

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«НОВОСИБИРСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Физический факультет

Кафедра физико-технической информатики

Тихонова Анна Андреевна

ДОКЛАД

Внешние устройства для управления.

4 курс, группа 16312

Новосибирск, 2019 г.

Оглавление

1. Введение	3
2. Кнопки	3
2.1. Типичные проблемы и ограничения	3
2.2. Схемы подключения кнопки	4
2.3. Программное взаимодействие микроконтроллера с кнопкой	5
3. Матрицы кнопок.....	6
4. Потенциометр.....	7
4.1. Типичные проблемы и ограничения	7
4.2. Схема подключения потенциометра	7
4.3 Программное взаимодействие микроконтроллера с потенциометром	8
5. Энкодер.....	8
5.1. Типичные проблемы и ограничения	9
5.2. Схема подключения энкодера	9
5.3. Программное взаимодействие микроконтроллера с энкодером.....	9
6. Датчики физических величин.....	10
6.1. Схема подключения	10
Список литературы	10

1. Введение

Предназначение Arduino состоит в быстрой и удобной разработке электронных устройств. Это универсальный контроллер, на базе которого можно сделать законченный девайс абсолютно произвольного назначения. Для достижения этих целей к Arduino подключаются различные внешние устройства, с которыми она может взаимодействовать. Для измерения физических величин, условий окружающей среды, реакции на изменение состояний и положений применяются датчики. Их различают по роду сигнала и по назначению. По роду сигнала датчики делят на аналоговые и цифровые.

Благодаря наличию АЦП Arduino может считывать показания любых аналоговых датчиков: термисторы, термопары, фоторезисторы, потенциометры, напряжение шунта и прочее из мира аналоговой электроники. Для считывания цифровых значений могут использоваться как цифровые, так и аналоговые входы микроконтроллера.

Далее рассмотрены наиболее популярные внешние устройства для управления.

2. Кнопки

Кнопка (или кнопочный переключатель) – самый простой и доступный из всех видов датчиков. Это достаточно простое устройство, замыкающее и размыкающее электрическую сеть.

Далее описываются простые тактовые кнопки с 4 ножками, представляющие собой переключатель с двумя парами контактов. На рис. 1 представлена такая кнопка. Между парой на одной стороне реализован выключатель. Вторая пара контактов дублирует первую.

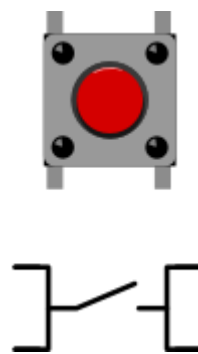


Рис. 1.

2.1. Типичные проблемы и ограничения

В процессе работы с кнопками мы можем столкнуться с очень неприятным явлением, называемым дребезгом кнопки. Это явление обуславливается дребезгом контактов внутри кнопочного переключателя. Металлические пластины соприкасаются друг с другом не мгновенно, поэтому на короткое

время в зоне контакта возникают скачки и провалы напряжения. Переходные процессы протекают очень быстро и исчезают за доли миллисекунд. Но, обрабатывая сигнал от кнопки на таком быстром устройстве, как Arduino, мы сталкиваемся с такими переходными эффектами и должны их учитывать при программировании.

Для устранения дребезга используют программные и аппаратные решения, описанные далее.

2.2. Схемы подключения кнопки

Типичная схема подключения кнопки к Arduino показана на рис. 2. Состояние кнопки считывается с цифрового входа pin 2. В указанной схеме используется стягивающий резистор (~10 КОм), без которого цифровой порт находился бы в неопределенном «подвешенном» состоянии. Когда кнопка отжата, pin 2 - LOW, так как он подключен к земле. Но если нажать на кнопку, то ток пойдет по пути наименьшего сопротивления и значение на pin 2 станет HIGH. Аналогично строится схема с подтягивающим резистором. В таком случае логика инвертируется.

Данную схему можно модифицировать для устранения дребезга на аппаратном уровне. Сглаживающий RC – фильтр сглаживает всплески сигналов

за счет добавления в схему конденсатора (рис. 3). Однако, сделать квадратную форму сигнала с помощью простой RC цепочки невозможно. Для “ограничения” сглаженных форм используется триггер Шмидта. На выходе мы получим или высокий, или низкий уровень сигнала.

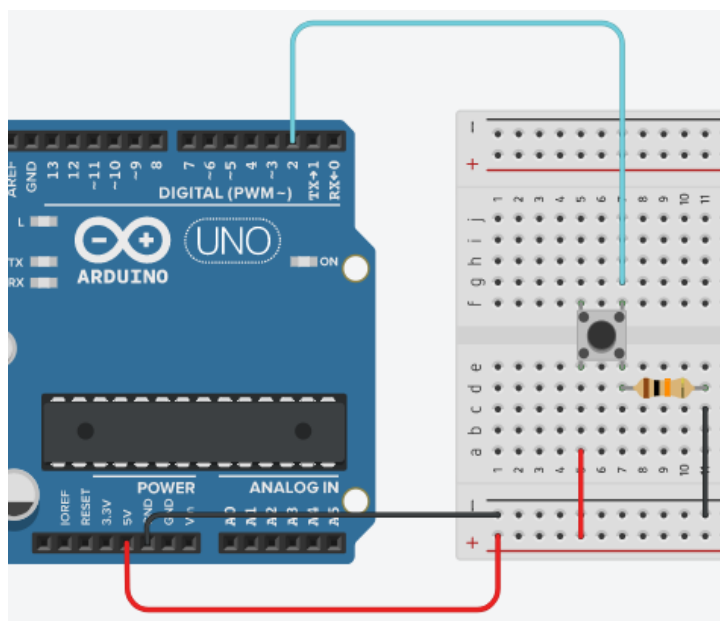


Рис. 2. Схема подключения кнопки к Arduino со стягивающим резистором.

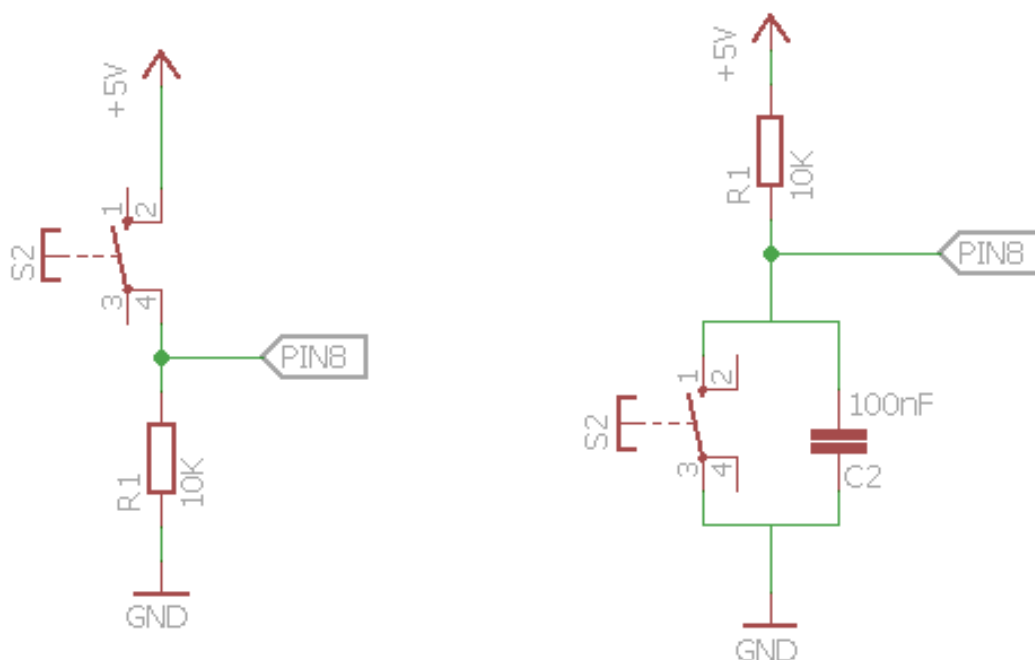


Рис. 3. Схема подключение фильтра для устранения дребезга.

2.3. Программное взаимодействие микроконтроллера с кнопкой

Используя схему подключения кнопки как на рис. 2, назначаем pin 2 на вход. Если считывать значение с кнопки непрерывно, в каждом новом рабочем цикле, то мы будем замечать все “всплески” и “падения” сигнала:

```
while(1){
    if(PIND & (1 << PD2)){
        // to do something
    }
}
```

Самым простым способом справиться с проблемой дребезга кнопки является выдерживание паузы. Мы просто останавливаемся и ждем, пока переходный процесс не завершится (10-50 миллисекунд – вполне нормальное значение паузы для большинства случаев). В данном примере используется задержку в программе, чтобы не реагировать на случайные всплески и определить реальную смену сигнала.

```
while(1){
    currentValue = PIND & (1 << PD2);
    if(currentValue != prevValue){
        _delay_ms(10);
```

```

currentValue = PIND & (1 << PD2);

}

prevValue = currentValue;

// to do something

}

```

3. Матрицы кнопок

Зачастую свободных выводов микроконтроллера не хватает для подключения необходимого количества кнопок. Для более рационального использования линий портов можно воспользоваться матричной схемой подключения. Кнопки группируются в ряды и столбцы, а полученная матрица последовательно опрашивается микроконтроллером, что позволяет резко снизить количество нужных выводов ценой усложнения алгоритма опроса (рис. 4).

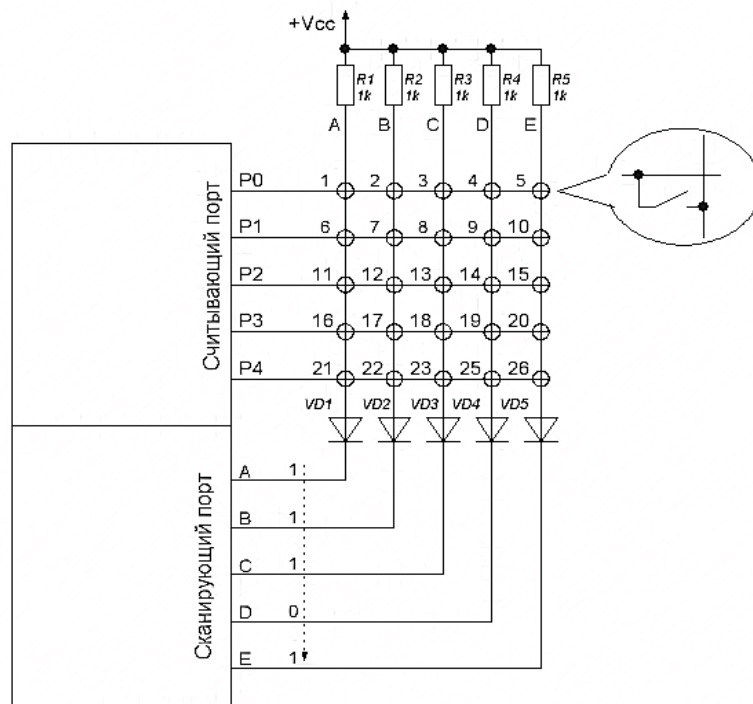


Рис. 4. Матрица кнопок.

Считывающий порт включается в режиме Pull-up входа. С него производится считывание состояния кнопок. Сканирующий порт работает в режиме выхода, он подключен к столбцам. Столбцы должны быть подтянуты резисторами к питанию.

В сканирующий порт выводится значение, состоящее из одного нуля и единицы на всех остальных выводах. Наличие нуля сразу же придавливает подтяжку и весь столбец ложится на землю. Теперь только при нажатии кнопок этого столбца на линиях считывающего порта может возникнуть низкий уровень.

Далее число в сканирующем порту сдвигается на один бит влево/вправо и сканируется второй столбец и т.д. В итоге, зная какой столбец мы сканируем, получив ноль на считывающем порту, мы поймем какая кнопка из матрицы нажата.

4. Потенциометр

Регулировка различных параметров, таких как громкость звука, мощность, напряжение и т.д., осуществляется при помощи переменных резисторов с регулируемым уровнем сопротивления. Примером такого устройства является потенциометр. Потенциометры делятся на два класса: цифровые и аналоговые. Основным элементом цифрового потенциометра является резистивная лестница, где на каждом шаге схемы имеются электронные переключатели. Аналоговый потенциометр может изменять свое значение непрерывно, но, как правило, в более узком диапазоне.

4.1. Типичные проблемы и ограничения

Для подключения потенциометра используется аналоговый вход, который прекрасно ловит помехи и наводки. Также на работу схемы влияет дрейзг контактов, который проявляется при кручении ручки потенциометра.

4.2. Схема подключения потенциометра

Подключение потенциометра к Arduino выполняется в соответствии со схемой, представленной на рис. 5. Крайние ножки переменного резистора подключаются к портам питания (5V и GND). Средний контакт подключен к аналоговому входу A0.

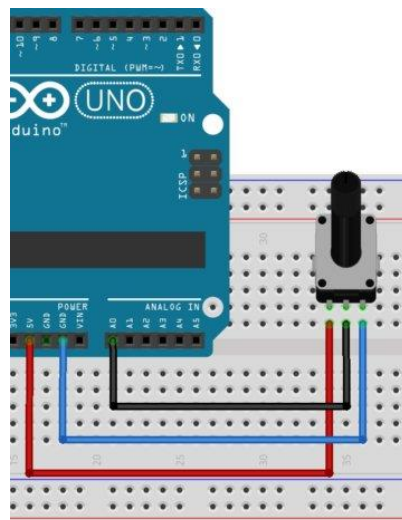


Рис. 5. Подключение потенциометра

Для борьбы с дребезгом также могут применяться сглаживающие фильтры (RC – фильтр).

4.3 Программное взаимодействие микроконтроллера с потенциометром

В плату Arduino встроен 10 битный АЦП, способный считывать напряжение и переводить его в цифровые показатели со значением от 0 до 1023. При повороте указателя до конечного значения в одном из двух возможных направлений, напряжение на пине равно 0, и, следовательно, напряжение, которое будет генерироваться составляет 0 В. При повороте вала до конца в противоположном направлении на пин поступает напряжение величиной 5В, а значит числовое значение будет составлять 1023. Эти значения считываются в бесконечном цикле (с задержкой для удобства).

Для борьбы с дребезгом на программном уровне считываемые значения в состоянии покоя могут быть усреднены.

5. Энкодер

Энкодер или датчик угла поворота – это электромеханическое устройство, предназначенное для преобразования углового положения вала или оси в электрические сигналы. Существует два основных типа энкодеров - инкрементные и абсолютные.

Инкрементный энкодер при вращении формирует импульсы, число которых пропорционально углу поворота (рис. 6). Подсчет числа этих импульсов дает величину угла поворота вала энкодера относительно его начального положения. Этот тип энкодеров не формирует выходные импульсы, когда его вал находится в покое. У него имеется 3 вывода, из которых 2 сигнальных (на рис. 6 сигналы а и b) и 1 общий.



Рис. 6. Возможные варианты сигналов инкрементного энкодера.

Абсолютный энкодер для каждой позиции своего вала выдает уникальный код, угол вращения всегда известен. Абсолютный энкодер формирует сигнал в любом состоянии, не теряет информацию о своем положении при потере питания и не требует возврата в начальную позицию.

5.1. Типичные проблемы и ограничения

Внутри энкодера имеются контакты, которые при вращении то замыкаются, то размыкаются. Этот процесс естественно сопровождается дребезгом.

5.2. Схема подключения энкодера

Схема подключения энкодера ничем не отличается от подключения обычных кнопок с подтягивающим резистором (рис. 7). Сигнальные выводы энкодера подключаются к любому порту ввода/вывода микроконтроллера. Общий вывод - GND. Для защиты от дребезга контактов добавляется пара керамических конденсаторов номиналом в несколько нФ.

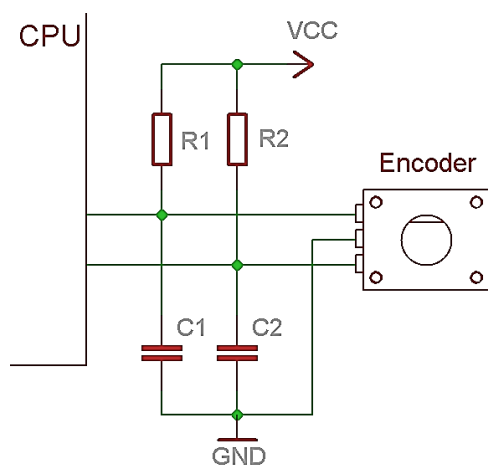


Рис. 7. Схема подключения энкодера к микроконтроллеру.

5.3. Программное взаимодействие микроконтроллера с энкодером

Выводы микроконтроллера в программе конфигурируются как входы. Если а и б идут на одни порт контроллера (например, на b=PB0 b=PB1), то при вращении энкодера у возникает меняющийся код: 11 = 3, 10 = 2, 00 = 0, 01 = 1, 11 = 3. В обработчике прерывания таймера запускается функция опроса энкодера, сравнивающая текущее состояние с новым и на основании этого делающая выводы о вращении. Частота опроса должна быть такой, чтобы не пропустить ни одного импульса.

Если используется аппаратное подавление дребезга, то можно применять прерывания. Настраиваем одно прерывание по одному из внешних сигналов так, чтобы сработка шла по восходящему фронту. А в обработчике прерывания проверяем уровень второго сигнала и инкрементируем/декрементируем счетный регистр в зависимости от уровня.

6. Датчики физических величин

Помимо перечисленных выше существует множество других датчиков, которые преобразуют физические величины, например, яркость света или температуру, в электрическую величину. Существует множество различных датчиков: датчик освещенности, датчик движения, датчик температуры, датчик давления, звуковой датчик и т.д.

6.1. Схема подключения

Так как большинство из аналоговых датчиков работают по принципу изменения сопротивления под действием внешних источников – температуры, света, излучений разного рода для них характерно подключение к микроконтроллеру по схеме как для потенциометра (резистивный делитель). Её суть сводится к тому, что между плюсом и минусом питания включаются два последовательно соединённых резистора, а сигнал снимается со средней точки. При этом один резистор имеет постоянное сопротивление, а второй – может его изменять (как исследуемый датчик). Подбрав нужные параметры на выходе средней точки можно получить диапазон напряжений от 0В до напряжения питания. Для цифровых датчиков соответственно выбирается цифровой вход микроконтроллера.

Список литературы

1. Инкрементальный энкодер < <http://easyelectronics.ru/avr-uchebnyj-kurs-inkrementalnyj-enkoder.html> >
2. Матричная клавиатура < <https://cxem.net/mc/book45.php> >
3. Самые популярные датчики для Arduino < <http://electrik.info/microcontroller/1447-samye-populyarnye-datchiki-dlya-arduino.html> >