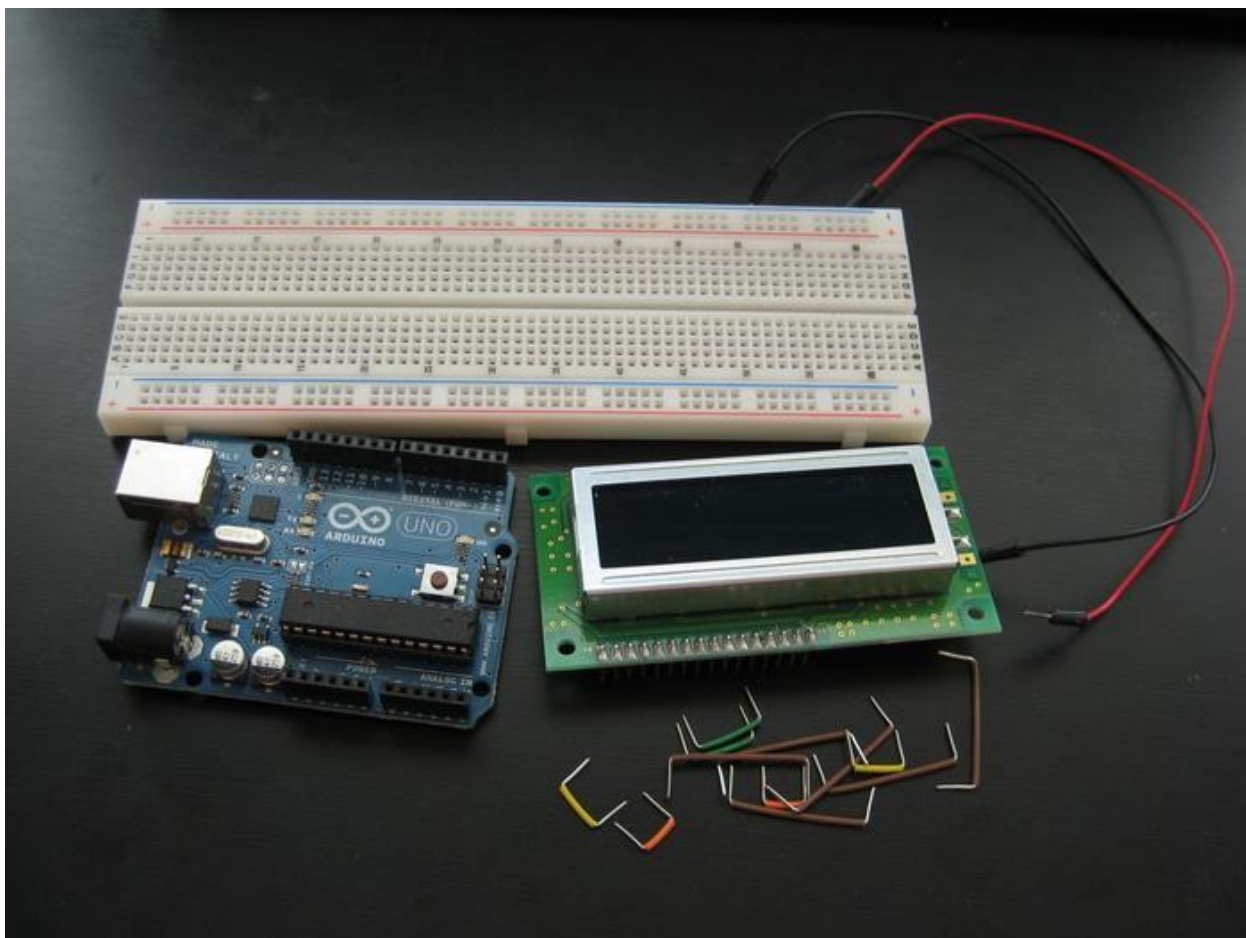


«Подключение текстового экрана к Arduino»

Список деталей:

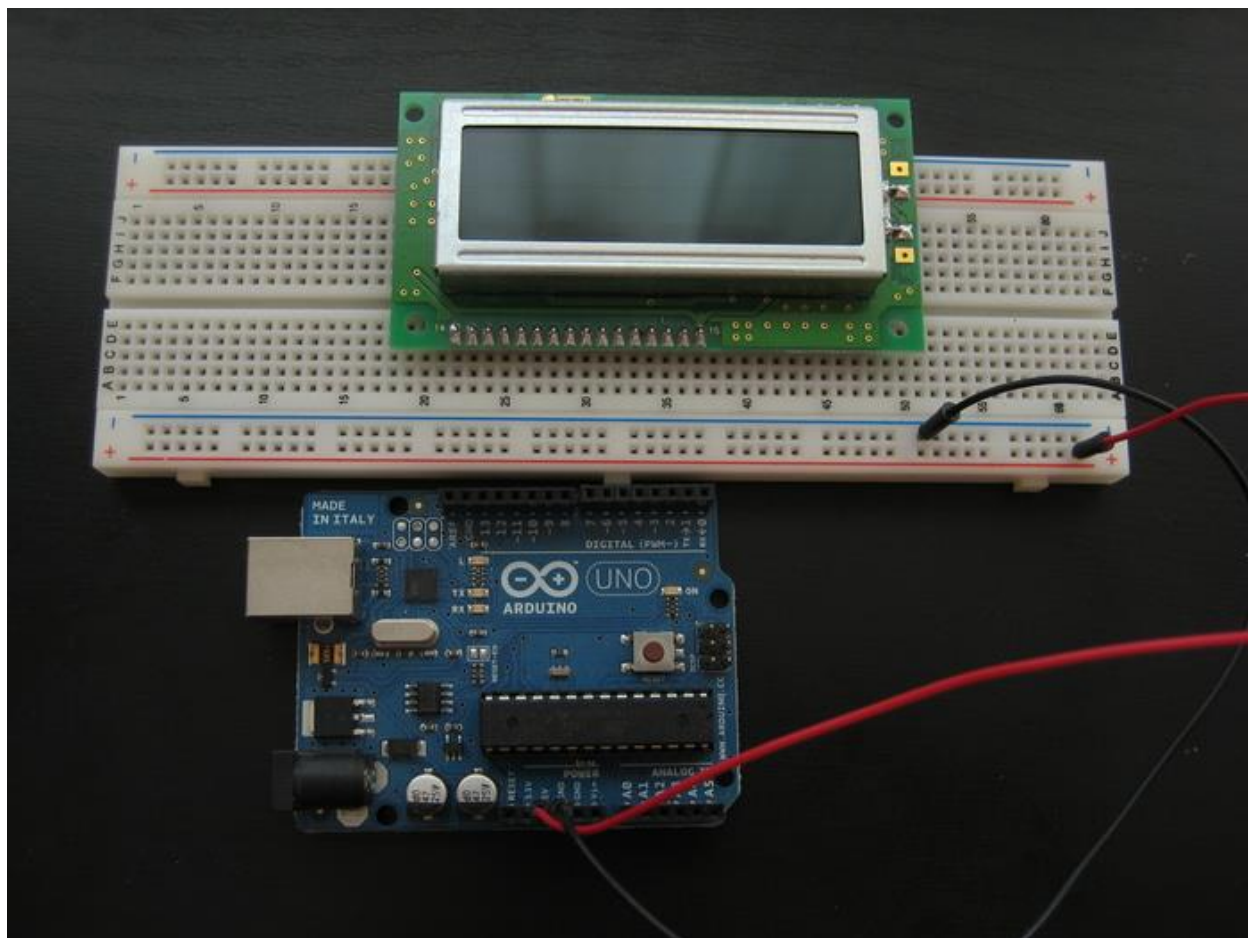
- плата Arduino Uno;
- макетная плата;
- провода папа-папа;
- кабель USB;
- текстовый экран 16×2 (Мэлт).



При проектировании автономных устройств, часто возникают ситуации, когда необходимо отображать сложную информацию (показания датчиков, текущее время, настройки и т.д.). Для этих целей удобно использовать жидкокристаллические дисплеи (LCD).

Для работы с LCD в среде Arduino IDE есть стандартная библиотека LiquidCrystal в которой удобно реализованы функции вывода текста и произвольных символов на экран.

Подключение

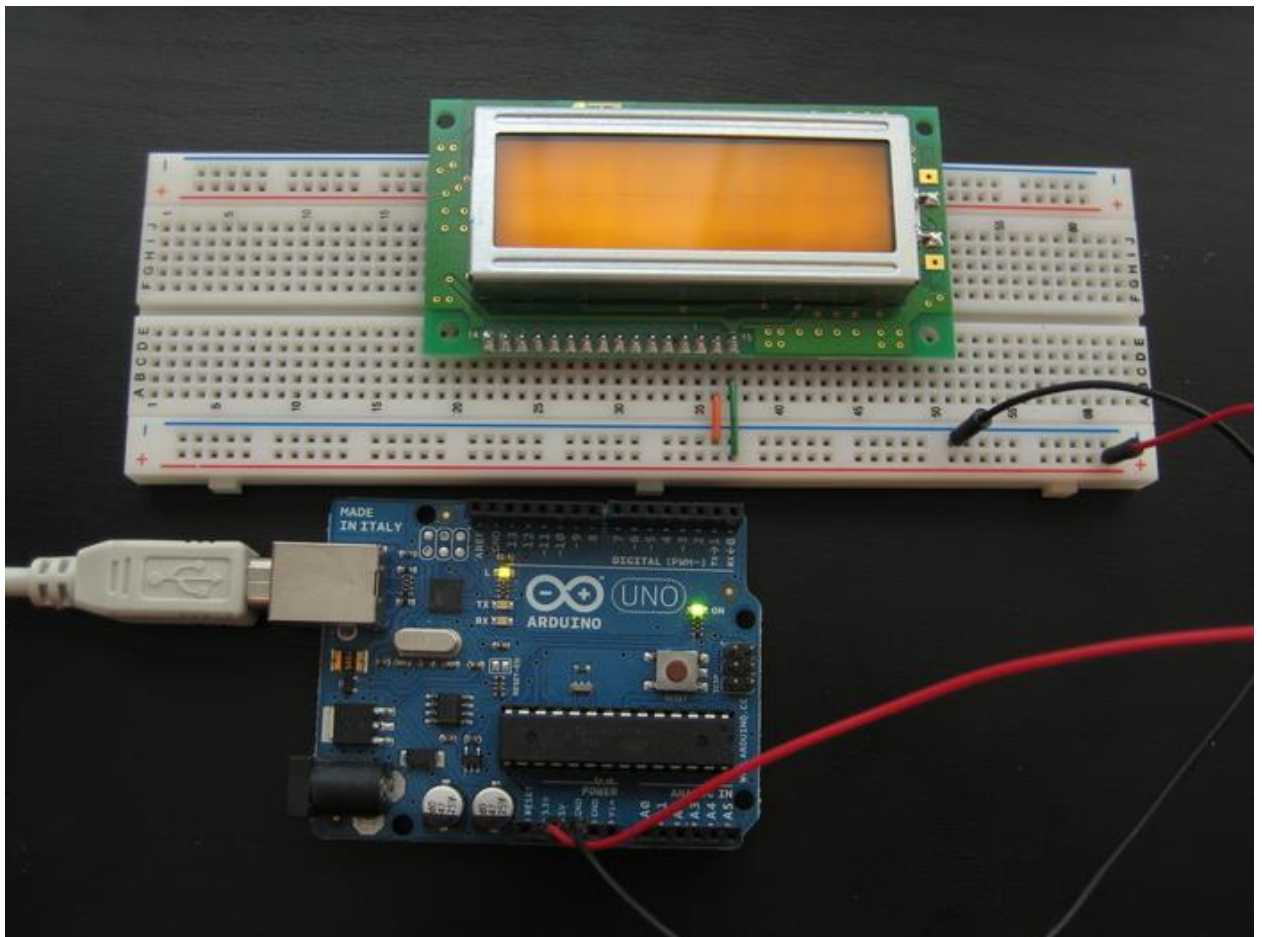


Закрепите экран на макетной плате и подведите к рельсам питания макетной платы питание +5 В и землю с Arduino.

Включение подсветки

Фоновая подсветка дисплея — это отдельный контур, не связанный с остальным. Включить её можно подав +5 В на 15-й контакт дисплея и подключив 16-й контакт к земле. Соединив эти два контакта с соответствующими рельсами, можно включить Arduino и увидеть, что дисплей засветился.

Обратите внимание, что на некоторых моделях нумерация контактов идёт не просто справа-налево от первого до шестнадцатого, а несколько более хитро. Так, например, на экране 16×2 от Мэлт первый контакт физически находится на 14-й позиции, второй на 13-й и так далее справа-налево вплоть до 14-го на первой позиции, а 15-й и 16-й расположены справа. Нумерация около контактов дисплея поможет не запутаться.

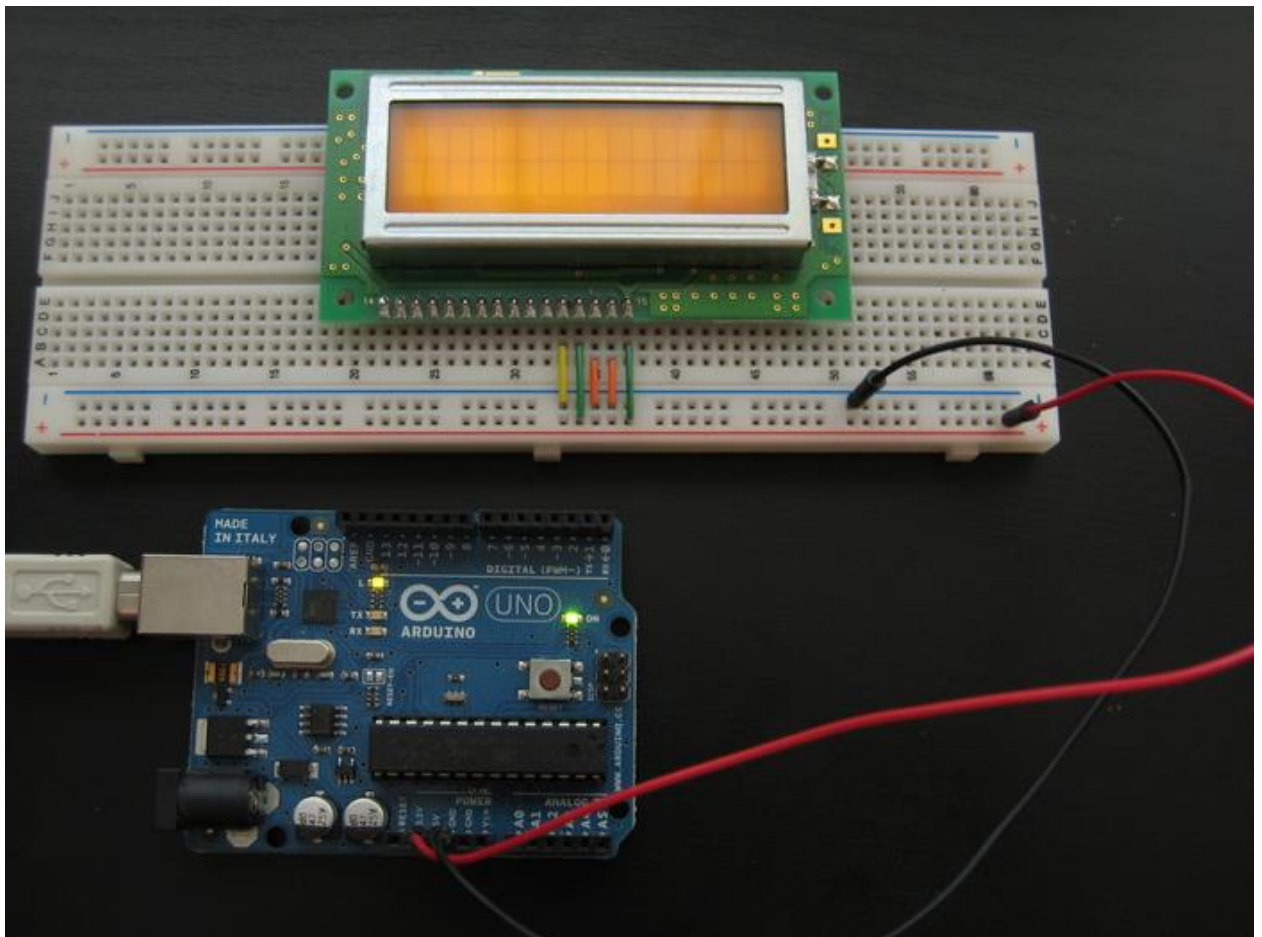


Включение питания знаковинтезатора

Далее необходимо подключить цепь, отвечающую за отображение символов. Для этого предназначены контакты 1, 2 и 3 на дисплее. Перед подключением отключите Arduino от питания.

- Первый — это земля. Соедините его с рельсой земли.
- Второй — питание. Соедините его с рельсой +5 В.
- Третий — контрастность. Для получения максимально контрастного изображения соедините его с рельсой земли. Вы можете подать на этот контакт произвольное напряжение от 0 до 5 В, чем оно выше, тем тусклее будет изображение, но вместе с этим снизится энергопотребление. Для возможности плавной регулировки контрастности можете подать на этот контакт выходной сигнал потенциометра.

После подключения, если включить Arduino, вы можете увидеть прямоугольные знакоместа. В зависимости от комбинации цветов текста и подсветки они могут быть как яркими и хорошо заметными, так и едва заметными. Это нормально: в любом случае, текст будет смотреться отлично.



Подключение шины данных

Для коммуникации между Arduino и экраном необходимо использовать несколько линий взаимодействия:

- 2 или 3 для командования дисплеем
- 4 или 8 для передачи данных (кодов символов и команд)

Таким образом занятыми окажутся от 6-ти до 11-ти контактов от обоих устройств. Если вам не требуется считывать с дисплея, что подходит под большинство сценариев использования, для команд понадобится 2 линии.

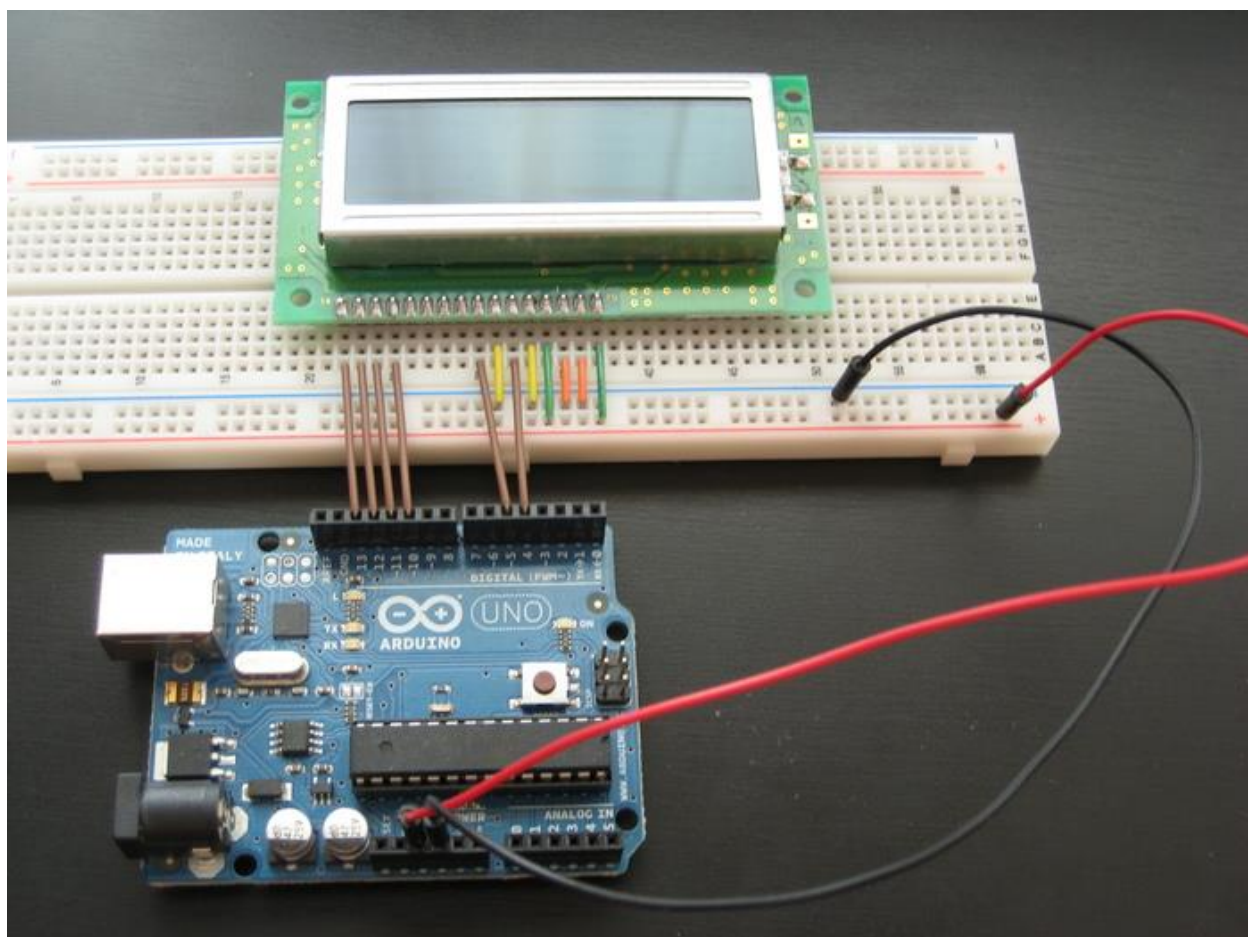
Если скорость обновления данных так же не является проблемой, для передачи данных достаточно 4-х линий.

Итак, для подключения дисплея достаточно использовать 6 линий, 6 контактов на Arduino. Рассмотрим именно этот сценарий.

Как упоминалось, нам не за чем считывать с дисплея, мы будем в него только писать. Поэтому соединим 5-й контакт дисплея, который отвечает за выбор чтение/запись с рельсой земли. Это означает «всегда писать».

Затем, соединяем Arduino и экран нашими 6-ю линиями коммуникации. Какие именно контакты будут выбраны на Arduino не имеет значения: мы зададим их в программе, но для примера была выбрана такая конфигурация:

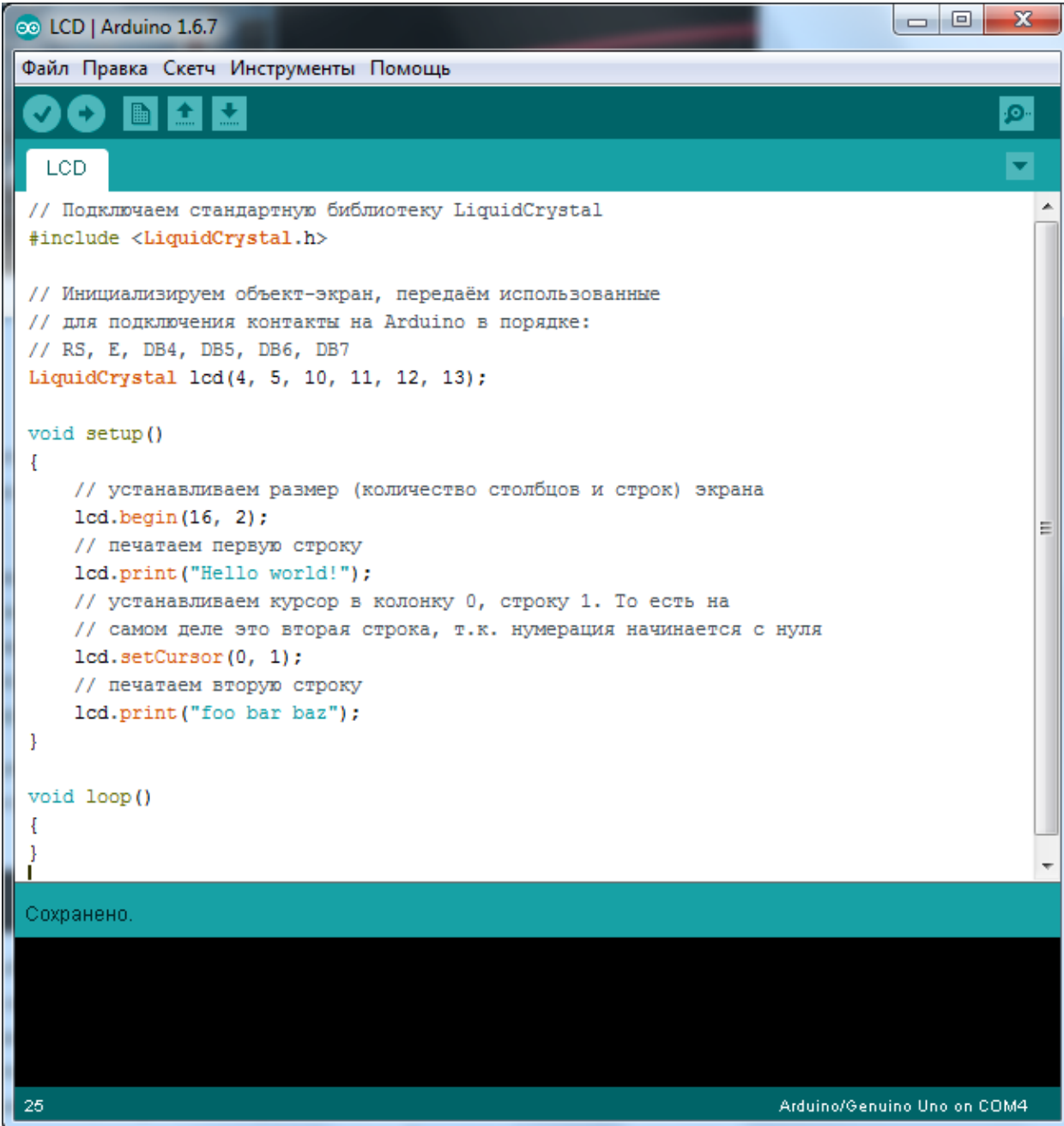
- 4-й контакт дисплея — 4-й контакт Arduino. Это линия адресного сигнала. Известная как A0 или RS. В зависимости от того, 0 она или 1, дисплей понимает имеем ли мы на линии данных команду вроде «передвинуть курсор» или код символа для отображения.
- 6-й контакт дисплея — 5-й контакт Arduino. Это линия разрешения доступа к данным. Известная, как E или Enable. Когда эта линия становится единицей, дисплей исполняет команду или выводит символ с линии данных.
- 11-й, 12-й, 13-й, 14-й контакт дисплея — 10-й, 11-й, 12-й, 13-й контакт Arduino соответственно. Это линии данных. Известные как DB4, DB5, DB6, DB7.



Экран подключен и готов к приёму данных. Осталось написать программу для Arduino.

Программирование

Для вывода текста с Arduino удобнее всего воспользоваться встроенной библиотекой Liquid Crystal. Для вывода приветствия и таймера, воспользуйтесь кодом вроде этого:

The image shows a screenshot of the Arduino IDE interface. The window title is "LCD | Arduino 1.6.7". The menu bar includes "Файл", "Правка", "Скетч", "Инструменты", and "Помощь". The toolbar contains icons for file operations and execution. The main editor area shows the following code:

```
// Подключаем стандартную библиотеку LiquidCrystal
#include <LiquidCrystal.h>

// Инициализируем объект-экран, передаём использованные
// для подключения контакты на Arduino в порядке:
// RS, E, DB4, DB5, DB6, DB7
LiquidCrystal lcd(4, 5, 10, 11, 12, 13);

void setup()
{
  // устанавливаем размер (количество столбцов и строк) экрана
  lcd.begin(16, 2);
  // печатаем первую строку
  lcd.print("Hello world!");
  // устанавливаем курсор в колонку 0, строку 1. То есть на
  // самом деле это вторая строка, т.к. нумерация начинается с нуля
  lcd.setCursor(0, 1);
  // печатаем вторую строку
  lcd.print("foo bar baz");
}

void loop()
{
}
```

At the bottom of the editor, a status bar displays "Сохранено." (Saved.) on the left, "25" in the center, and "Arduino/Genuino Uno on COM4" on the right.

Кириллица

Информация в этом разделе относится именно к дисплеям от Мэлт. Китайские и Европейские аналоги вряд ли имеют в своём наборе символов кириллицу, обратитесь к документации на дисплей, чтобы узнать об этом подробнее.

Вывод русских букв не совсем тривиален: вы не можете просто написать `lcd.print("Вася")`. Это связано с понятием кодировок. Вы знаете, что каждому символу соответствует код и при компиляции программы, если строка содержит кириллицу, она будет сконвертирована в коды по таблице utf-8, cp-1251 или какой-то другой в зависимости от настроек компилятора. Экран в свою очередь, ожидает увидеть данные в собственной кодировке.

Так например, букве «Я» соответствует код В1 в шестнадцатиричной системе. Чтобы передать на экран строку «Яndex», необходимо в явном виде с помощью последовательности `\x##` встроить в строку код символа:

```
lcd.print("\xB1ndex");
```