# СБОРНИК Примеров





В данном сборнике мы постарались в краткой форме описать работу Arduino и ее базовых компонентов. Исходники программ доступны в архиве, скачать который Вы можете на нашем сайте. AmperMarket.kz желает окунуться в интересный мир любительской робототехники!















**AmperMarket.kz** – интернет-магазин радиодеталей, работающий в Астане. Мы предлагаем широкий ассортимент продукции: платы Arduino, модули, датчики, провода, двигатели, резисторы, транзисторы и многое другое! Хотим отметить, что все рассматриваемые в данном сборнике компоненты Вы можете найти и купить их на нашем сайте. Рекомендуем приобрести один из наборов <u>Arduino Starter Kit</u>: во-первых, там есть практически все необходимое; во-вторых, это выйдет дешевле, нежели покупать комплектующие отдельно.



Arduino — это огромный конструктор, в котором нет конечного, определённого набора деталей, и нет ограничений в разнообразии того, что можно собрать. Всё ограничено лишь вашей фантазией. Это новый мир, убойное хобби и отличный подарок.



Arduino представляет собой небольшую плату с собственным процессором и памятью. На ней также есть пара десятков контактов, к которым можно подключать всевозможные компоненты: светодиоды, датчики, моторы, роутеры, магнитные дверные замки и вообще всё, что работает от электричества.



В процессор Arduino можно загрузить программу, которая будет управлять всеми этими устройствами по заданному алгоритму. Таким образом, можно создать бесчисленное количество оригинальных устройств, сделанных своими руками и по собственной задумке. Для того, чтобы понять идею, взгляните на иллюстрацию.



#### В чем преимущество Arduino?

Пользователь современного компьютера не задумывается о функционировании отдельных частей ПК. Он просто запускает нужные программы и работает с ними. Точно так же и Arduino позволяет пользователю сосредоточиться на разработке проектов, а не на изучении устройства и принципов функционирования отдельных элементов.

Наличие готовых модулей и библиотек программ позволяет непрофессионалам в электронике создавать готовые работающие устройства для решения своих задач. А варианты использования Arduino ограничены только возможностями микроконтроллера и имеющегося варианта платы, ну и, конечно, фантазией разработчика.





#### Сложно ли начать изучение?

Свою огромную популярность Arduino приобрела именно благодаря простоте и дружелюбности. Даже абсолютный ноль в программировании и робототехнике может освоить основы работы с Arduino за пару часов! Этому способствуют тысячи публикаций, учебников, заметок в Интернете и куча роликов на YouTube.

Для удобства работы с Arduino существует бесплатная официальная среда программирования «Arduino IDE», работающая под Windows, Mac OS и Linux. С помощью неё загрузка новой программы в Arduino становится делом одного клика, только лишь подключите плату к компьютеру через USB.



Вам не понадобится паяльник. Полноценные устройства можно собирать, используя специальную макетную доску, перемычки и провода абсолютно без пайки. Конструирование ещё не было таким быстрым и простым.





Arduino IDE – программное обеспечение для пользователей, позволяющее писать свои программы (скетчи) для платформы Arduino. Эта платформа в первую очередь ориентируется на конструкторов-любителей, которые применяют Arduino для построения простых систем автоматики и робототехники.

Язык программирования Arduino является стандартным C++ (используется компилятор AVR-GCC) с некоторыми особенностями, облегчающие написание программ новичкам в этом деле.



Преимущества Arduino IDE

- доступность
- удобный интерфейс
- совместимость с популярными ОС
- наличие необходимых инструментов
- варианты языков программирования
- возможность углубить знания С++
- встроенный набор примеров программ
- функции сохранения, экспорта, проверки, поиска, замены скетчей

¢

¢

#### Установка Arduino IDE

Сначала необходимо установить на компьютер интегрированную среду разработки Arduino – Arduino IDE. Для этого необходимо скачать программу с официального сайта разработчика:

https://www.arduino.cc/en/Main/Software

Выберите операционную систему, которая стоит на Вашем ПК и нажмите кнопку «Just download». Установка Arduino IDE с помощью инсталлятора избавит вас от большинства потенциальных проблем с драйверами и программным окружением. Если у Вас стоит Windows 7, то проделайте следующие шаги:

#### Download the Arduino IDE



#### ARDUINO 1.8.8

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on Processing and other opensource software.

This software can be used with any Arduino board. Refer to the <u>Getting Started</u> page for Installation instructions. Windows Installer, for Windows XP and up Windows ZIP file for non admin install

Windows app Requires Win 8.1 or 10

Mac OS X 10.8 Mountain Lion or newer

Linux 32 bits Linux 64 bits Linux ARM

Release Notes Source Code Checksums (sha512)



После того, как загрузили и установили среду разработки Arduino, давайте запустим её!

# Note: Additional data in the state of the second is a second in the second is a second in the second is a second if the second is a second second if the second if the second is a second second if the second if the second if the second is a second second if the second is a second second if the second is a second second if the second if the second is a second second if the second second if the second sec

Перед нами окно Arduino IDE. Обратите внимание — мы ещё не подключали нашу плату Arduino Uno к компьютеру, а в правом нижнем углу уже красуется надпись **«Arduino Uno on COM1**». Таким образом, Arduino IDE сообщает нам, что в данный момент она настроена на работу с целевой платой **Arduino Uno**. А когда придёт время, Arduino IDE будет искать Arduino Uno на порту **COM1**.

#### Подключение Arduino к компьютеру

После установки Arduino IDE пришло время **подключить** Arduino Uno к компьютеру.

Соедините Arduino Uno с компьютером через USB-кабель. Вы увидите, как на плате загорится светодиод «ON», и начнёт **мигать светодиод «L»**. Это означает, что на плату подано питание, и микроконтроллер Arduino Uno начал выполнять прошитую на заводе программу «Blink» (мигание светодиодом).

Чтобы настроить Arduino IDE на работу с Arduino Uno, нам необходимо узнать, какой номер COM-порта присвоил компьютер Arduino Uno. Для этого нужно зайти в **«Диспетчер устройств»** Windows и раскрыть вкладку **«Порты (COM и LPT)»**. Мы должны увидеть примерно следующую картину:



Операционная система распознала нашу плату Arduino Uno как СОМ-порт, подобрала для неё правильный драйвер и назначила этому СОМ-порту **номер 7**. Если мы подключим к компьютеру другую плату Arduino, то операционная система назначит ей другой номер. Поэтому, если у вас несколько плат Arduino, очень важно не запутаться в номерах СОМ-портов.

Если Ваша плата на **СН340G**, то необходимо скачать соответствующие драйвера:

#### https://ampermarket.kz/files/Driver-CH340G-Win-MAC-Linux.zip



#### Настройка Arduino IDE на работу с Arduino Uno

Теперь нам необходимо сообщить Arduino IDE, что плата, с которой ей предстоит общаться, находится на СОМ-порту «СОМ7».

Для этого переходим в меню **«Сервис»** → **«Последовательный порт»** и выбираем порт **«СОМ7»**. Теперь Arduino IDE знает – что-то находится на порту «СОМ7». И с этим «чем-то» ей вскоре предстоит общаться.

<b></b>	sketch_jan29a   Arduino 1.0.5-r2	2		×
Файл Правка Скетч	Сервис Справка			
sketch_jan29a	Автоформатирование Архивировать скетч Исправить кодировку и перезагрузить	Ctrl+T		.©. ▼
	Монитор порта	Ctrl+Shift+M		^
	Плата	•		_
	Последовательный порт	•	COM7	·
	Программатор Записать загрузчик	×		~
<b>‹</b> ل				>
1		Ardui	ino Uno on C	OM1

Чтобы у Arduino IDE не осталось никаких сомнений, необходимо прямо указать: «Мы будем использовать Arduino Uno!». Для этого переходим в меню **«Сервис»** — **«Плата»** и выбираем нашу **«Arduino Uno»**.

<b>8</b>	sketch_jan29a   Arduino 1.0.5-r2	-				
Файл Правка Скетч С	Сервис Справка					
sketch_jan29a	Автоформатирование Архивировать скетч Исправить кодировку и перезагрузить	Ctrl+T				
	Монитор порта	Ctrl+Shift+M		^		
	Плата	+	•	Arduino Uno		
	Последовательный порт	•		Arduino Duemilanove w/ ATmega328		
4	Программатор	÷		Arduino Diecimila or Duemilanove w/ ATmega168 Arduino Nano w/ ATmega328		
-	Записать загрузчик			Arduino Nano w/ ATmega168		
				Arduino Mega 2560 or Mega ADK		
				Arduino Mega (ATmega1280)		
				Arduino Leonardo		
				Arduino Esplora		
1		Arduino I		Arduino Micro		

Среда настроена, плата подключена. Теперь можно переходить к загрузке скетча.

Arduino IDE содержит очень много готовых примеров, в которых можно быстро подсмотреть решение какой-либо задачи. Есть в ней и простой пример **«Blink»** – выберем его.

00	sketch_jan29	a   Arduino 1.	0.5-r2 -	- 🗆	x I		
Фай.	п Правка Скетч Сервис Справка						
	Создать	Ctrl+N					
	Открыть	Ctrl+0			_		
	Папка со скетчами	•					
	Примеры	÷	01.Basics	•	Analog	gReadSerial	
	Закрыть	Ctrl+W	02.Digital	•	BareM	BareMinimum Blink DigitalReadSerial	
	Сохранить	Ctrl+S	03.Analog	•	Blink		
	Сохранить как	Ctrl+Shift+S	04.Communicatio	on ⊁	Digital		
	Загрузить	Ctrl+U	05.Control	•	Fade		
	Загрузить с помощью программатора	Ctrl+Shift+U	06.Sensors	•	ReadA	ReadAnalogVoltage	
	Настройки рецати	Ctrl+Shift+D	07.Display	- F			
		Ctrl+D	08.Strings	- <b>F</b>			
	TICHOID	Cultr	09.USB	- 1-			
	Настройки	Ctrl+Comma	10.StarterKit				
	Выход	Ctrl+Q	Q ArduinoISP		COM7		

Немного модифицируем код, чтобы увидеть разницу с заводским миганием светодиода.

Вместо строчки:delay(1000);напишем:delay(100);

В результате получим следующий код:

int led = 13; void setup() { pinMode(led, OUTPUT); }

void loop() {
 digitalWrite(led, HIGH);
 delay(100);
 digitalWrite(led, LOW);
 delay(100);
}



Теперь светодиод «L» должен загораться и гаснуть на десятую часть секунды. То есть в 10 раз быстрее, чем в заводской версии.

Загрузим наш скетч в Arduino Uno и проверим, так ли это? После загрузки светодиод начнёт мигать быстрее. Это значит, что всё получилось!

# Пример 1 Светодиоды

Светодиод (англ. Light Emitting Diode или просто LED) – энергоэффективная, надёжная, долговечная «лампочка».

Светодиод — вид диода, который светится, когда через него проходит ток от анода (+) к катоду (–).



### Пример 1 Светодиоды

#### Описание

Восемь светодиодов загораются один за другим, слева направо, затем так же затухают справа налево. После этого, светодиоды загораются справа налево, а затухают слева направо. Этот процесс повторяется бесконечное количество раз.



### Пример 1 Светодиоды

#### Схема



Скетч **«ex1»** смотрите в архиве **«Исходники»**, скачать который можете на сайте <u>ampermarket.kz</u>.

Кнопки являются распространенным компонентом, используемым для управления электронными устройствами. Они, как правило, используются в качестве переключателей для подключения или отключения цепи. На изображении снизу показаны ножки кнопки, стрелки одного цвета предназначены для подключения. Когда кнопка нажата, пины, изображенные синими стрелками, соединяются с пинами, изображенными красными стрелками.



#### Описание

В данном примере будем включать и выключать встроенный светодиод, используя I/O-порт и кнопку. I/O-порт представляет собой порт ввода (INPUT) и вывода (OUTPUT). Будем использовать функцию входного порта платы Arduino, чтобы прочитать вывод внешнего устройства. Так как плата имеет встроенный светодиодный индикатор (подключен к пину 13), для удобства будем использовать его.

Как правило, кнопка напрямую подключается к схеме для включения или выключения светодиода. Это соединение довольно простое. Однако иногда светодиод загорается автоматически, не нажимая кнопку, что вызвано различными **помехами.** Чтобы избежать этих внешних помех, подключим **резистор с сопротивлением 1k**Ω – 10kΩ между портом кнопки и GND (землей). Функция резистора заключается в том, чтобы потреблять внешние помехи при подключении к GND до тех пор, пока кнопка выключена.

При замыкании и размыкании между пластинами кнопки возникают микроискры, провоцирующие до десятка переключений за несколько миллисекунд. Явление называется дребезгом (англ. bounce). Это нужно учитывать, если необходимо фиксировать «клики».



#### Схема



Скетч **«ex2»** смотрите в архиве **«Исходники»**, скачать который можете на сайте <u>ampermarket.kz</u>.

# Пример 3 Переключатель

Ползунковый переключатель (тумблер, свитч) отличается от тактовой кнопки лишь тем, что у него есть фиксированные положения, а у тактовой кнопки – нет. Всё остальное идентично – контакты либо замыкаются, либо размыкаются; включить тумблер – это то же самое, что нажать и удерживать кнопку.



#### Пример 3 Переключатель

#### Описание

В данном примере подключим простейший переключатель SS12D00G4 к плате Arduino Uno, в Монитор порта выведем информацию о положении тумблера.



# Пример 3 Переключатель

#### Схема



Скетч **«ех3»** смотрите в архиве **«Исходники»**, скачать который можете на сайте <u>ampermarket.kz</u>.

#### Пример 4 Активный зуммер

Активный зуммер, именуемый также как «пищалка» или active buzzer — самый простой модуль для получения звука частотой около 2 кГц, что часто может понадобиться при работе c Arduino и в других проектах.

Для его работы достаточно подать на него напряжение **5 В**, так как внутри модуля имеется собственный генератор. Потребляемый ток **— 50 мА**.



#### Пример 4 Активный зуммер

#### Описание

В данном примере подключим активный зуммер к плате Arduino Uno, схема подключения и код программы простейшие.



#### Пример 4 Активный зуммер

#### Схема



Скетч **«ех4»** смотрите в архиве **«Исходники»**, скачать который можете на сайте <u>ampermarket.kz</u>.
### Пример 5 Пассивный зуммер

Пассивный зуммер — электромагнитная пищалка, используется для генерации звуковых сигналов различной частоты. Зуммер поддерживает шкалу из 8 различных звуков (каждый звук шкалы 0.5 с): До (523 Гц), Ре (587 Гц), Ми (659 Гц), Фа (698 Гц), Соль (784 Гц), Ля (880 Гц), Си (988 Гц) и тройное До (1047 Гц).

Пассивному зуммеру нужен не постоянный ток, а ток той частоты, на которой он будет «пищать». Преимущество пассивного зуммера в том, что ему можно менять частоту и делать «трели», «сирены» и пр.



### Пример 5 Пассивный зуммер

#### Описание

В данном примере подключим пассивный зуммер к плате Arduino Uno и проиграем популярную мелодию.



### Пример 5 Пассивный зуммер

#### Схема



Скетч **«ех5»** смотрите в архиве **«Исходники»**, скачать который можете на сайте <u>ampermarket.kz</u>.

### Пример 6 Фоторезистор

Фоторезистор – компонент, меняющий сопротивление в зависимости от количества света, падающего на него. В полной темноте он имеет максимальное сопротивление в сотни килоом, а по мере роста освещённости сопротивление уменьшается до десятков килоом. На его основе очень просто создать схему, которая бы поставляла данные об уровне освещённости в виде аналогового сигнала на управляющую электронику.



### Пример 6 Фоторезистор

#### Описание

Если интенсивность падающего света является высокой, сопротивление фоторезистора уменьшается; если низкой – увеличивается. В этом примере будем использовать восемь светодиодов для индикации интенсивности света. Чем выше интенсивность света, тем больше светодиодов горят. Когда интенсивность света достаточно высока, все светодиоды будут гореть. Когда нет света – все светодиоды гаснут.



### Пример 6 Фоторезистор

#### Схема



Скетч **«ех6»** смотрите в архиве **«Исходники»**, скачать который можете на сайте <u>ampermarket.kz</u>.

### Пример 7 RGB-светодиод

Трёхцветный светодиод или RGB-светодиод — это совмещённые в одном корпусе светодиоды красного, зелёного и синего цветов. Различают RGB-светодиоды с общим анодом и общим катодом. Используя широтно-импульсную модуляцию (PWM-сигнал) для всех анодов (катодов) одновременно, можно получить свечение произвольным цветом.

Характеристика	Красный	Зеленый	Синий
Максимальное прямое напряжение <b>(В)</b>	1,9	3,8	3,8
Сила тока (мА)	20	20	20





### Пример 7 RGB-светодиод

#### Описание

В данном примере будем использовать ШИМ для управления RGB-светодиодом и отображения различных цветов. В случае, если Вы используете RGB LED модуль, то подключение аналогичное, за исключением того, что дополнительно резисторы подключать не нужно: они уже припаяны на модуль.



### Пример 7 RGB-светодиод

#### Схема



Скетч **«ех7»** смотрите в архиве **«Исходники»**, скачать который можете на сайте <u>ampermarket.kz</u>.

7-сегментный цифровой LED индикатор – это индикатор, состоящий из семи светодиодов, установленных в форме цифры 8. Зажигая или выключая соответствующие LED-ы (сегменты) можно отображать цифры от нуля до девяти, а так же некоторые буквы.



74HC595N — восьмиразрядный сдвиговый регистр с последовательным вводом, последовательным или параллельным выводом информации, стриггером-защелкой и тремя состояниями на выходе. Самое распространенное применение данного регистра – экономия выходов микроконтроллера. Данный сдвиговый регистр позволяет управлять напряжением на своих восьми выходах, заняв всего три выхода микроконтроллера. Таким образом, количество рабочих выводов увеличивается на пять.

7/.110606

<b>/4</b> ΠC393							
1	Q1	VCC	16				
2	02	QO	15				
3	03	DS	14				
_4	Q4		13				
5	05	=0 90 T2	12				
6			11				
7	07		10				
8		MR	9				
	GND	Ų/'					

#### Описание

В данном примере продемонстрируем работу семисегментного индикатора с общим катодом и сдвигового регистра 74HC595N. Зажигая определённую комбинацию сегментов, мы получаем на экране определенный символ. Так, например, если зажечь сегменты В и С то загорится единичка. А, В, С – семёрка, и т. д. (см. схему сегментов индикатора на с. 45).

#### Компоненты



#### Схема



Скетч **«ех8»** смотрите в архиве **«Исходники»**, скачать который можете на сайте <u>ampermarket.kz</u>.

Четырёхразрядный цифровой индикатор — простое решение для проектов, где требуется отображать числа или ограниченный набор букв. Например, время или показания с сенсоров. Расположение и электросхема четырехразрядного индикатора представлены ниже:



**D1 ... D4** — разряды **а ... g** — сегменты

AMPERMARKET.KZ

#### Описание І

В данном примере с помощью четырехразрядного 7сегментного индикатора сделаем простой счетчик, нам понадобится библиотека **TimerOne**, она находится в каталоге примера в папке **libraries**. Можете скопировать данную библиотеку в папку: <u>C:\Program Files (x86)\Arduino\libraries</u> (путь по умолчанию).





#### Схема



Скетч **«ex9\_1»** смотрите в архиве **«Исходники»**, скачать который можете на сайте <u>ampermarket.kz</u>.

Индикатор	а	b	C	d	е	f	g	dp	D1	D2	D3	D4
Uno R3	2	3	4	5	6	7	8	9	13	12	11	10

#### Описание II

В прошлом примере нам пришлось использовать все цифровые выводы Uno, поэтому сейчас попробуем сэкономить их, используя сдвиговый регистр **74HC595N**. Программу переделаем так, чтобы индикатор показывал то значение, которое мы введем сами. В этом примере понадобится библиотека **Timer**, она находится в каталоге примера в папке **libraries**. Можете скопировать данную библиотеку в папку: <u>C:\Program Files (x86)\Arduino\libraries</u> (путь по умолчанию).



#### Схема



Скетч **«ex9\_2»** смотрите в архиве **«Исходники»**, скачать который можете на сайте <u>ampermarket.kz</u>.

#### Описание III

Наиболее простой способ подключить четырехразрядный индикатор – использовать готовый модуль. В этом примере понадобится библиотека **TM1637**, она находится в каталоге примера в папке libraries. Можете скопировать данную библиотеку в папку: C:\Program Files (x86)\Arduino\libraries (путь по умолчанию).

#### Компоненты



#### Схема



Скетч **«ex9\_3»** смотрите в архиве **«Исходники»**, скачать который можете на сайте <u>ampermarket.kz</u>.

Светодиодная матрица — это графический индикатор, который можно использовать для вывода простых изображений, букв и цифр. Расположение и электросхема светодиодной матрицы представлены ниже:



#### Описание І

В этом примере научимся пользоваться удобным матричным дисплеем 8×8 с помощью уже знакомого сдвигового регистра 74HC595N.



#### Схема



Скетч **«ex10\_1»** смотрите в архиве **«Исходники»**, скачать который можете на сайте <u>ampermarket.kz</u>.

#### Описание II

Гораздо проще использовать модуль матричного дисплея с микросхемой **MAX7219**, несколько модулей также удобно подключать в цепочку. Рассмотрим простейший пример подключения модуля к Arduino. Используемые библиотеки: **Adafruit\_GFX** и **Max72xxPanel**, они находятся в каталоге примера в папке **libraries**. Можете скопировать эти библиотеки в папку: <u>C:\Program Files (x86)\Arduino\libraries</u> (путь по умолчанию).

#### Компоненты



#### Схема



Скетч **«ex10\_2»** смотрите в архиве **«Исходники»**, скачать который можете на сайте <u>ampermarket.kz</u>.

### Пример 11 Светодиодная шкала

Научившись работать с одним, двумя, тремя и т. д. светодиодами, вам не составит труда разобраться с новым компонентом **«световая шкала»**, который представляет собой 10 независимых светодиодов в одном корпусе. Только в отличие от стандартного светодиода в виде шарика, мы видим десять полосок, которые могут включаться и выключаться.

У световой шкалы соответственно имеется 20 ножек под каждый анод и катод светодиода. Расположение и электросхема светодиодной шкалы представлены ниже:



### Пример 11 Светодиодная шкала

#### Описание

В данном примере научимся пользоваться светодиодной шкалой. Поворачивая потенциометр, Вы заметите, что количество включенных светодиодов на шкале будет изменяться.



### Пример 11 Светодиодная шкала

#### Схема



Скетч **«ex11»** смотрите в архиве **«Исходники»**, скачать который можете на сайте <u>ampermarket.kz</u>.

# Пример 12 ЖК-дисплей

В данном примере мы будем использовать популярный жидкокристаллический дисплей 1602 (Liquid Crystal Display), экран которого способен отображать одновременно до 32 символов (16 столбцов, 2 строки). Подключение к Arduino осуществляется по синхронному 8-битному параллельному интерфейсу.

#### Дисплей LCD1602 имеет 16 выводов:

- 1-VSS-GND («земля»);
- 2-VDD-(+5В питание);
- 3 VO установка контрасности;
- 4 RS (команды, данные);
- 5 RW (чтение, запись);
- 6 E (Enable);
- 7–14– (DB0...DB7) линия данных;
- 15 А (плюс подсветки);



## Пример 12 ЖК-дисплей

#### Описание І

Выведем на наш дисплей текстовую информацию. В этом примере понадобится библиотека LiquidCrystal, она находится в каталоге примера в папке libraries. Можете скопировать данную библиотеку в папку: <u>C:\Program Files</u> [x86]\Arduino\libraries (путь по умолчанию).



## Пример 12 ЖК-дисплей

#### Схема



Скетч **«ex12\_1»** смотрите в архиве **«Исходники»**, скачать который можете на сайте <u>ampermarket.kz</u>.

### Пример 12 ЖК-дисплей

#### Описание II

Гораздо проще использовать дисплей с I2C-модулем. Вместо 6 цифровых выходов будут использоваться лишь 2 аналоговых. Также, необходимо будет использовать библиотеку LiquidCrystal\_I2C, она находится в каталоге примера в папке libraries. Можете скопировать данную библиотеку в папку: <u>C:\Program Files (x86)\Arduino\libraries</u> (путь по умолчанию).



#### Компоненты



### Пример 12 ЖК-дисплей

#### Схема



Скетч **«ex12\_2»** смотрите в архиве **«Исходники»**, скачать который можете на сайте <u>ampermarket.kz</u>.

### Пример 13 Джойстик

Модуль джойстика по функционалу схож с «грибочком» в манипуляторах от консолей PlayStation, XBox. Он комбинирует в себе двухосный джойстик и тактовую кнопку.

По направлениям отклонения **H** (horizontal) и **V** (vertical), на плате джойстика установлены два потенциометра. Основываясь на их показаниях, джойстик отслеживает отклонения влево, вправо, вверх, вниз и выдает аналоговый сигнал от 0 до 1023.

Ось Z посажена на кнопку и выдает цифровой сигнал (вкл/выкл). Джойстик имеет пятипиновый разъем стандарта **2,54 мм**: VCC, Ground, X, Y, Key.



### Пример 13 Джойстик

#### Описание

В этом примере подключим модуль джойстика к Arduino. В Монитор порта выведем информацию о положении джойстика по осям X и Y, а также есть ли нажатие на него.

## Компоненты



### Пример 13 Джойстик

#### Схема



Скетч **«ex13»** смотрите в архиве **«Исходники»**, скачать который можете на сайте <u>ampermarket.kz</u>.
Чтобы заставить перемещаться заряженные частицы от одного полюса к другому необходимо создать между полюсами **разность потенциалов** (или **напряжение**). Единица измерения напряжения – **Вольт** (**B** или **V**). В формулах и расчетах напряжение обозначается буквой **V**. Чтобы получить напряжение величиной **1 B** нужно передать между полюсами заряд в **1 Кл**, совершив при этом работу в **1 Дж** (Джоуль).

Вольтметр – прибор для измерения напряжения (разности потенциалов) между двумя точками электрической схемы.



## Описание I

В этом примере с помощью Arduino сделаем простейший вольтметр на 5 В.



### Схема



Скетч **«ex14\_1»** смотрите в архиве **«Исходники»**, скачать который можете на сайте <u>ampermarket.kz</u>.

## Описание II

В данном примере будем использовать переменный резистор для деления напряжения.



### Схема



Скетч **«ex14\_2»** смотрите в архиве **«Исходники»**, скачать который можете на сайте <u>ampermarket.kz</u>.

# Пример 15 Термистор

Термистор — это переменный резистор, который меняет собственное сопротивление в зависимости от температуры.

По типу зависимости сопротивления от температуры различают терморезисторы с отрицательным (NTCтермисторы, от слов «Negative Temperature Coefficient») и положительным (PTC-термисторы, от слов «Positive Temperature Coefficient» или позисторы) температурным коэффициентом сопротивления (или TKC). Для позисторов – с ростом температуры растёт их сопротивление; для NTCтермисторов увеличение температуры приводит к падению их сопротивления.



## Пример 15 Термистор

### Описание

В этом примере с помощью термистора NTC 10К выведем на дисплей информацию о текущей температуре окружающей среды.



# Пример 15 Термистор

### Схема



Скетч **«ex15»** смотрите в архиве **«Исходники»**, скачать который можете на сайте <u>ampermarket.kz</u>.

# **Пример 16** Модуль DHT11

Модуль датчика DHT11 предназначен для снятия данных о температуре и влажности окружающей среды. Этот довольно популярный датчик не обладает высоким быстродействием и точностью, но зато прост, недорог и отлично подходит для обучения. Он выполнен из двух частей — емкостного датчика влажности и термистора. Чип, находящийся внутри, выполняет аналого-цифровое преобразование и выдает цифровой сигнал, который можно считать с помощью любого микроконтроллера.

Для работы с модулем DHT11 не потребуется какой-либо дополнительной обвязки, при питании от источника 5 В постоянного тока (справедливо для данной сборки, но не для датчика отдельно).



# **Пример 16** Модуль DHT11

### Описание

В этом примере продемонстрируем работу с данным модулем. Выведем значения температуры и влажности на ЖК-дисплей. Вам также будет необходимо использовать библиотеку dht11, она находится в каталоге примера в папке libraries. Можете скопировать данную библиотеку в папку: C:\Program Files (x86)\Arduino\libraries (путь по умолчанию).



# **Пример 16** Модуль DHT11

### Схема



Скетч **«ex16»** смотрите в архиве **«Исходники»**, скачать который можете на сайте <u>ampermarket.kz</u>.

# Пример 17 Модуль расстояния HC-SR04

Ультразвуковой модуль измерения расстояния HC-SRO4 может служить прекрасным датчиком для робота, благодаря которому он сможет определять расстояния до объектов, объезжать препятствия, или строить карту помещения. Его можно также использовать в качестве датчика для сигнализации, срабатывающего при приближении объектов.

Ультразвуковой дальномер определяет расстояние до объектов точно так же, как это делают дельфины или летучие мыши. Он генерирует звуковые импульсы на частоте 40 кГц и **слушает эхо**. По времени распространения звуковой волны туда и обратно можно однозначно определить расстояние до объекта.



# Пример 17 Модуль расстояния HC-SR04

## Описание

В этом примере продемонстрируем работу с модулем расстояния HC-SRO4. Выведем значение расстояния между ним и препятствием на ЖК-дисплей. Вам также будет необходимо использовать библиотеку **NewPing**, она находится в каталоге примера в папке **libraries**. Можете скопировать данную библиотеку в папку: C:\Program Files (x86)\Arduino\libraries (путь по умолчанию).



# Пример 17 Модуль расстояния HC-SR04

#### Схема



Скетч **«ex17»** смотрите в архиве **«Исходники»**, скачать который можете на сайте <u>ampermarket.kz</u>.

# **Пример 18** Датчик движения HC-SR501

**PIR** (пассивные инфракрасные) сенсоры позволяют улавливать движение. Очень часто используются в системах сигнализации. Эти датчики малые по габаритам, недорогие, потребляют мало энергии, легки в эксплуатации, практически не подвержены износу. Кроме PIR, подобные датчики называют пироэлектрическими и инфракрасными датчиками движения.

Инфракрасный датчик движения HC-SR501 позволяет обнаруживать движение человека или домашнего животного на расстоянии до 7 метров. Расстояние обнаружения можно регулировать. Имеет два входа питания (+5 В и Земля) и один цифровой выход, по которому можно снимать данные. Если помех нет – на нем будет высокий уровень (~3,3 В), если есть – низкий (~0 В).



# Пример 18 Датчик движения HC-SR501

### Описание

В этом примере продемонстрируем работу с датчиком движения. Светодиод будет загораться при обнаружении движения сенсором.



## **Пример 18** Датчик движения HC-SR501

### Схема



Скетч **«ex18»** смотрите в архиве **«Исходники»**, скачать который можете на сайте <u>ampermarket.kz</u>.

Объединение в один модуль нескольких тактовых кнопок упрощает конструкцию панели управления прибора и экономит порты Arduino или микроконтроллера. Матричная 16кнопочная клавиатура найдет применение в кодовых замках, пультах дистанционного управления и в других интересных проектах.

Принцип работы такой клавиатуры достаточно прост. Микроконтроллер поочередно подает логическую единицу на каждый из выводов 4 рядов, а с выводов столбцов наоборот считывает значения. Если нажать какую-либо из кнопок, то она замкнет вывод ряда и вывод столбца, с которыми связана, в результате чего, на соответствующем выводе столбца появится логическая единица.



Схема матричной клавиатуры:	
	<u>-</u>
	<u>c</u>
	<u>c</u>
	<u>c</u>

	Столбец 1 Столбец 2		бец 2 Стол	ібец 3 Столбец 4
Строка 1	┎╍╍	_ <del></del>		
	S1 ['0']	S2 ['1']	S3 ['2']	S4 ['3']
Строка 2	┎╍╍┙	┎╺╴╺		
	S5 ['4']	S6 ['5']	S7 ['6']	S8 ['7']
Строка 3	┎╍╺╍	┎╺╴╺		
	S9 ['8']	S10 ['9']	S11 ['A']	S12 ['B']
Строка 4	┎╦╦┓	┎╦╦┙	_ <del></del>	└┍ <del>╍</del> ┙
	S13 ['C']	S14 ['D']	S15 ['E']	S16 ['F']

Например, если нажать кнопку SW7, то она замкнет выводы Row2 и Col3. Нетрудно догадаться, что в таком случае, при подаче на Row2 логической единицы, через кнопку SW7 логическая единица установится и на выводе Col3.



Очевидно, что можно работать и в обратном порядке: подавать напряжение на выводы столбцов, а считывать с выводов рядов.

## Описание

В этом примере с помощью библиотеки **Keypad** (находится в nanke **libraries** примера) подключим матричную клавиатуру 4×4 к плате Arduino Uno, при нажатии на кнопку в Мониторе порта будет отображаться соответствующий символ.

## Компоненты



### Схема



Скетч **«ex19»** смотрите в архиве **«Исходники»**, скачать который можете на сайте <u>ampermarket.kz</u>.

# Пример 20 ИК датчик и пульт

Инфракрасный датчик VS1838В, работающий на частоте 38 кГц. Применяется в связке с ИК пультом. Инфракрасный пульт дистанционного управления «Car mp3» на 21 кнопку. Радиус действия: до 8-10 м.



# Пример 20 ИК датчик и пульт

## Описание

В данном примере научимся использовать ИК датчик для приема сигнала пульта дистанционного управления с помощью библиотеки IRremote (находится в каталоге примера в папке libraries).



# Пример 20 ИК датчик и пульт

### Схема



Скетч **«ex20»** смотрите в архиве **«Исходники»**, скачать который можете на сайте <u>ampermarket.kz</u>.

## **Пример 21** Модуль часов DS-1302

Многие проекты или задачи требуют точного временного исполнения. Например, в системе автополива могут быть несколько режимов: утренний полив, дневной и вечерний. Значит, для стабильной работы всей этой системы и всего рабочего цикла необходимо, чтобы система, построенная на Arduino, имела возможность точно определять текущее время. Данная функция в плате не была предусмотрена, поэтому такую проблему поможет решить **RTC модуль часов реального** времени.

Ориентация во времени очень полезна не только в автополиве, но и в других системах: включение света или отопления по расписанию, включение электрочайника по таймеру и пр. Рассмотрим работу модулей часов реального времени на примере **DS-1302**.



## **Пример 21** Модуль часов DS-1302

### Описание

В этом примере продемонстрируем работу с RTC-модулем DS-1302, используя библиотеку virtuabotixRTC (находится в папке libraries). Установим текущее время с помощью примера из библиотеки. Для питания модуля нам понадобится литиевая батарейка «таблетка» CR2032.



## **Пример 21** Модуль часов DS-1302

## Схема



Скетч **«ex21»** смотрите в архиве **«Исходники»**, скачать который можете на сайте <u>ampermarket.kz</u>.

Радиочастотная идентификация (RFID) – это технология автоматической бесконтактной идентификации объектов при помощи радиочастотного канала связи. Базовая система RFID состоит из:

- радиочастотной метки;
- считывателя информации (ридера);
- компьютера для обработки информации.

Идентификация объектов производится по уникальному цифровому коду, который считывается из памяти электронной метки, прикрепляемой к объекту идентификации. Считыватель содержит в своем составе передатчик и антенну, посредством которых излучается электромагнитное поле определенной частоты. Попавшие в зону действия считывающего поля радиочастотные метки «отвечают» собственным сигналом, содержащим информацию (идентификационный номер товара, пользовательские данные и т. д.). Сигнал улавливается антенной считывателя, информация расшифровывается и передается в компьютер для обработки.

Настоящую защиту от копирования и подделки обеспечивают такие идентификаторы, в чипах которых реализована криптографическая защита. Это бесконтактные смарт-карты, работающие на частоте **13,56 МГц**, наиболее распространенными из них являются карты **Mifare**<sup>®</sup>. В картах этих стандартов криптозащита организована на высоком уровне, и подделка таких карт практически невозможна.

Набор RFID считывателя RC522 13,56 МГц с SPI-интерфейсом. В комплекте к модулю идут 2 RFID-метки – в виде карты и брелока.





## Описание

В данном примере с помощью модуля RC522 узнаем UID метки, для этого необходимо записать скетч в Arduino (используя библиотеку MFRC522), собрать схему, изложенную ниже, и открыть Монитор порта. Когда вы поднесете метку к RFID, там выведется идентификационный номер. Запомните его и запишите в условие if (uidDec == полученный номер). Затем, если UID метки равен заданному, загорается зеленый светодиод, иначе красный.



#### Схема



Скетч **«ex22»** смотрите в архиве **«Исходники»**, скачать который можете на сайте <u>ampermarket.kz</u>.



# Пример 23 Сервопривод

Сервопривод — механизм с электромотором, который можно «попросить» повернуться в заданный угол и удерживать это положение. Однако это не совсем полное определение.

Если сказать полнее, сервопривод — это привод с управлением через отрицательную обратную связь, позволяющую точно управлять параметрами движения. Сервоприводом является любой тип механического привода, имеющий в составе датчик (положения, скорости, усилия и т. п.) и блок управления приводом, автоматически поддерживающий необходимые параметры на датчике и устройстве согласно заданному внешнему значению.



# Пример 23 Сервопривод

## Описание

В данном примере подключим простой и легкий сервопривод SG90, пусть он будет каждую секунду поворачиваться с шагом 15 градусов в одном направлении до 90 градусов, затем в обратном. Подключение сервопривода так же очень простое:

Коричневый провод — Земля (Ground, подключается к пину GND на плате Arduino)

Красный провод — Питание +5 V (подключается к пину 5V на плате Arduino)

Желтый провод — Сигнал управления (подключается к цифровому пину Arduino)

## Компоненты





# Пример 23 Сервопривод

### Схема



Скетч **«ex23**» смотрите в архиве **«Исходники**», скачать который можете на сайте <u>ampermarket.kz</u>.

10

# Пример 24 Модуль реле

Реле — это электрически управляемый, механический переключатель. Внутри этого корпуса есть электромагнит, при попадании на него электрического тока, он срабатывает, в результате чего якорь притягивается к электромагниту и контактная группа замыкает или размыкает цепь питания нагрузки.





# Пример 24 Модуль реле

## Описание

В данном примере для демонстрации принципа работы реле будем использовать обычный светодиод. Стандартно подключим модуль реле к Arduino, а для управления светодиодом будем использовать контакт NO (нормально разомкнутый), это означает, что при подаче тока на реле, светодиод будет гореть. Попробуйте поменять контакт NO на NC, тогда при подаче тока на реле светодиод гореть не будет.


## Пример 24 Модуль реле

#### Схема



Скетч **«ex24»** смотрите в архиве **«Исходники»**, скачать который можете на сайте <u>ampermarket.kz</u>.



## Пример 25 Двигатель постоянного тока

Двигатель постоянного тока — электрическая машина постоянного тока, преобразующая электрическую энергию постоянного тока в механическую энергию.





## Пример 25 Двигатель постоянного тока

#### Описание

В данном примере научимся управлять небольшим DCмоторчиком, используя плату Arduino, выпрямительный диод и транзистор. Транзистор действует как переключатель, управляя мощностью двигателя. 3-й пин Arduino (**motorPin** в скетче ниже) используется для включения и выключения транзистора. Когда скетч загрузится, откройте **Монитор порта**. Там Вы сможете ввести число **от 0 до 255** для управления скоростью вращения моторчика.



## Пример 25 Двигатель постоянного тока

#### Схема



Скетч **«ex25**» смотрите в архиве **«Исходники**», скачать который можете на сайте <u>ampermarket.kz</u>.

111

Шаговый двигатель — это мотор, который управляется несколькими электромагнитными катушками. На центральном валу – роторе – расположены магниты. В зависимости от того, есть ток на катушках, которые находятся вокруг вала, или нет, создаются магнитные поля, которые притягивают или отталкивают магниты на роторе. В результате вал шагового двигателя вращается. Подобная конструкция позволяет реализовать очень точное управление углом поворота ротора шагового двигателя относительно катушек – статора.





Униполярные шаговые двигатели имеют пять или шесть контактов для подключения и четыре электромагнитные катушки в корпусе (если быть более точными, то две катушки, разделенные на четыре). Центральные контакты катушек соединены вместе и используются для подачи питания на двигатель. Эти шаговые моторы называются униполярными, потому что питание всегда подается на один из этих полюсов.



#### Описание

Модуль драйвера шагового двигателя ULN2003 подключается к Arduino контактами IN1 – IN4 к D8 – D11 соответственно. Для подачи питания на мотор, рекомендуется использовать внешний источник питания 5 В с силой тока 500 мА. Вал шагового двигателя в результате выполнения данного примера будет делать один полный оборот в одну сторону, затем в обратную.

#### Компоненты





#### Схема



Скетч **«ex26**» смотрите в архиве **«Исходники»**, скачать который можете на сайте <u>ampermarket.kz</u>.



Модуль драйвера на L298N является двуканальным драйвером двигателей постоянного тока и основан на двух Hмостовых драйверах. Через этот модуль вы можете управлять двумя двигателями постоянного тока, либо одним четырехпроводным двухфазным шаговым двигателем. При управлении двумя двигателями постоянного тока имеется возможность одновременно с направлением их вращения управлять и скоростью с помощью подачи ШИМ-сигнала на соответствующие выводы. Если управление скоростью не требуется, то выводы ENA и ENB можно замкнуть перемычкой на соответствующие выводы +5V.



#### Обратите внимание на клеммник на драйвере:



Если на контакты рядом с клеммной колодкой питания установлена перемычка, то весь модуль будет питаться только от клеммника питания двигателей +12V. В данном случае перемычка направит ток на стабилизатор, который понизит напряжение до значения 5 В для корректной работы микросхем. В таком режиме на модуль можно подавать не более 12 В. Если **перемычка НЕ установлена**, то необходимо подавать на модуль отдельно питание для двигателей на клеммник +12V (от 6 до 35 В) и отдельно питание для логики микросхем на клеммник +5V (5 В).

#### Описание

В этом примере с помощью драйвера на L298N будем управлять двумя DC-моторчиками. Установим скорость и направление вращения, поменяем их. В теле функции demoOne() будем включать двигатели и начинаем с ними работать при ШИМ-значении 100. Для демонстрации возможностей изменения скорости вращения, используем доступный ШИМ-диапазон в теле функции demoTwo(). Сигнал на пине меняется от 0 до 255 и вновь до нуля.





Скетч **«ex27»** смотрите в архиве **«Исходники»**, скачать который можете на сайте <u>ampermarket.kz</u>.

זר

## Пример 28 MP3-плеер DFPlayer Mini

**МРЗ-плеер DFPlayer Mini** позволяет воспроизводить аудиофайлы, записанные на карту памяти формата microSD. С помощью данного плеера возможны также приостановка и возобновление воспроизведения, выбор одного из 30-ти уровней громкости и одного из 6-ти режимов эквалайзера. Для работы с DFPlayer Mini подойдет любая карта microSD с файловой системой FAT16 или FAT32 и объемом до 32 Гб.



VCC – Питание «+»
GND – Питание «-»
RX – UART приём
TX – UART передача
SPK1 – Громкоговоритель «+»
SPK2 – Громкоговоритель «-»
BUSY – Индикатор состояния («О» – простой, «1» –
проигрывание)
DAC_R – Выход на наушник или усилитель (канал «R»)
DAC_L – Выход на наушник или усилитель (канал «L»)
IO1 – Вход управления: короткое нажатие – «назад»,
длинное – уменьшить громкость
102 – Вход управления: короткое нажатие – «вперёд»,
длинное – увеличить громкость
<b>АДКЕУ1</b> – Порт для подключения резистивной
клавиатуры, вход 1
<b>АДКЕУ2</b> – Порт для подключения резистивной
клавиатуры, вход 2
<b>USB+</b> – USB порт, вывод «+»
<b>USB-</b> – USB порт, вывод «-»



## Пример 28 MP3-плеер DFPlayer Mini

#### Описание

В этом примере с помощью DFPlayer Mini и пассивного зуммера (для лучшего звучания лучше использовать динамик) проиграем мелодию. Все звуковые файлы должны лежать в папке «mp3» на microSD в следующем формате: «0001.mp3», «0002.mp3», «0003.mp3» и т. д. Также, понадобится библиотека DFPlayer\_Mini\_Mp3, ее можно найти в каталоге примера в папке libraries.



## Пример 28 MP3-плеер DFPlayer Mini

#### Схема



Скетч **«ех28»** смотрите в архиве **«Исходники»**, скачать который можете на сайте <u>ampermarket.kz</u>.



## Пример 29 Датчик уровня воды

**Датчик уровня воды** предназначен для определения уровня воды в различных емкостях, где недоступен визуальный контроль, с целью предупреждения перенаполнения емкости водой через критическую отметку. Конструкции датчиков уровня воды могут быть различными – поплавковые, погруженные, врезные. Данный датчик воды – **погруженный**. Чем больше погружение датчика в воду, тем меньше сопротивление между двумя соседними проводами.



## Пример 29 Датчик уровня воды

#### Описание

В данном примере подключим датчик уровня воды к Arduino, выведем полученное значение в Монитор порта и, если значение будет превышать 100, для индикации включать светодиод.





## Пример 29 Датчик уровня воды

#### Схема



Скетч **«ex29**» смотрите в архиве **«Исходники**», скачать который можете на сайте <u>ampermarket.kz</u>.



## Пример 30 Serial Port

Физическое подключение платы Ардуино к компьютеру через USB кабель существует всегда. Среда Arduino IDE имеет монитор последовательного порта, позволяющий получать и посылать данные обмена с платой. Можно передать на компьютер любую информацию о состоянии программы и вывести ее на дисплей. UART в переводе это универсальный асинхронный приемопередатчик. Данные UART передаются последовательным кодом в следующем формате:

ПЕРЕДАЧА БАЙТА ДАННЫХ ПО ПОСЛЕДОВАТЕЛЬНОМУ ИНТЕРФЕЙСУ



В классе Serial данные могут передаваться в двух форматах:

- как бинарный код;
- как ASCII символы.



## Пример 30 Serial Port

#### Описание

В данном примере продемонстрируем работу функции передачи и приема данных через последовательный порт. На компьютере мы будем передавать данные на плату Arduino, которая исходя из этих данных будет контролировать состояние **RGB LED модуля**. После загрузки скетча в плату откройте Монитор порта (Ctrl + Shift + M). В соответствии с программой выберите скорость передачи данных **9600 бод**. Теперь, отправляя «**r**», модуль засветится красным, «**g**» – зеленым и «**b**» – голубым.

# <section-header>KORMOHEHTHIImage: Single stateImage: Single state</t

## Пример 30 Serial Port

#### Схема



Скетч **«ех30»** смотрите в архиве **«Исходники»**, скачать который можете на сайте <u>ampermarket.kz</u>.



## Пример 31 Симуляция парковки

В данном примере мы будем использовать уже рассмотренные ранее компоненты: зуммер в <u>примере 4</u>, ЖК-дисплей в <u>примере</u> <u>12</u> и ультразвуковой модуль измерения расстояния HC-SRO4 в <u>примере 17</u>.



## Пример 31 Симуляция парковки

#### Описание

В данном примере смоделируем процесс парковки автомобиля с помощью дальномера HC-SRO4. Когда расстояние между датчиком расстояния и препятствием **больше 5 см и меньше 15 см**, зуммер издает звук в средней частоте. Если расстояние **меньше 5 см**, зуммер будет издавать звук в высокой частоте. Если **больше 15 см** – в низкой частоте. Между тем, текущая дистанция до препятствия будет отображаться на ЖК-дисплее.



## Пример 31 Симуляция парковки

#### Схема



Скетч **«ex31»** смотрите в архиве **«Исходники»**, скачать который можете на сайте <u>ampermarket.kz</u>.

## Пример 32 Датчик температуры DS18B20

Датчик температуры в Arduino – один из самых распространенных видов сенсоров. Разработчику проектов с термометрами на Arduino доступно множество разных вариантов, отличающихся по принципу действия, точности, конструктивному исполнению. Например, в <u>примере 15</u> для определения температуры мы использовали термистор, а в <u>примере 16</u> – датчик DHT11.

Цифровой датчик DS18B2O является одним из наиболее популярных температурных датчиков, часто он используется в водонепроницаемом корпусе для измерения температуры воды или других жидкостей.





## Пример 32 Датчик температуры DS18B20

#### Описание

В данном примере подключим цифровой датчик температуры DS18B20 (в двух его вариациях: в корпусе TO-92 и герметичном водонепроницаемом корпусе) к Arduino и выведем значения температуры в Монитор порта. Также, нам пригодится библиотека OneWire.



## Пример 32 Датчик температуры DS18B20

#### Схема



Скетч **«ех32»** смотрите в архиве **«Исходники»**, скачать который можете на сайте <u>ampermarket.kz</u>.



## **Пример 33** Bluetooth-модуль HC-06

**Bluetooth** обеспечивает обмен информацией между различными устройствами на доступной радиочастоте для ближней связи. Bluetooth позволяет этим устройствам сообщаться, когда они находятся в радиусе **до 10 м** друг от друга (дальность сильно зависит от преград и помех).

## Bluetooth<sup>®</sup>

**Bluetooth-модуль HC-06** – простой и наиболее популярный способ беспроводного дистанционного управления вашим устройством с помощью Bluetooth. Со стороны управляемого устройства, такого как Arduino, этот модуль выглядит как обычный последовательный интерфейс. Поэтому вы можете отладить всё общение с устройством на компьютере, а потом просто подключить этот Bluetooth-модуль, и всё заработает как надо. С HC-06 вы можете, например, управлять устройством прямо со своего смартфона. Поставив на телефон или планшет одну из многочисленных программ (терминалов) для управления через Bluetooth, вы можете превратить его в настоящий продвинутый джойстик, и ваше устройство сможет принимать и выполнять все отправленные команды.



## **Пример 33** Bluetooth-модуль HC-06

#### Описание

В данном примере подключим Bluetooth-модуль HC-06 к Arduino и настроим дистанционное управление с телефона с помощью Bluetooth-терминала (ссылка на Google Play). После того, как установите терминал, включим Bluetooth на телефоне и ищем новые устройства. Находим в списке устройств «HC-06» и подключаемся к нему. Телефон спросит пин-код – необходимо ввести «1234» или «0000».

После загрузки пишем в терминале цифру «0» и отправляем. Светодиод L, который находится на плате Arduino рядом с 13-м цифровым выводом, должен погаснуть. Теперь отправим через терминал цифру «1», и светодиод L должен зажечься.

# Компоненты Пата Arduino 1 к USB-кабель

## **Пример 33** Bluetooth-модуль HC-06

#### Схема



Скетч **«ех33»** смотрите в архиве **«Исходники»**, скачать который можете на сайте <u>ampermarket.kz</u>.

## Пример 34

#### Плата расширения с дисплеем и кнопками

LCD Keypad Shield — одна из самых популярных плат расширения для Arduino. Она может существенно упростить работу с жидкокристаллическими экранами, позволяя разработчику сосредоточиться на других задачах. Подключение, как и в случае с другими шилдами, производится одним движением: необходимо просто состыковать шилд с платой из семейства Arduino.





### Пример 34 Плата расширения с дисплеем и кнопками

#### Описание

В данном примере поместим LCD Keypad Shield в плату Arduino. По нажатию встроенных на шилд кнопок (SELECT, LEFT, UP, DOWN, RIGHT) выведем название соответствующей кнопки в Монитор порта и на экран дисплея. С помощью подстроечного резистора настройте контрастность дисплея. Также, нам понадобится уже знакомая библиотека LiquidCrystal.

#### Компоненты



Скетч **«ех34»** смотрите в архиве **«Исходники»**, скачать который можете на сайте <u>ampermarket.kz</u>.

## Пример 35 Motor Shield на L293D

Двигатели постоянного тока потребляет очень большой ток. И для того, чтобы управлять ими требуются специальные силовые транзисторные ключи. Существуют специальные микросхемы, которые заточены под эту задачу, например, L293D. А для плат Arduino существует уже готовое решение – плата расширения для двигателей на L293D. Она позволяет подключить 4 DC-мотора (либо 2 шаговых двигателя) и 2 серводвигателя.



## Пример 35 Motor Shield на L293D

#### Описание

В этом примере с помощью мотор-шилда на L293D будем одновременно управлять 4 двигателями постоянного тока (меняя скорость и направление). Для подключения шилда достаточно вставить его в плату Arduino Uno. Для работы с данным шилдом понадобится библиотека **AFMotor** (находится к каталоге примера в папке **libraries**).



## Пример 35 Motor Shield на L293D

#### Схема

![](_page_142_Figure_2.jpeg)

Скетч **«ех35»** смотрите в архиве **«Исходники»**, скачать который можете на сайте <u>ampermarket.kz</u>.

![](_page_142_Picture_4.jpeg)

## Пример 36 Ethernet Shield на W5100

**Ethernet шилд W5100** предназначен для коммуникации микроконтроллера с локальной сетью и реализует физический уровень интерфейса. Подключается к Arduino по интерфейсу SPI. Скорость обмена Internet – 10–100 Mb/s. Напряжение питания – 5В. Кроме собственно элементов для связи с сетью на плате расположен **разъём для microSD** карты памяти (для упрощения организации памяти для веб-сервера).

Шилд был разработан одним из первых и стал стандартом, поэтому библиотека для работы с ним включена в комплект поставки Arduino IDE и называется она просто – Ethernet library. Для подключения шилда к Arduino просто наденьте его на плату.

![](_page_143_Picture_3.jpeg)
### Пример 36 Ethernet Shield на W5100

#### Описание

В данном примере вставьте плату расширения в Arduino, затем откройте среду и загрузите пример WebServer (Файл > Примеры > Ethernet). После чего подключите Arduino к сети и перейдите по IP-адресу 192.168.1.177.

#### Компоненты



Скетч **«ех36»** смотрите в архиве **«Исходники»**, скачать который можете на сайте <u>ampermarket.kz</u>.



**GSM** (Global System for Mobile Communications) – глобальный стандарт цифровой мобильной сотовой связи с разделением каналов по времени (TDMA) и частоте (FDMA). GSM является самым распространенным стандартом связи в мире. На него приходится более 80% всего мирового рынка мобильной связи.



Плата расширения SIM900 GSM/GPRS функционирует на основе популярного четырехдиапазонного GSM/GPRS модуля SIM900. Как следует из названия, с этой платой расширения вы можете привнести в ваш проект на Arduino функционал связи по стандартам GSM/GPRS. Другими словами, данный шилд позволяет вам совершать голосовые звонки, отправлять SMS сообщения или наладить подключение к сети Интернет. Также на плате предусмотрена возможность режима отладки посредством Arduino или ПК, который включается переключателем DBG-Port.



146

#### Описание

В этом примере совершим звонок и отправим SMS-сообщение на указанный номер телефона, используя Arduino, GSM-шилд от Keyestudio и SIM-карту. В этом примере не понадобится чтолибо соединять, необходимо лишь вставить SIM-карту в шилд, а шилд – в нашу плату Arduino. Также, полезной окажется библиотека GPRSk\_Shield\_Arduino (находится в папке libraries данного примера).

#### Компоненты



#### Схема

### Установите перемычки следующим образом (RXD к D6, TXD к D7):



Скетчи **«ex37\_1»** и **«ex37\_2»** смотрите в архиве **«Исходники»**, скачать который можете на сайте <u>ampermarket.kz</u>.



GPS (Global Positioning System — система глобального позиционирования) — спутниковая система навигации, обеспечивающая измерение расстояния, времени и определяющая местоположение во всемирной системе координат WGS 84. Позволяет почти при любой погоде определять местоположение в любом месте Земли (исключая приполярные области) и околоземного космического пространства.

GPS-модули, в том числе и Ublox NEO-6M, позволяют вашему автономному устройству отслеживать свои координаты и параметры перемещения. Такая функциональность важна для всевозможных летательных аппаратов, трекеров, умных ошейников и рюкзаков.



#### Описание І

В данном примере с помощью программы U-Center определим местоположение, координаты, текущее время по спутнику, скорость объекта и другую информацию, используя GPS модуль Ublox NEO-6M. Программу для Windows можете найти в каталоге примера, либо зайдите на официальный сайт программы. Также, для TTL-преобразователя необходимо установить драйвера CH340G.

#### Компоненты





#### Схема





Теперь откройте программу U-Center. Здесь будет необходимо выбрать задействованный **порт**.



Далее в меню View выберите Text Console, либо просто нажмите клавишу **F8**. При верном подключении должна появляться информация такого вида:

Text Console			
22:51:38	\$GPG5A.A.3.08.21.14.27.324.00.3.55.1.85*03		
22:51:38	\$GPGSV.4.1.14.01.14.31608.23.264.33.10.70.08911.24.309.*7B		
22:51:38	\$GPGSV.4.2.14.14.47.219.36.18.38.303.20.41.105.21.07.155.34*70		
22:51:38	\$GPG5V.4.3.14.22.05.284.24.28.052.27.14.232.34.28.00.339.*7C		
22:51:38	\$GPGSV.4.4.14.31.01.180.32.64.219.37*7B		
22:51:38	\$GPGLL.5109.61826.N.07128.37129.E.225138.00.A.A*62		
22:51:39	\$GPRMC.225139.00.A.5109.61823.N.07128.37124.E.0.164260319A*7E		
22:51:39	\$GPVTGTM.0.164.N.0.303.K.A*20		
22:51:39	\$GPGGA,225139.00,5109.61823,N,07128.37124,E,1,05,3.55,364.3,M,-33.3,M,.*	74	
22:51:39	\$GPG5A,A,3,08,21,14,27,32,,,,,4.00,3.55,1.85*03		
22:51:39	\$GPG5V,4,1,14,01,14,316,,08,23,264,32,10,70,089,,11,24,309,*7A		
22:51:39	\$GPG5V,4,2,14,14,47,219,36,18,38,303,20,41,105,21,07,155,34*70		
22:51:39	\$GPG5V,4,3,14,22,05,284,,24,28,052,,27,14,232,34,28,00,339,*7C		
22:51:39	\$GPG5V,4,4,14,31,01,180,,32,64,219,37*7B		
22:51:39	\$GPGLL,5109.61823,N,07128.37124,E,225139.00,A,A*6B		
22:51:40	\$GPRMC,225140.00,A,5109.61816,N,07128.37119,E,0.144,,260319,,,A*7A		
22:51:40	\$GPVTG,,T,,M,0.144,N,0.267,K,A*21		
22:51:40	\$GPGGA,225140.00,5109.61816,N,07128.37119,E,1,05,3.55,364.2,M,-33.3,M,,*	73	
22:51:40	\$GPGSA,A,3,08,21,14,27,32,,,,,,,4.00,3.55,1.85*03		
22:51:40	\$GPG5V,4,1,14,01,14,316,,08,23,264,32,10,70,089,,11,24,309,*7A		
22:51:40	\$GPGSV,4,2,14,14,47,219,36,18,38,303,,20,41,105,,21,07,155,34*70		
22:51:40	\$GPG5V,4,3,14,22,05,284,,24,28,052,,27,14,232,34,28,00,339,*7C		
22:51:40	\$GPGSV,4,4,14,31,01,180,,32,64,219,37*7B		
22:51:40	\$GPGLL,5109.61816,N,07128.37119,E,225140.00,A,A*6D		
22:51:41	\$GPRMC,225141.00,A,5109.61808,N,07128.37111,E,0.180,,260319,,,A*74		
		-	
a 🗙 🗣			

152

Крайне желательно GPS модуль поместить ближе к окну, если находитесь в помещении, для лучшего сигнала. Спустя некоторое время **(5 – 30 минут)** появится полная информация, увидеть которую можно на диаграммах.





#### Описание II

Для подключения GPS модуля будем использовать библиотеку TinyGPS++. С помощью этого модуля можно получить следующие данные: широта, долгота, высота над уровнем моря, время, дата и количество используемых спутников. **RX** не соединяем, т.к. нет необходимости отправлять на него данные от Arduino, в этом примере Arduino просто получает данные от модуля.

#### Компоненты





#### Схема



Скетч **«ех38»** смотрите в архиве **«Исходники»**, скачать который можете на сайте <u>ampermarket.kz</u>.



# Пример 39

### Модуль считывания отпечатков пальцев

Оптические датчики отпечатков пальцев обычно используются в системах безопасности. Эти сенсоры включают в себя DSP чип, который обрабатывает изображение, производит необходимые расчеты для обнаружения соответствия между записанными и текущими данными.

Модуль поставляется с софтом для Windows, что значительно облегчает его тестирование. С использованием родного софта м о ж н о да же от образить фотографию вашего отпечатка на мониторе! Однако есть отдельная библиотека для Arduino, с использованием которой можно настроить датчик быстро и просто.



### Пример 39 Модуль считывания отпечатков пальцев

#### Описание

В данном примере нам понадобится удобная библиотека Adafruit Fingerprint Sensor Library (находится в папке libraries). С помощью нее мы сначала запишем два отпечатка и присвоим им ID, затем по отпечатку программа покажет указанный ID.

#### Компоненты



## Пример 39

### Модуль считывания отпечатков пальцев

#### Схема



Скетчи **«ex39\_enroll»** и **«ex39\_fingerprint»** смотрите в архиве **«Исходники»**, скачать который можете на сайте <u>ampermarket.kz</u>.



# Пример 39

### Модуль считывания отпечатков пальцев

Теперь зайдем в папку **examples** скачанной ранее библиотеки и запустим скетч **«enroll»**. Здесь необходимо сначала задать номер ID, затем приложить палец к датчику, чтобы он запомнил соответствие.

© COM4	_ <b>_</b> ×		
	Отправить		
Timage converted			
Remove finger			
ID 35			
Place same finger again			
Image taken			
Image converted			
Creating model for #35			
Prints matched!			
ID 35			
Stored!			
Waiting for valid finger to enroll as #35			
Image taken			
Image converted			
Remove finger	E		
ID 35			
Place same finger again			
	-		
Автопрокрутка Показать отметки времени   NL (Новая строка) 9600 бод	Очистить вывод		

### После этого запустите «fingerprint» и проверьте соответствие отпечатков.

∞ COM4				
	Отправить			
Addfruit finger detect test				
round ingerprint sensor: Sensor contains 3 templates				
Waiting for valid finger				
Found ID #27 with confidence of 84				
Found ID #35 with confidence of 77				
Found ID #13 with confidence of 108				
Found ID #35 with confidence of 97				
Found ID #27 with confidence of 104				
round 1D #13 with confidence of 108				
Автопрокрутка Показать отметки времени       NL (Новая строка)     9600 б	од 🔻 Очистить вывод			

В предыдущих примерах мы уже использовали простой шаговый двигатель **28ВҮЈ-48** для Arduino. В данном примере подключим более мощный шаговый двигатель **NEMA17HS4401**, ток на обмотку 1.7А. Как правило, используется в 3D принтерах, экструдерах и небольших ЧПУ станках. Для подключения данного двигателя будем использовать драйвер ТВ6600 4А.





ТВ6600 – это простой в использовании профессиональный драйвер шагового двигателя, который может управлять двухфазным шаговым двигателем. Он совместим с Arduino и другими микроконтроллерами, которые могут выдавать цифровой импульсный сигнал 5 В. ТВ6600 с питанием 9 – 40 В постоянного напряжения предназначен для использования с двигателями типа NEMA42 – NEMA86 с максимальным током фазы до 4А. Пиковый ток в 4А достаточен для большинства шаговых двигателей. Драйвер поддерживает управление скоростью и направлением движения. Вы можете установить его микро шаг и выходной ток с помощью переключателя. Существует 7 видов микрошагов (1, 2/A, 2/B, 4, 8, 16, 32) и 8 видов контроля тока (от 0.5 до 4 А).



#### Описание

В данном примере подключим шаговый двигатель 17HS4401 к драйверу ТB6600. В результате вал двигателя будет совершать цикл в 6400 шагов в одну сторону, затем в обратную. В качестве источника питания для драйвера можно, например, использовать импульсный блок питания на 12 В.





#### Схема



Скетч **«ех40»** смотрите в архиве **«Исходники»**, скачать который можете на сайте <u>ampermarket.kz</u>.



# интернет-магазин Ampermarket.kz



https://ampermarket.kz +7 775 137 27 09 +7 701 552 09 59