] = 0;
    iconos[f][DELTAY] = random(5) + 1;

    Serial.print("x: ");
    Serial.print(iconos[f][XPOS], DEC);
    Serial.print(" y: ");
    Serial.print(iconos[f][YPOS], DEC);
    Serial.print(" dy: ");
    Serial.println(iconos[f][DELTAY], DEC);
  }

  while (1) {
```

Pantalla LCD 84x48

```
// dibujar cada icono
for (uint8_t f=0; f< NUMFLAKES; f++) {
    display.drawBitmap(icons[f][XPOS], icons[f][YPOS],
logo16_glcd_bmp, w, h, BLACK);
}
display.display();
delay(200);

while(Serial.available()) {
    switch (Serial.read()) {
        case 'w':display.setContrast(display.getContrast() +
1);
            romper;
        case 's':if(display.getContrast())
display.setContrast(display.getContrast() - 1);
            break;
        case 'e':display.setBias(display.getBias() + 1);
            break;
        case 'd':if(display.getBias())
display.setBias(display.getBias() - 1);
            romper;
        case 'R':display.setReinitInterval(10);
            break;
        case 'r':display.initDisplay();
            display.setReinitInterval(0);
            break;
    }
}
Serial.print("contraste (c/s): 0x");
Serial.print(display.getContrast(), HEX);
Serial.print("    sesgo (e/d): 0x");
Serial.print(display.getBias(), HEX);
Serial.print("    reinicializar pantalla (r/R):
0x"); Serial.print(display.getReinitInterval(),
HEX);
Serial.print("    \r");

// luego borrarlo + moverlo
for (uint8_t f=0; f< NUMFLAKES; f++) {
    display.drawBitmap(icons[f][XPOS], icons[f][YPOS],
logo16_glcd_bmp, w, h, WHITE);
    // moverlo
```

Pantalla LCD 84x48

```

        iconos[f][YPOS] += iconos[f][DELTAY];
        // si no está, reinicia
        if (icons[f][YPOS] > display.height()) {
            icons[f][XPOS] = random(display.width());
            icons[f][YPOS] = 0;
            iconos[f][DELTAY] = random(5) + 1;
        }
    }
}

void testdrawchar(void) {
    display.setTextSize(1);
    display.setTextColor(BLACK);
    display.setCursor(0,0);

    for (uint8_t i=0; i < 168; i++) { if
        (i == '\n') continue;
        display.write(i);
        //if ((i > 0) && (i % 14 == 0))
            //display.println();
    }
    display.display();
}

void testdrawcircle(void) {
    for (int16_t i=0; i<display.height(); i+=2) {
        display.drawCircle(display.width()/2, display.height()/2, i,
        BLACK);
        display.display();
    }
}

void testfillrect(void) {
    uint8_t color = 1;
    for (int16_t i=0; i<display.height()/2; i+=3) {
        // colores alternativos
        display.fillRect(i, i, display.width()-i*2,
        display.height()-i*2, color%2);
        display.display();
    }
}

```

Pantalla LCD 84x48

```

        color++;
    }
}

void testdrawtriangle(void) {
    for (int16_t i=0; i<min(display.width(),display.height())/2;
        i+=5) {
        display.drawTriangle(display.width()/2, display.height()/2-i,
                             display.width()/2-i, display.height()/2+i,
                             display.width()/2+i, display.height()/2+i,
        NEGRO);
        display.display();
    }
}

void testfilltriangle(void) {
    uint8_t color = BLACK;
    for (int16_t i=min(display.width(),display.height())/2; i>0;
        i-=5) {
        display.fillTriangle(display.width()/2, display.height()/2-i,
                             display.width()/2-i, display.height()/2+i,
                             display.width()/2+i, display.height()/2+i,
        color);
        if (color == WHITE) color = BLACK;
        else color = WHITE;
        display.display();
    }
}

void testdrawroundrect(void) {
    for (int16_t i=0; i<display.height()/2-2; i+=2) {
        display.drawRoundRect(i, i, display.width()-2*i,
        display.height()-2*i, display.height()/4, BLACK);
        display.display();
    }
}

void testfillroundrect(void) {
    uint8_t color = NEGRO;
    for (int16_t i=0; i<display.height()/2-2; i+=2) {
        display.fillRoundRect(i, i, display.width()-2*i,

```

Pantalla LCD 84x48

```

display.height()-2*i, display.height()/4, color);
    if (color == WHITE) color = BLACK;
    else color =
    BLANCO;
    display.display();
}
}

void testdrawrect(void) {
    for (int16_t i=0; i<display.height()/2; i+=2) {
        display.drawRect(i, i, display.width()-2*i,
display.height()-2*i, BLACK);
        display.display();
    }
}

void testdrawline() {
    for (int16_t i=0; i<display.width(); i+=4) {
        display.drawLine(0, 0, i, display.height()-1, BLACK);
        display.display();
    }
    for (int16_t i=0; i<display.height(); i+=4) {
        display.drawLine(0, 0, display.width()-1, i, BLACK);
        display.display();
    }
    delay(250);

    display.clearDisplay();
    for (int16_t i=0; i<display.width(); i+=4) {
        display.drawLine(0, display.height()-1, i, 0, BLACK);
        display.display();
    }
    for (int8_t i=display.height()-1; i>=0; i-=4) {
        display.drawLine(0, display.height()-1, display.width()-1, i,
BLACK);
        display.display();
    }
    delay(250);

    display.clearDisplay();
    for (int16_t i=display.width()-1; i>=0; i-=4) {
        display.drawLine(display.width()-1, display.height()-1, i, 0,

```

Pantalla LCD 84x48

```

NEGRO);
    display.display();
}
for (int16_t i=display.height()-1; i>=0; i-=4) {
    display.drawLine(display.width()-1, display.height()-1, 0, i,
BLACK);
    display.display();
}
delay(250);

display.clearDisplay();
for (int16_t i=0; i<display.height(); i+=4) {
    display.drawLine(display.width()-1, 0, 0, i, BLACK);
    display.display();
}
for (int16_t i=0; i<display.width(); i+=4) {
    display.drawLine(display.width()-1, 0, i, display.height()-1,
BLACK);
    display.display();
}
delay(250);
}

void setup() {
    Serial.begin(9600);

    display.begin();
    // init realizado

    // puedes cambiar el contraste para adaptar la visualización
    // ¡para una mejor visualización!
    display.setContrast(50);

    display.display(); // mostrar splashscreen
    delay(2000);
    display.clearDisplay(); // borra la pantalla y el buffer

    // dibujar un único píxel
    display.drawPixel(10, 10, BLACK);
    display.display();
    retraso(2000);

```

Pantalla LCD 84x48

```
display.clearDisplay();

// dibujar muchas líneas
testdrawline();
display.display();
delay(2000);
display.clearDisplay();

// dibujar
rectángulos
testdrawrect();
display.display();
delay(2000);
display.clearDisplay();

// dibujar múltiples
rectángulos testfillrect();
display.display();
delay(2000);
display.clearDisplay();

// dibujar múltiples círculos
testdrawcircle();
display.display();
delay(2000);
display.clearDisplay();

// dibujar un círculo, 10 píxeles de radio
display.fillCircle(display.width()/2, display.height()/2, 10,
BLACK);
display.display();
delay(2000);
display.clearDisplay();

testdrawroundrect();
delay(2000);
display.clearDisplay();

testfillroundrect();
delay(2000);
display.clearDisplay();
```

Pantalla LCD 84x48

```
testdrawtriangle();
delay(2000);
display.clearDisplay();

testfilltriangle();
delay(2000);
display.clearDisplay();

// dibujar los primeros ~12 caracteres de la
fuente testdrawchar();
display.display();
delay(2000);
display.clearDisplay();

// pruebas de visualización de texto
display.setTextSize(1);
display.setTextColor(BLACK);
display.setCursor(0,0);
display.println("¡Hola, mundo!");
display.setTextColor(WHITE, BLACK); // texto
'invertido' display.println(3.141592);
display.setTextSize(2);
display.setTextColor(NEGRO);
display.print("0x"); display.println(0xDEADBEEF, HEX);
display.display();
retraso(2000);

// ejemplo de rotación
display.clearDisplay();
display.setRotation(1); // girar 90 grados en sentido
contrario a las agujas del reloj, también puede utilizar
valores de 2 y 3 para ir más lejos.
display.setTextSize(1);
display.setTextColor(BLACK);
display.setCursor(0,0);
display.println("Rotación");
display.setTextSize(2);
display.println("¡Ejemplo!");
display.display();
retraso(2000);

// volver a la no rotación
```

Pantalla LCD 84x48

```
display.setRotation(0);

// pantalla de mapa de bits en
miniatura display.clearDisplay();
display.drawBitmap(30, 16, logo16_glcd_bmp, 16, 16, 1);
display.display();

// invertir la visualización
display.invertDisplay(true);
delay(1000);
display.invertDisplay(false);
delay(1000);

// dibujar un icono bitmap y 'animar' el movimiento
testdrawbitmap(logo16_glcd_bmp, LOGO16_GLCD_WIDTH,
LOGO16_GLCD_HEIGHT);
}

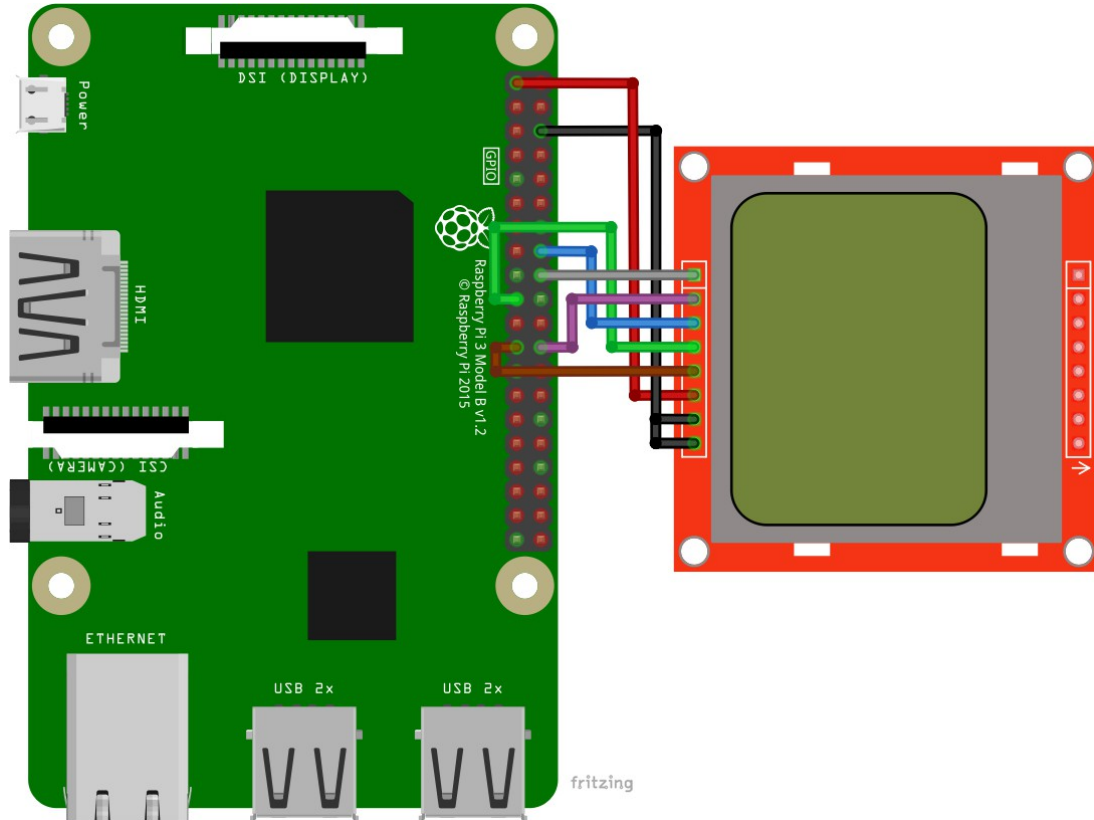
void loop() {

}
```

Pantalla LCD 84x48

Conexión de la pantalla con la Raspberry Pi

Conecta el módulo con la Raspberry Pi como se muestra en la siguiente imagen:



Pin de la pantalla	Pin Raspberry Pi	Pin físico	Color del cable
RST	GPIO24	18	gris
CE	GPIO8	24	morado
DC	GPIO23	16	azul
DIN	GPIO10	19	verde
CLK	GPIO11	23	marrón
VCC	3V3	1	rojo

Pantalla LCD

84x48

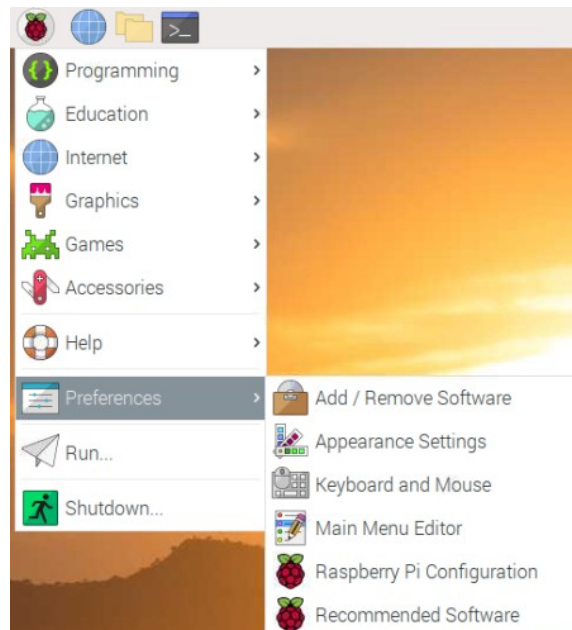
LUZ	GND	6	negro
GND	GND	6	negro

Pantalla LCD 84x48

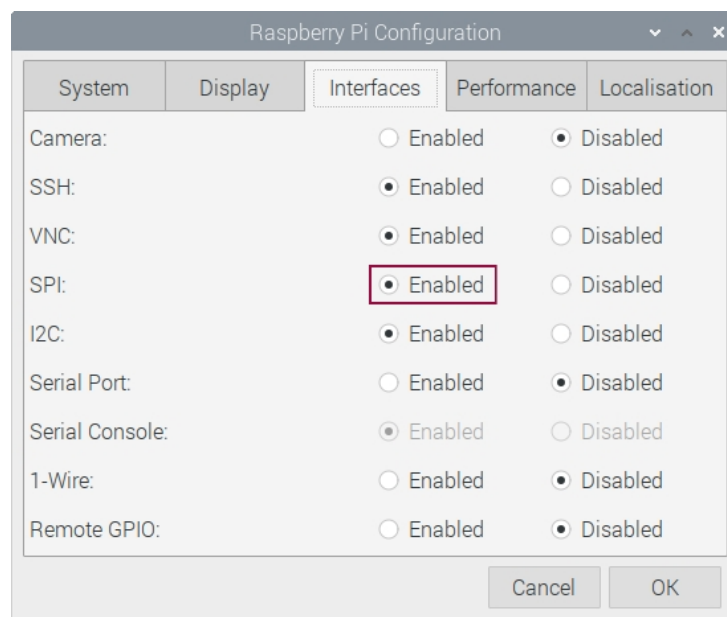
Activación de la interfaz SPI

Para utilizar el módulo con Raspberry Pi, la interfaz SPI de Raspberry Pi tiene que estar habilitada. Abra el siguiente menú:

Menú Aplicación > Preferencias > Configuración Raspberry Pi



En la nueva ventana, en la pestaña Interfaces, active el botón de opción SPI, como en la siguiente imagen:



Pantalla LCD 84x48

Bibliotecas y herramientas para Python

Para usar el módulo con la Raspberry Pi, hay que instalar varias librerías y herramientas. La librería que se va a utilizar se llama Adafruit_Nokia_LCD, pero antes hay que instalar algunas herramientas para Python.

Para instalar las herramientas para Python, abra el terminal y ejecute los siguientes comandos, uno a uno:

```
sudo apt-get update && sudo apt-get upgrade  
sudo apt-get install python3-setuptools  
sudo apt install python3-pip  
sudo pip3 install almohada
```

Para descargar la biblioteca, ejecute el siguiente comando:

```
git clone https://github.com/adafruit/Adafruit_Nokia_LCD.git
```

A continuación, cambie el directorio a la carpeta descargada:

```
cd Adafruit_Nokia_LCD
```

Para instalar la biblioteca ejecute el siguiente comando:

```
sudo python3 setup.py install
```

Pantalla LCD 84x48

Script Python

```
tiempo de importación

import Adafruit_Nokia_LCD as LCD

import Adafruit_GPIO.SPI as SPI

from PIL import Image

from PIL import

ImageDraw from PIL

import ImageFont

DC = 23

RST = 24

SPI_PORT = 0

SPI_DEVICE = 0

disp = LCD.PCD8544(DC, RST, spi=SPI.SpiDev(SPI_PORT, SPI_DEVICE,
max_speed_hz=4000000))

print('Pantalla LCD 84x48 script')

print('[Pulse Ctrl+C para finalizar el
programa.]')
```

Pantalla LCD
84x48
Inventario:

Pantalla LCD 84x48

`mientras sea verdad:`

```
disp.begin(contraste=60)
```

```
disp.clear()
```

```
disp.display()
```

```
image = Image.new('1', (LCD.LCDWIDTH, LCD.LCDHEIGHT))
```

```
draw = ImageDraw.Draw(imagen)
```

```
#una pestaña
```

```
draw.rectangle((0,0,LCD.LCDWIDTH,LCD.LCDHEIGHT), outline=255, fill=255)
```

```
draw.ellipse((2,2,22,22), outline=0, fill=255)
```

```
draw.rectangle((24,2,44,22), outline=0, fill=255)
```

```
draw.polygon([(46,22), (56,2), (66,22)], outline=0, fill=255)
```

```
draw.line((68,22,81,2), fill=0)
```

```
draw.line((68,2,81,22),
```

```
fill=0) print('Dibujando
```

```
formas')
```

```
fuente = ImageFont.load_default()
```

```
draw.text((8,30), 'AZ-Entrega', font=font)
```

```
print('Dibujando texto\n')
```

```
disp.image(imagen)
```

Pantalla LCD 84x48

```
time.sleep(1)

disp.display()

disp.clear()

print('Borrando la pantalla\n')

time.sleep(2)

except

KeyboardInterrupt:

print('\nFin del

script!') disp.clear()

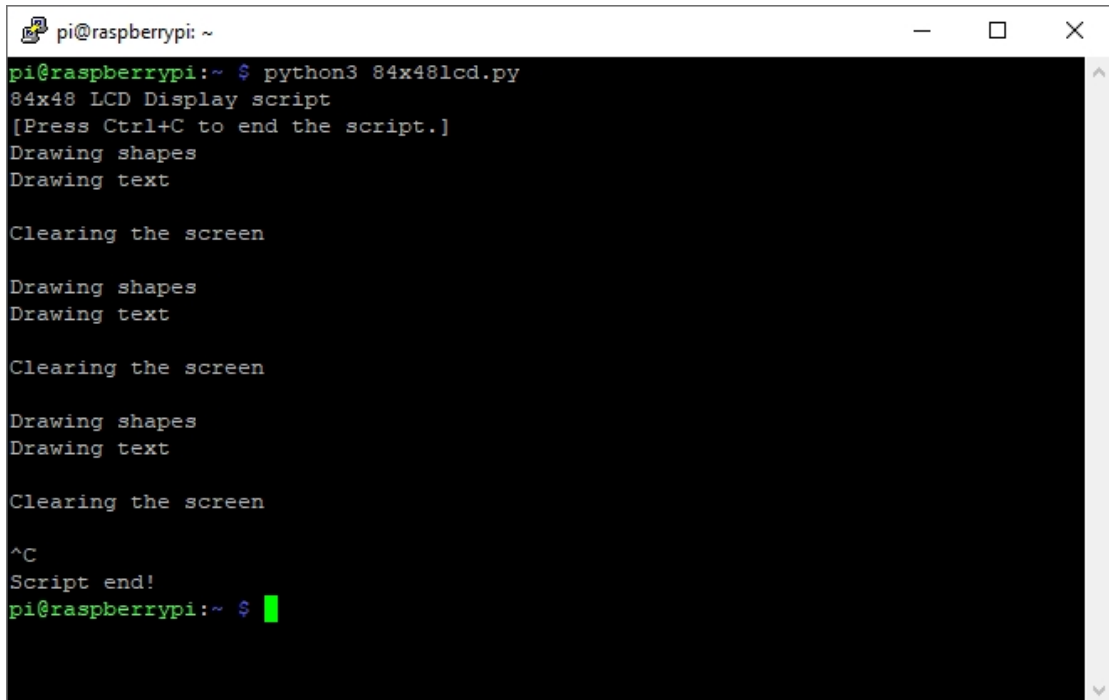
disp.display()
```

Pantalla LCD

84x48

Guarde el script con el nombre 84x48lcd.py. Para ejecutar el script, abra el terminal en el directorio donde está guardado el script y ejecute el siguiente comando: **python3 84x48lcd.py**

El resultado debería ser como el de la siguiente imagen:



```
pi@raspberrypi: ~  
pi@raspberrypi:~ $ python3 84x48lcd.py  
84x48 LCD Display script  
[Press Ctrl+C to end the script.]  
Drawing shapes  
Drawing text  
  
Clearing the screen  
  
Drawing shapes  
Drawing text  
  
Clearing the screen  
  
Drawing shapes  
Drawing text  
  
Clearing the screen  
  
^C  
Script end!  
pi@raspberrypi:~ $
```

Para detener el script pulse 'CTRL + C' en el teclado.

Pantalla LCD 84x48

Ahora es el momento de aprender y crear tus propios proyectos. Puedes hacerlo con la ayuda de muchos scripts de ejemplo y otros tutoriales que puedes encontrar en Internet.

Si está buscando productos de alta calidad para Arduino y Raspberry Pi, AZ-Delivery Vertriebs GmbH es el lugar adecuado para usted. Usted recibirá numerosos ejemplos de aplicación, instrucciones de instalación completas, libros electrónicos, bibliotecas y el apoyo de nuestros expertos técnicos.

<https://az-delivery.de>

¡Diviértete!

Impresionante

<https://az-delivery.de/pages/about-us>