

Последовательный эксперимент с динамическим дисплеем

LCD1602

LCD1602 Введение

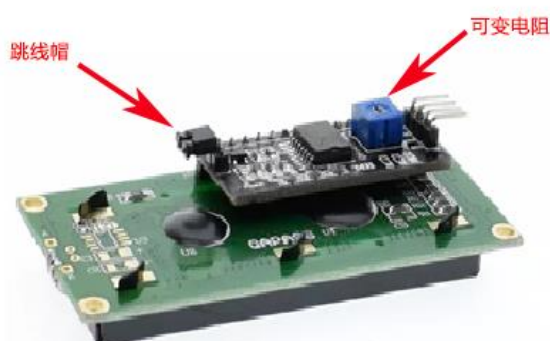
LCD1602 - это символьный ЖК-модуль, специально используемый для отображения букв, цифр и символов. Широко используется в промышленности, например, электронные часы, индикаторы температуры. Большинство символьных ЖК-дисплеев на рынке представляют собой ЖК-чипы, основанные на символах HD44780, и принципы управления точно такие же. «1602» означает 2 строки и 16 символов в строке. Дисплей LCD1602 с платой адаптера использует интерфейс ИС и сохраняет много портов ввода / вывода Жидкокристаллический дисплей 1602 (жидкокристаллический дисплей 1602, далее именуемый 1602 ЖК-дисплей) - это обычный символьный ЖК-дисплей, он так назван поскольку он может отображать $16 * 2$ символа. Обычно используемый нами ЖК-дисплей 1602 интегрирован с чипом шрифтов. Из API, предоставляемого библиотекой классов LiquidCrystal, мы можем легко использовать ЖК-дисплей 1602 для отображения английских букв и некоторых символов. Перед использованием 1602 LCD нам нужно подключить его к Arduino.

В комплекте мы используем модуль ИС LCD1602 для интеграции микросхемы расширения ввода / вывода ИС PCA8574, что упрощает использование LCD 1602. Через двухпроводную шину ИС (последовательная линия синхронизации SCL, последовательная линия передачи данных SDA) Arduino может достичь цели управления дисплеем LCD 1602. Не только упрощает схему, но и сохраняет порт ввода / вывода, так что Arduino может выполнять больше функций. Контрастность ЖК-дисплея также можно регулировать с помощью потенциометра на модуле. Вы также можете установить адрес, установив перемычки: 0x20-0x27. Позволяет Arduino управлять несколькими ЖК 1602.

На задней части модуля виден синий потенциометр, поверните его, чтобы отрегулировать контрастность 1602 LCD. Разъемами на задней панели являются GND,

VCC, SDA, SCL (SDA и SCL - это линия передачи данных и линия синхронизации iic Communication соответственно). Физическая схема LCD1602:

Примечание. Если подсветка ЖК-дисплея слишком темная, вы можете настроить синий переменный резистор на задней панели ЖК-дисплея (примечание: подсоедините перемычку на задней панели)



● Экспериментальный принцип

Подключите последовательный ЖК-экран LCD1602 через главную плату управления Arduino UNO R3 и используйте коммуникацию I2C для управления ЖК-экраном LCD1602 для отображения символов.

● Цель эксперимента

Используйте главную плату управления Arduino UNO R3 для управления жидкокристаллическим дисплеем LCD1602 для отображения символов.

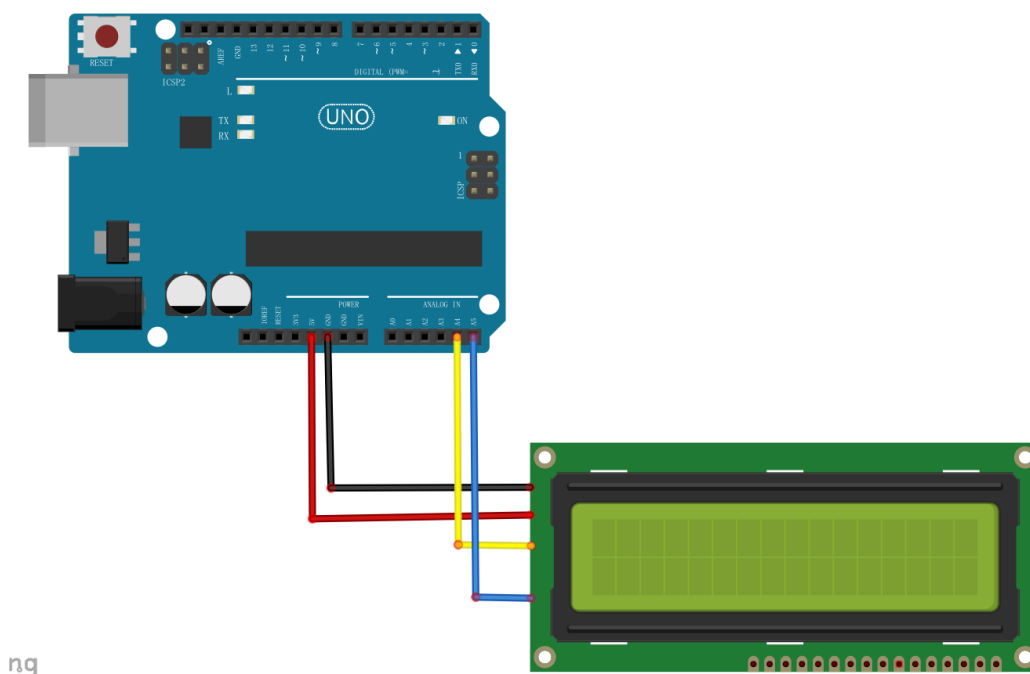
◆ Компоненты

- ◆ Материнская плата Keywish Arduino UNO R3
- ◆ Макетная плата
- ◆ USB-кабель для передачи данных
- ◆ Дисплей LCD1602 с платой адаптера
- ◆ Несколько перемычек

Проводка

Сначала припаяйте плату адаптера к ЖК-дисплею

| LCD1602модуль | Arduino |
|---------------|---------|
| GND | GND |
| VCC | 5V |
| SDA | A4 |
| SCL | A5 |



● Программа

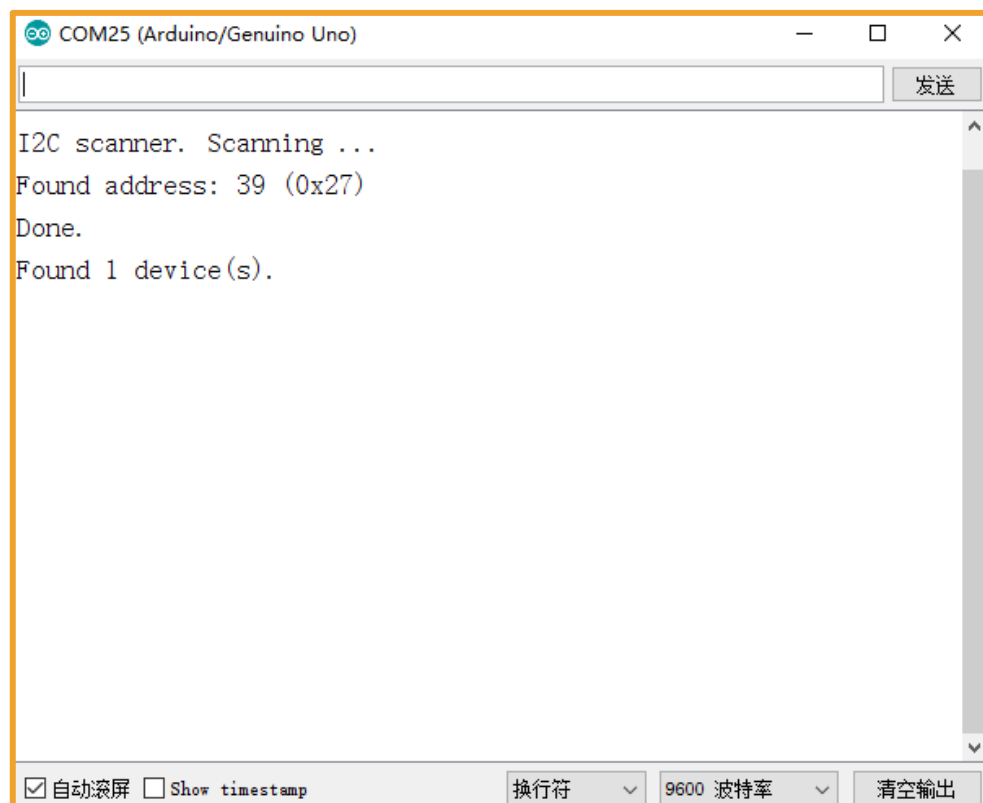
Сканирование адреса I2C

Скопируйте следующий код в Arduino IDE и выполните его. Затем выберите Tools->Serial Monitor и измените скорость передачи в правом нижнем углу на 9600, чтобы прочитать адрес I2C. Процедура следующая. По умолчанию адрес многих адаптерных плат I2C равен 0x27, но для того, чтобы иметь возможность подтвердить свое собственное преобразование I2C Лучше сканировать адрес платы самостоятельно, чтобы избежать ошибок в последующих экспериментах.

```
#include <Wire.h>

void setup() {
    Serial.begin (9600); // Leonardo: wait for serial port to connect
    while (!Serial) { }
    Serial.println ();
}
```

```
Serial.println ("I2C scanner. Scanning ...");  
byte count = 0;  
Wire.begin();  
for (byte i = 8; i < 120; i++) {  
  Wire.beginTransmission (i);  
  if (Wire.endTransmission () == 0) {  
    Serial.print ("Found address: ");  
    Serial.print (i, DEC);  
    Serial.print (" (0x");  
    Serial.print (i, HEX);  
    Serial.println ("");  
    count++;  
    delay (1); // maybe unneeded?  
  } // end of good response  
} // end of for loop  
Serial.println ("Done.");  
Serial.print ("Found ");  
Serial.print (count, DEC);  
Serial.println (" device(s).");  
} // end of setup  
void loop() {}
```



Адрес I2C, сканируемый последовательным монитором, равен 0x27.

Программа записи символьного дисплея

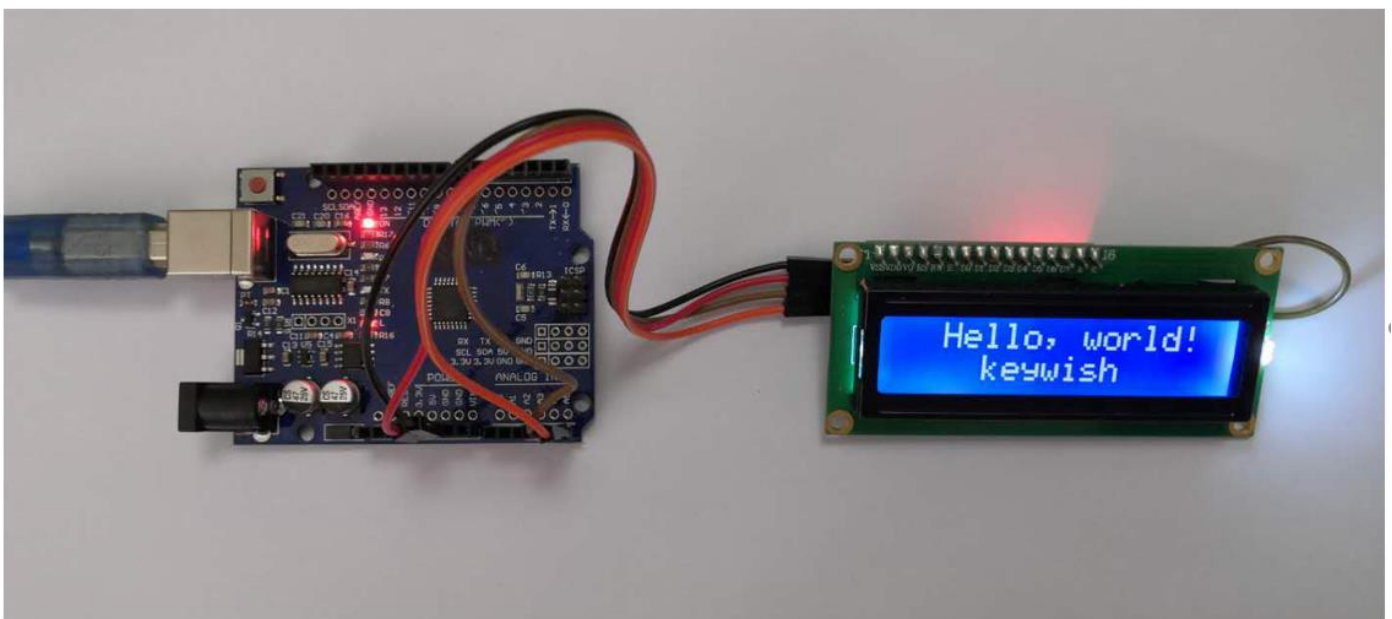
```
#include <Wire.h>
#include "LiquidCrystal_I2C.h"

LiquidCrystal_I2C lcd(0x27,16,2); // set the LCD address to 0x27 for a 16 chars and 2
line display

void setup()
{
    lcd.init(); // initialize the lcd
    // Print a message to the LCD.
    lcd.backlight();
    lcd.setCursor(2,0); // go to start of 2nd line
    lcd.print("Hello, world!");
    lcd.setCursor(4,1); // go to start of 2nd line
    lcd.print("keywish");
}

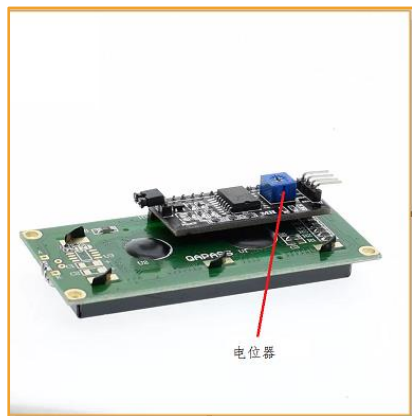
void loop()
{
}
```

● Результаты эксперимента








LCD1602 отображает общие проблемы и решения

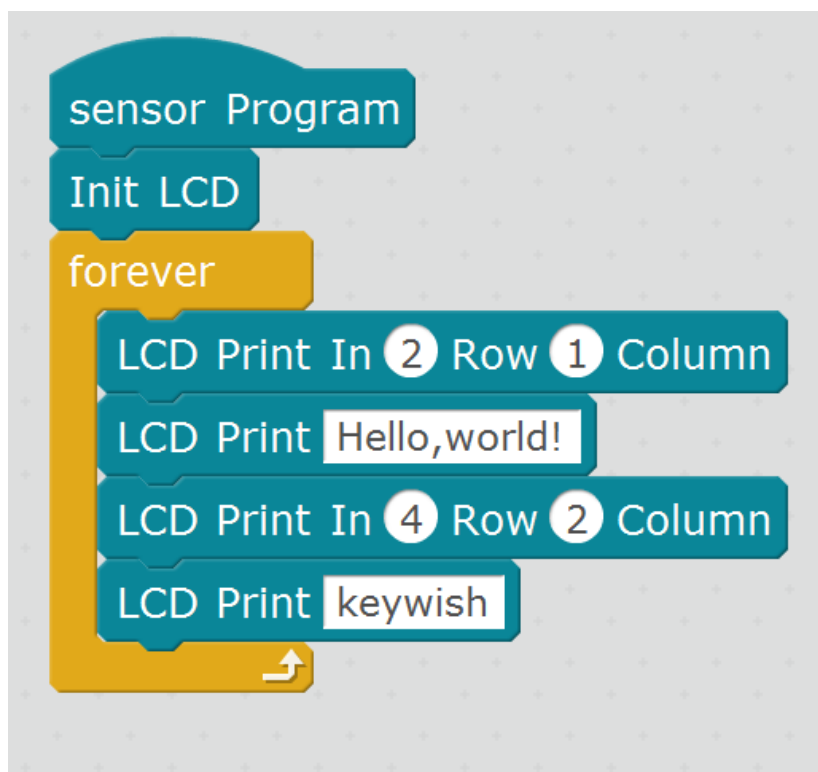
- 1) Если после загрузки программы подсветка включается, но символы не отображаются, попробуйте отрегулировать потенциометр за платой адаптера, чтобы отрегулировать яркость и эффект отображения.



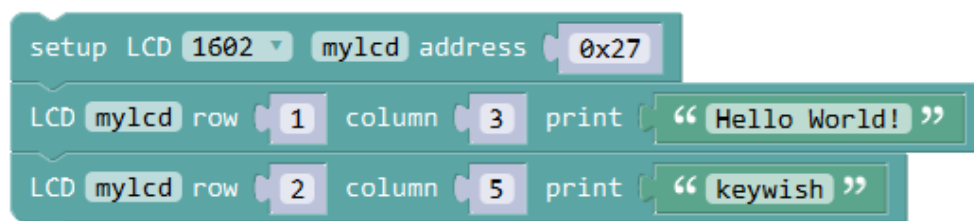
Если подсветка включается после загрузки программы, но отображаются только некоторые символы, это связано с различными версиями используемых микросхем. Точно так же адрес по умолчанию PCF8574 - 0x27.

● Программа графического программирования mBlock

-  - инициализировать экран дисплея LCD1602;
-   - Символы Hello, world отображаются на ЖК-экране в первом столбце первого ряда! ;
-   - отображение ключа на ЖК-дисплее в третьем столбце второго ряда;



- Программа графического программирования Mixly



- Программа графического программирования MagicBlock

MagicBlock пишет экспериментальную программу с последовательным ЖК-дисплеем, как показано на следующем рисунке:

