

СОДЕРЖАНИЕ

Вступление	6
------------------	---

1 Что такое микроконтроллер Atmel ATtiny13	17
1.1. Микроконтроллер – мозг устройства	18
1.1.1. Обзор микроконтроллеров Atmel AVR	21
1.1.2. ATtiny13 – маленькая, но шустрая	22
1.1.3. Особенности применения	24
1.1.4. Заключение	24
1.2. Ассемблер Atmel AVR	25
1.2.1. Почему именно ассемблер?	25
1.2.2. Регистры	26
1.2.3. Стек и переменные	28
1.2.4. Проверка и пропуск	29
1.2.5. Выполнение логических операций	30
1.2.6. Побитовые операции	31
1.2.7. Арифметические операции	32
1.2.8. Пересылаем данные	32
1.2.9. Управление системой	33
1.2.10. Вызов процедур	33
1.2.11. Заключение	34
1.3. Пишем первую программу	34
1.3.1. Электронный «Hello world»	35
1.3.2. Основные элементы программы	36
1.3.3. Полный исходный код	39
1.3.4. Заключение	40

2 Компилируем, отлаживаем и заливаем	41
2.1. Средства разработки	42
2.1.1. Работа в AVR Studio	42
2.1.2. Получаем hex-файл	48
2.1.3. Программирование микроконтроллера с помощью AVRdude	49
2.1.4. Заключение	51
2.2. Моделируем работу устройства	52
2.2.1. Системы моделирования	52
2.2.2. Начало работы с ISIS Proteus	53

2.2.3. Учимся рисовать схемы.....	55
2.2.4. Оживляем схему	57
2.2.5. Отладка	58
2.2.6. Заключение.....	61

3 Устройства для обнаружения различных событий..... 62

3.1. Разрыв и замыкание цепи.....	63
3.1.1. Простейший «антивор»	63
3.1.2. Передаем код одной кнопкой	71
3.1.3. Заключение.....	83
3.2. Реагируем на различные явления с помощью датчиков.....	83
3.2.1. Обнаружение света	84
3.2.2. Устройство для обнаружения перегрева.....	89
3.2.3. Датчик влажности	92
3.2.4. Обнаруживаем движение.....	96
3.2.5. Датчик удара.....	99
3.2.6. Датчики дыма и газа	103
3.2.7. Заключение.....	107

4 По мотивам кухонных таймеров... .. 108

4.1. Работаем со временем	109
4.1.1. Циклы для таймера	109
4.1.2. Таймер.....	110
4.1.3. Программируемый таймер	113
4.2. Заключение.....	121

5 Охранные системы на ATtiny13 122 |

5.1. Основы разработки безопасных устройств	123
5.1.1. Надежность	123
5.1.2. Безопасность	125
5.2. Прикладные задачи.....	127
5.2.1. Скрытый кодовый замок.....	127
5.2.2. Не влезай... ..	135
5.2.3. Закладка в автомобиле	138
5.2.4. Интеграция с ПК и Raspberry	143
5.3. Концепции других устройств.....	148

5.3.1. ATTiny vs Arduino	148
5.3.2. ESP8266 и Интернет вещей	150
5.3.3. Onion – темная луковица IoT.....	151
5.4. Заключение.....	151
Подведение итогов.....	152
Приложение.....	154
П.1. Микроконтроллеры семейства AtmelATTiny	154
П.2. Команды ассемблера AtmelATTiny.....	155
П.3. Где взять исходный код.....	161
П.4. Библиография	161

ВСТУПЛЕНИЕ

Развитие микроэлектроники привело к появлению достаточно мощных, но при этом недорогих микроконтроллеров, с помощью которых радиолюбители могут разрабатывать собственные устройства. На рынке имеется множество различной литературы, посвященной различным микроконтроллерам. Значительная часть этих книг носит справочный характер, то есть в них большое внимание уделяется общим технологиям, применяемым в том или ином микроконтроллере, рассматривается система команд языка Ассемблер для данного семейства микроконтроллеров. Несомненно, данный тип книг очень полезен, но в них, как правило, не делается упора на разработку практических устройств, а ведь это следующий шаг при практическом изучении микроконтроллеров. Поэтому, на мой взгляд, имеется определенная потребность в практическом описании устройств, их схемах, исходных кодах прошивок с комментариями, а также способах их применения. Конечно, часть этой информации можно найти в Интернете. При написании своей книги я использовал материалы некоторых наиболее интересных ресурсов. Однако данная информация в большинстве своем отрывистая или неполная, например прошивка устройства представлена в виде скомпилированного hex-файла, без исходных кодов, что не позволяет изучить этот код и подправить его под свои нужды. Кроме того, часто примеры нужных устройств приводятся не под ту номенклатуру деталей, и прежде всего микроконтроллер, которые хотелось бы использовать. В таком случае также могут возникнуть сложности с внесением исправлений в схему устройства и исходный код.

В своей книге я делаю попытку исправить приведенные выше недостатки практических описаний устройств. Рассмотрев базовые основы программирования микроконтроллеров, я перейду к рассмотрению практических устройств. Все примеры будут содержать список необходимых для данного устройства деталей, схему, исходный код прошивки с комментариями, а также блок-схему, на которой будет представлен алгоритм работы программы. Данная блок-схема, во-первых, позволит лучше понимать, как работает программа, а во-вторых, при необходимости читатель сможет самостоятельно реализовать данный алгоритм на другом языке программирования. Обо всем этом мы поговорим в моей книге.

Все приведенные устройства построены на базе микроконтроллера ATtiny 13 фирмы Atmel. Выбор именно этого микроконтроллера не случаен. Конечно, в линейке Atmel имеется много интересных и более мощных контроллеров для решения различных задач. Од-

нако для разработки тех устройств, о которых пойдет речь в книге, данная микросхема является наиболее подходящей по следующим причинам: невысокая стоимость (около \$1), наличие АЦП, требования к питанию от 2,7 до 5,5 В, объема энергонезависимой памяти в 1 Кб и оперативной в 64 байта вполне достаточно для решения большинства задач. Кроме того, маленький размер позволяет делать миниатюрные, портативные устройства.

Почему устройства безопасности? Будучи по роду своей деятельности специалистом по информационной безопасности, я немало внимания уделяю аспектам, связанным с физической безопасностью. Причем здесь речь идет не только о средствах, предназначенных для предотвращения хищений и другой криминальной активности, но и о пожарной безопасности, утечек газа, протечек воды и тому подобных угроз. Таким образом, я считаю, что устройства безопасности являются необходимыми, и поэтому решил написать эту книгу.

А зачем вообще что-то делать самому, ведь все можно купить? Сами радиолюбители и вообще люди, увлекающиеся различными хобби, вряд ли задаются подобным вопросом. А вот их близкие и знакомые довольно часто об этом спрашивают. Объясняют они это тем, что сейчас производители из Юго-Восточной Азии предлагают множество различных устройств по низким ценам. Казалось бы, зачем делать устройства самому, если их можно легко приобрести дешево через Интернет. Однако на практике я часто сталкиваюсь с ситуацией, когда готовые устройства либо не обладают нужным функционалом, либо имеют избыточные возможности, которые мешают их нормальному применению, или просто недостаточно качественно сделаны, что также мешает их долговременной эксплуатации. Таким образом, зачастую недорогое, специально разработанное под конкретные задачи устройство собственного изготовления может лучше решать поставленные задачи, чем покупное.

На этом, я думаю, вступление можно завершить. Думаю, мне удалось убедить читателя в полезности моей книги.

С чего начать

Новичку в области радиоэлектроники непросто разобраться в том, как работает даже самое простое устройство. Кроме того, при работе с микроконтроллерами необходимо наличие базовых знаний по программированию. Для того чтобы получить общее представление о работе электронных устройств, я рекомендовал бы прочесть книгу «Искусство схемотехники» П. Хоровица и У. Хилла [1]. В этой книге рассматриваются базовые аспекты электроники. О более специфич-

ной литературе, связанной с микроконтроллерами Atmel, мы будем говорить в соответствующих разделах. Ну а теперь поговорим о том, что нам потребуется для работы.

Необходимые инструменты

Радиолюбительство традиционно ассоциируется с паяльниками, припоем, печатными платами и прочим оборудованием. Все это нам, несомненно, потребуется при сборке готового устройства. Но для быстрого «прототипирования», то есть разработки прототипа макета будущего устройства, нам потребуются другие компоненты.

Прежде всего нам необходима бесплаечная макетная плата. Применение такой платы позволяет проверить, наладить и протестировать схему еще до того, как устройство будет собрано на готовой печатной плате. Это дает возможность избежать ошибок при конструировании, а также быстро внести изменения в разрабатываемую схему и тут же проверить результат. Понятно, что макетная плата, безусловно, экономит массу времени и является очень полезной в мастерской радиолюбителя. Самый важный плюс бесплаечной монтажной платы – это отсутствие процесса пайки при макетировании схемы. Данное обстоятельство значительно сокращает процесс макетирования и отладки устройств.

Бесплаечная макетная плата состоит из пластмассового основания, в котором имеется набор токопроводящих контактных разъемов (рис. В.1).

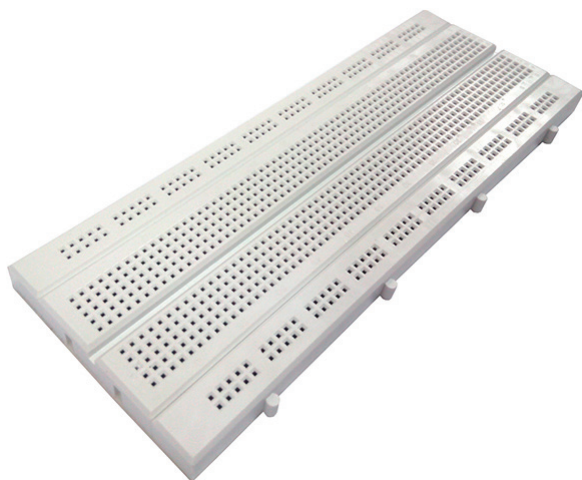


Рис. В.1. Бесплаечная макетная плата

Представленная на рисунке плата имеет горизонтальные ряды контактов, подключенных к общему проводнику. А вот по вертикали эти ряды контактов изолированы друг от друга. Таким образом, если нам нужен прототип устройства, мы размещаем детали, подключая их в разные ряды контактов. Для подключения микросхем имеется специальная бороздка посередине. Над ней крепятся микросхемы, при этом левый и правый ряды ножек изолированы друг от друга.

Еще одним полезным средством является набор монтажных перемычек, предназначенных для соединения удаленных друг от друга контактов. Конечно, можно использовать обычные провода, например можно «разобрать» кусок сетевого кабеля с витой парой. Однако такие проводки могут плохо держаться в макетной плате, в результате чего может возникать «дребезг контактов». Поэтому я бы рекомендовал использовать наборы готовых перемычек, например ВВJ-65.

Для работы любого электронного устройства необходимо электропитание. Для большинства описываемых в книге устройств для прототипирования достаточно питания 5 В. Случаи, когда используются батарейки с напряжением 3 В, будут рассматриваться особо.

А для того чтобы подать на плату напряжение 5 В, проще всего взять старую зарядку от мобильного телефона соответствующего номинала и срезать с нее штекер, затем зачистить и залудить контакты проводов. Теперь необходимо надежно закрепить провода в плате, например с помощью изолянты. Затем, не забывая про полярность, с помощью перемычек выводим питание на собранный макет устройства.

Думаю, с прототипированием устройств все понятно, теперь перейдем к рассмотрению элементной базы.

Какие потребуются детали

Перед началом разработки устройств нам потребуется некоторый набор деталей. Для большинства из них этот набор будет идентичен, отличаясь лишь двумя-тремя компонентами.

Прежде всего нам нужен микроконтроллер. В последующих главах мы подробно рассмотрим всю номенклатуру микроконтроллеров линейки ATTiny от Atmel. Но для наших задач нам потребуется микроконтроллер ATTINY13A-PU. Обращаю внимание на индекс PU, продаются также детали с индексом SSU, однако для разработки устройств, приведенных в моей книге, необходимо использовать микроконтроллеры с индексом PU.

Неотъемлемой частью практически любого устройства являются резисторы. Прежде всего рекомендую запастись резисторами номиналом в 10 кОм, они будут использоваться в каждом устройстве. Также для индикации сигналов на светодиоды предлагается использовать резисторы 300–400 Ом. В приведенных устройствах для работы с диодами я, как правило, использую 330 Ом. Также нам потребуются сами светодиоды. Кроме того, для подачи звуковых сигналов необходимо приобрести несколько динамиков. Здесь подойдут самые простые угольные.

Для получения информации о различных событиях окружающей среды нам потребуются соответствующие сенсоры, однако о конкретных моделях мы будем говорить в главах, посвященных данным устройствам (таких как датчик дыма, температуры и т. д.).

Ну и для разработки конечного устройства нам потребуется монтажная плата. В отличие от безопасной макетной платы, здесь нам обязательно потребуется пайка. Так как все описываемые в книге устройства достаточно простые, я предлагаю не заниматься травлением платы, а использовать готовую печатную плату. Такая плата содержит поле с изолированными друг от друга контактами, находящимися на расстоянии 2,54 мм. Для монтажа детали сначала впаиваются в контакты, а затем с помощью припоя соединяются друг с другом. Однако, если читателю больше нравится вариант с травлением платы, материалы по данной теме можно легко найти в Интернете.

Немного о программном обеспечении

Поговорив о необходимых деталях, теперь перейдем к не менее важному вопросу – какие программы нам потребуются. Прежде всего ответим на вопрос: зачем вообще нужно ПО при разработке устройств? У нас есть микроконтроллер, на который необходимо установить соответствующую прошивку. Для этого нам потребуется, во-первых, средство разработки и компиляции, во-вторых, средство эмуляции и, в-третьих, программатор, с помощью которого мы будем записывать прошивку на микроконтроллер.

В качестве основного средства разработки я предпочитаю использовать AVR Studio [2]. Это интегрированная среда разработки (IDE) для создания 8- и 32-битных AVR-проектов от компании Atmel, работающая в операционных системах семейства Windows. AVR Studio содержит компилятор ассемблера и C/C++ и симулятор, позволяющий отследить выполнение программы. Текущая версия 6 поддерживает все выпускаемые на сегодняшний день контроллеры AVR и средства разработки. AVR Studio содержит в себе менеджер

проектов, редактор исходного кода, инструменты виртуальной симуляции и внутрисхемной отладки, позволяет писать программы на ассемблере или на C/C++.

В своей книге я буду использовать AVR Studio в качестве среды разработки и компиляции. Для эмуляции работы мы будем использовать Proteus.

Пакет Proteus [3] предназначен для синтеза и моделирования непосредственно электронных схем. Работает также только под Windows.

Еще в состав пакета входит ARES – программа разработки печатных плат, однако в рамках данной книги мы ее рассматривать не будем. Вместе с программой устанавливается набор демонстрационных проектов для ознакомления.

AVR Studio и Proteus являются коммерческими решениями. Бесплатная ознакомительная версия характеризуется полной функциональностью, но не имеет возможности сохранения файлов.

И третий программный компонент – это программатор. Здесь, правда, нам потребуется не только софт, но и аппаратные средства. В качестве программного средства выступает небольшая бесплатная утилита AVRdude [4]. Здесь у многих бывалых специалистов могут возникнуть вопросы, почему я использую именно эту утилиту для прошивки микроконтроллеров, ведь существует множество других. Тут каждый волен использовать тот инструментарий, который ему больше нравится. Это же относится и к аппаратному программатору. Для разработки своих устройств я использовал ATMEL AVR DIP PROGRAMMER (рис. В.2).

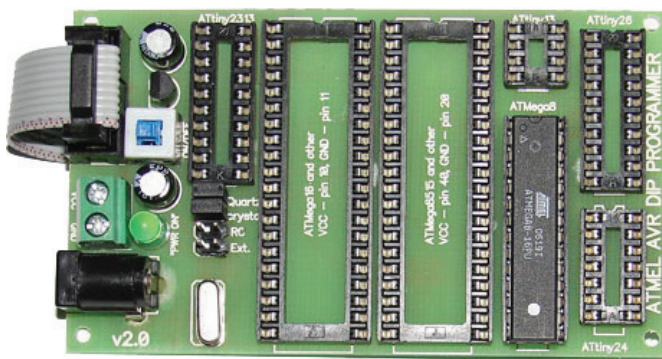


Рис. В.2. Программатор

Более подробно о том, как использовать описанные в этом разделе приложения и устройства, я расскажу далее в своей книге.

Некоторые особенности сборки готовых устройств

Итак, вы прошли весь путь от проектирования устройства до его сборки на реальной печатной плате. И теперь необходимо правильно продумать его форм-фактор. Все описанные в книге устройства являются малогабаритными, причем многие предполагают портативное использование и работу от батареек. В связи с этим требования к корпусу становятся жестче, ведь нам необходимо, с одной стороны, небольшое (не многим более спичечной коробки), но, с другой – надежно работающее устройство.

Конечно, готовые корпуса можно приобрести в магазинах радиотоваров. Однако я столкнулся с тем, что имеющиеся в продаже корпуса или слишком большие, или же недостаточно надежны и удобны для установки платы, батареек или вывода выключателей.

В связи с этим я хотел бы поделиться своим опытом по изготовлению корпусов для готовых устройств. Прежде всего необходимо определиться с тем, как устройство будет получать питание. Например, одно из моих устройств работало непосредственно от USB-порта. То есть его можно было включать в любой USB-порт, будь то компьютер или зарядное устройство от телефона, и оно начинало работать. Такое устройство вполне можно поместить в корпус от старой флешки или mp3-плеера. Также если детали на плате расположить соответствующим образом, то плата вполне помещается внутри гильзы от гладкоствольного охотничьего ружья, например 12-го калибра. Только гильза должна быть обязательно пластиковой, так как латунная может начать «коротить», если контакты попадут на ее корпус. Так что если у читателя есть знакомые охотники, то можно попробовать попросить несколько стрелянных гильз.

Если же устройство полностью портативное и предполагается часто его носить с собой, то здесь хорошим корпусом будет пластиковая сфера, которую используют для хранения игрушки в шоколадных яйцах, типа «Киндер-сюрприз». Небольшая овальная сфера помещается внутри яйца и используется для хранения игрушки. Так вот, в эту сферу можно поместить саму плату, а также батарейку типа CR2032, и при необходимости еще и динамик или другой элемент для вывода сигнала. Такой форм-фактор позволяет устройству быть более устойчивым при падениях и сотрясениях.

А кроме этого, корпуса можно делать из пластиковых электромонтажных коробов соответствующих размеров, просто отпиливая от короба кусочки нужного размера. Затем необходимо с одной стороны заклеить, например, кусочком пластика. После чего плату можно помещать в получившуюся коробочку.

Как видно, вариантов для изготовления корпуса устройства немало. Так что с помощью фантазии и напильника можно сделать не только надежное, но и оригинальное устройство.

Заключение

Итак, во вступлении я постарался рассказать о том, что мы собираемся делать, с помощью каких инструментов и как может выглядеть получившееся устройство. Думаю, начинающему радиолюбителю данной информации будет достаточно, для того чтобы начать искать необходимые инструменты, детали и программное обеспечение. На самом деле покупка деталей, необходимых для создания описанных в книге устройств, обойдется менее чем в тысячу рублей. Что касается инструментов, то начинающему вполне подойдет недорогой паяльник. Некоторые сложности могут вызвать программатор и плата для прошивки микроконтроллера, описанные в разделе 0.1.3, так как, например, плату мне пришлось заказывать через Интернет.

Однако в целом все это не так сложно, как может показаться. Далее мы поговорим о том, какие устройства мы будем делать.

Что мы будем делать

В этой главе мы обсудим, для чего нужны все те устройства, которые мы планируем собирать. Для специалистов по безопасности многое из того, что здесь написано, покажется очевидным, но для всех остальных эта информация может оказаться полезной.

Какие бывают устройства безопасности

Для обеспечения безопасности квартиры, дома или автомобиля нам необходимо прежде всего своевременно обнаруживать инциденты, которые могут привести к краже или разрушению охраняемого объекта. Только своевременное обнаружение позволяет предотвратить более тяжелые последствия. Угрозы могут быть природные, техногенные и антропогенные. Природные – это различные землетрясения, наводнения и другие разрушительные природные явления. Техногенные – это прежде всего пожары, прорывы трубопроводов, утечки газа и прочее. К антропогенным угрозам мы причислим ущерб и разрушения, причиняемые действиями человека. Здесь прежде всего стоит отметить криминальную активность, такую как кражи и незаконное проникновение на охраняемый объект. Также к этому виду угроз относятся ошибки и непреднамеренные действия обслуживающего персонала.

Определимся с используемой терминологией. Большинство датчиков состоит из двух элементов: сенсора, который реагирует на соответствующую активность, и основного блока. В последний входят: микроконтроллер, принимающий решение о срабатывании в случае инцидента, и блок реагирования. Это может быть светодиод, динамик или какой-то сервопривод, выполняющий физическое действие.

Разрыв и замыкание цепи

Наверное, наиболее распространенный вид устройств – это различные датчики размыкания цепи. Это и датчики открытия двери или окна (герконы), датчики размыкания, используемые в системах зажигания автомобилей, и другие простые, но весьма полезные устройства.

В основу всех этих устройств положен принцип контроля состояния электроцепи. Пока цепь замкнута, устройство находится в состоянии покоя, выходной сигнал отсутствует. Но как только цепь обрывается и сигнал пропадает, датчик подает сигнал тревоги.

Датчики освещения

Датчик света в задачах безопасности обычно используется для включения аварийного освещения. Основной принцип данных устройств заключается в контроле освещенности в помещении. Как только уровень света становится ниже определенного значения, датчик срабатывает, включая систему аварийного освещения.

Еще одна более оригинальная сфера применения датчиков освещения, которую любят показывать в фильмах, – это контроль периметра. На фоторезистор сенсора света направляется узкий луч света. Если между резистором и источником света появляется преграда, срабатывает сигнализация. Как видно, принцип работы таких датчиков аналогичен предыдущему.

Датчики температуры и влажности

Измерение температуры важно не только для комфортного пребывания человека в помещении, но и для корректной работы различного оборудования. В случае если в вашей серверной вышла из строя система кондиционирования, температура начнет быстро увеличиваться, что может привести к перегреву оборудования.

Датчик влажности обычно размещают на полу в тех помещениях, где возможна протечка воды.

Принцип работы обоих устройств схож: при превышении определенного порога срабатывает сигнализация. Хотя некоторые системы

обнаружения протечек используют наличие тока в качестве сигнала для срабатывания.

Датчики движения и удара

Принцип работы пассивных инфракрасных охранных датчиков движения (извещателей) основан на улавливании приемником инфракрасного излучения человека и выдаче сигнала, если человек двигается.

Датчик удара срабатывает при сотрясении сенсора. В обоих случаях признаком срабатывания датчика будет являться изменение значений (дельта) уровня сигнала, приходящих от сенсора сотрясений к датчику.

Датчики дыма и газа

Датчики дыма и газа срабатывают при появлении в воздухе соответствующих примесей. Для их работы необходимы соответствующие сенсоры. Принцип работы этих сенсоров довольно прост. Это прибор, структура которого включает в себя инфракрасный светодиод, а также приемник, который отвечает за управление его работой. Первый элемент предназначен для выработки световых импульсов, второй – их приема. В случае появления дыма на приемник поступает искаженный световой импульс, что приводит к срабатыванию датчика.

Обнаружение утечек газа производится аналогичным образом. Инфракрасные сенсоры определяют концентрацию газа, полосы поглощения которого находятся в диапазоне инфракрасного электромагнитного спектра. Для различных видов газов существуют разные сенсоры. Подробнее эта тема будет рассмотрена при описании соответствующего устройства.

Таймеры

Помимо различных датчиков, использующих сенсоры, для решения задач безопасности нам могут потребоваться различные таймеры. Например, если устройство должно сработать по прошествии определенного времени. В этом случае нам потребуется сделать таймер. Причем возможны варианты, когда таймер должен работать в различных режимах в зависимости от задачи. Например, в одном режиме нам необходимо задавать интервал времени в секундах, а в другом – в минутах. Для этого нужен программируемый таймер, которому сначала передается режим, в котором он будет работать, а только потом сам временной интервал для срабатывания.

Сигнализации и другие устройства

Сигнализации в наших устройствах – это блок реагирования. Здесь нам наиболее интересны системы звуковой сигнализации, в которых в случае инцидента сигнал подается на динамик.

Также полезным устройством безопасности является кодовый замок. Данные замки бывают как классические, с клавиатурой, так и однокнопочные, где для ввода кода используется последовательность нажатий.

«Сюрпризы» в автомобиле

Для защиты автомобилей традиционно используется множество различных устройств. Основная цель их – это помешать злоумышленнику угнать машину. Данные устройства представляют собой либо ту же сигнализацию, которая в случае инцидента подает сигнал, либо «молчаливый» блокиратор, который просто не дает завести двигатель, либо комбинацию этих устройств. В контексте данной книги мы поговорим о создании интерфейса управления данными охранными системами. Однако темы блока реагирования для автомобильных охранных систем (то есть того, что нужно сделать, чтобы машина не поехала) я касаться не буду. Дело в том, что времена «гаражных умельцев» уходят в прошлое. Электрика и электроника в современном автомобиле устроены достаточно сложно, и, для того чтобы вносить в них какие-то изменения, необходимы высокая квалификация и хорошее знание устройства данной конкретной модели автомобиля. Кроме того, при внесении изменений в гарантийную машину вы рискуете лишиться гарантии. И наконец, каждая самодельная система защиты должна быть уникальной, так что не стоит в «открытом эфире» обсуждать возможные варианты. Угонщики ведь тоже могут прочитать эту книгу.

Заключение

Итак, в этой главе мы поговорили о том, какие устройства безопасности бывают, какие принципы положены в основу их работы. Конечно, полный список таких устройств не ограничивается описанными здесь. Существуют еще всевозможные RFID-метки, устройства беспроводной связи и другие решения. Но в своей книге я рассматриваю наиболее полезные и применимые с точки зрения работы с ATTiny13 устройства.

Теперь мы перейдем к подробному рассмотрению «мозга» любого нашего устройства.

1 ЧТО ТАКОЕ МИКРОКОНТРОЛЛЕР ATMEL ATTINY13

2	Компилируем, отлаживаем и заливаем	46
3	Устройства для обнаружения различных событий	58
4	По мотивам кухонных таймеров...	78
5	Охранные системы на ATtiny13	

На сегодняшний день на рынке присутствует множество различных микроконтроллеров. Помимо Atmel, существуют также PIC, чьи контроллеры более подходят для тиражирования устройств, STM, ориентированны на более сложные, промышленные устройства и другие решения. Atmel из этого ряда выделяют дешёвизна, по сравнению с другими микроконтроллерами, простота в освоении. К тому же, на мой взгляд, именно младшие модели контроллеров Atmel из линейки ATtiny наиболее подходят для начинающих радиолюбителей, которые ещё не владеют всеми тонкостями работы с микроконтроллерами.

Но, прежде чем начать обсуждение продукции компании Atmel, я хотел бы поговорить о том, что такое микроконтроллер и зачем он нужен.

1.1. Микроконтроллер – мозг устройства

Микроконтроллер – это небольшая микросхема, на кристалле которой реализован функционал крошечного компьютера. То есть внутри одной микросхемы смонтированы процессор, оперативная и энергонезависимая память, периферийные устройства, аналогово-цифровой и цифроаналоговый преобразователи (АЦП и ЦАП). При этом данные элементы работают и взаимодействуют между собой и внешним миром с помощью специальной микропрограммы, которая хранится внутри микроконтроллера.

Основное назначение микроконтроллеров – это управление различными электронными устройствами. В контексте моей книги микроконтроллеры (МК) используются для управления устройствами безопасности. То есть именно он решает, какое значение входного параметра является поводом для выполнения тех или иных охранных действий. Например, именно МК решает, что надо включить сигнализацию, когда значение температуры, передаваемое термодатчиком, превысит некоторую пороговую величину.

Таким образом, можно без преувеличения сказать, что микроконтроллер является мозгом устройства (рис. 1.1).

Так выглядят современные микроконтроллеры.

В свою очередь, за правильность функционирования этого мозга отвечает программа, которая в него зашита. В обычных микросхемах функционал жестко зашивается при производстве и не может быть никак изменен. На микроконтроллер прошивку можно записывать многократно. Но не стоит считать МК «маленькими флешками». У МК гораздо меньше рабочих циклов записи, чем у USB-

накопителя. Не стоит более ста раз перепрошивать микроконтроллер, он может попросту не прошиться. В то же самое время данные на флешку можно записывать более 10 000 раз. Именно поэтому нам потребуется симулятор Proteus, с помощью которого можно будет протестировать работу прошивки и в определенной степени всего устройства, прежде чем начать прошивать живой микроконтроллер.

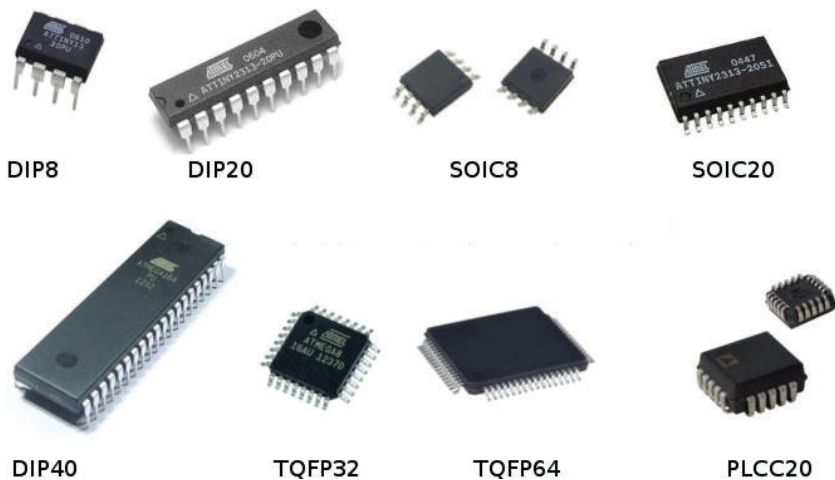


Рис. 1.1. Современные микроконтроллеры

Посмотрим, как логически устроен МК (рис. 1.2).

- Арифметико-логическое устройство (АЛУ) – предназначено для выполнения арифметических и логических операций, на самом деле в совокупности с регистрами общего назначения АЛУ выполняет функции процессора.
- Оперативно-запоминающее устройство (ОЗУ) – предназначено для временного хранения данных при работе микроконтроллера.
- Память программ – выполнена в виде перепрограммируемого постоянного запоминающего устройства и предназначена для записи микропрограммы управления микроконтроллером, так называемая прошивка.
- Память данных применяется в некоторых микроконтроллерах в качестве памяти для хранения всевозможных констант, табличных значений функций и т. д.

Микроконтроллер в своем составе может иметь и другие вспомогательные элементы.

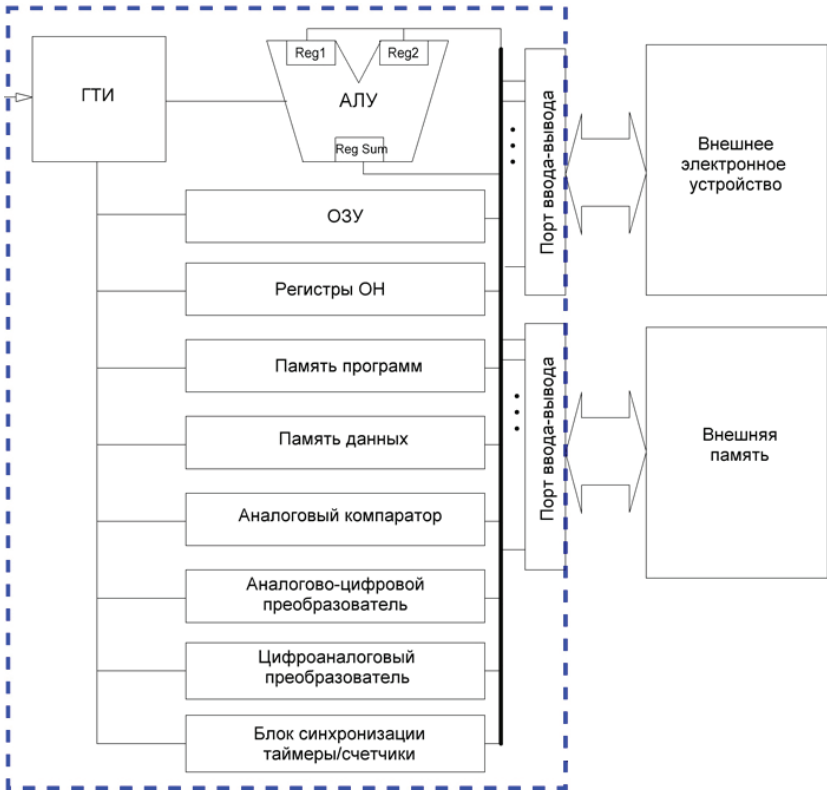


Рис. 1.2. Логическая схема устройства микроконтроллера

- Аналоговый компаратор – предназначен для сравнения двух аналоговых сигналов на его входах.
- Таймеры в микроконтроллерах применяются для осуществления различных задержек и установки различных интервалов времени в работе микроконтроллера.
- Аналого-цифровой преобразователь (АЦП) необходим для ввода аналогового сигнала в микроконтроллер, и его функция – перевести аналоговый сигнал в цифровой.
- Цифроаналоговый преобразователь (ЦАП) выполняет обратную функцию, то есть сигнал из цифрового вида преобразуется в аналоговый вид.

Программный код, который зашивается в микроконтроллер, отвечает за взаимодействие всех этих элементов в рамках поставленной программистом задачи.

Думаю, что с тем, что такое микроконтроллер, в целом понятно, теперь перейдем к рассмотрению МК от Atmel.

1.1.1. Обзор микроконтроллеров Atmel AVR

Компания Atmel производит несколько типов микроконтроллеров. На их сайте можно найти следующее описание:

32-разрядные микроконтроллеры AVR UC3

Самые эффективные 32-разрядные микроконтроллеры.

Предназначение: системы общего назначения.

Основные параметры:

- флеш-память 16–512 Кб;
- корпуса с 48–144 выводами;
- рабочая частота до 66 МГц;
- производительность 1,5 MIPS/МГц.

Микроконтроллеры AVR XMEGA

8-разрядное решение высокой производительности.

Предназначение: системы общего назначения.

Основные параметры:

- флеш-память 16–384 Кб;
- корпуса с 32–100 выводами;
- рабочая частота до 32 МГц;
- производительность 1,0 MIPS/МГц.

Микроконтроллер megaAVR

Больше функций и периферийных устройств.

Предназначение: системы общего назначения.

Основные параметры:

- флеш-память 4–256 Кб;
- корпуса с 28–100 выводами;
- рабочая частота до 20 МГц;
- производительность 1,0 MIPS/МГц.

8-разрядные микроконтроллеры tinyAVR

Компактность и мощность.

Предназначение: системы общего назначения.

Основные параметры:

- рабочее напряжение 0,7 В;
- флеш-память 0,5–8 Кб;
- корпуса с 6–32 выводами;
- рабочая частота до 20 МГц;
- производительность 1,0 MIPS/МГц.

Управление аккумулятором

Предназначение:

- управление литий-ионными аккумуляторами;
- измерение уровня заряда аккумулятора;
- балансировка элементов аккумулятора;

Основные параметры:

- рабочее напряжение 1,8–25 В;
- флеш-память 8–40 Кб;
- корпуса с 28–48 выводами;
- рабочая частота до 8 МГц;
- производительность 1,0 MIPS/МГц.

Микроконтроллеры AVR для автомобильной электроники

Предназначение: интеллектуальное управление и надежная конструкция, соответствующие жестким требованиям к автомобильной электронике.

Как видно, Atmel выпускает множество различных микроконтроллеров, предназначенных как для общего применения, так и для решения специализированных задач. Для решения задач, описываемых в этой книге, нам вполне подойдут 8-битные МК семейства *AttinyAVR*. Более мощные и, следовательно, более дорогие микроконтроллеры будут избыточны для этого.

1.1.2. ATtiny13 – маленькая, но шустрая

ATtiny – семейство AVR-микроконтроллеров, оптимизированных для приложений, требующих относительно большой производительности (до 1,0 MIPS и способных работать на частотах до 20,0 МГц), энергоэффективности и компактности.

Номенклатура микроконтроллеров ATtiny на сегодняшний день состоит из следующих устройств:

- ATtiny4, ATtiny5, ATtiny9, ATtiny10 – максимум 4 контакта ввода/вывода, рабочее напряжение 1,8–5,5 В, 32 байта SRAM, производительность до 12 MIPS (на частоте 12 МГц), Flash-память для хранения программ (1 Кбайт в ATtiny9/10 и 512 байт в ATtiny4/5), аналого-цифровой преобразователь (в ATtiny9/10).
- ATtiny13 – максимум 6 контактов ввода/вывода, рабочее напряжение 1,8–5,5 В, 64 байта SRAM, 64 байта EEPROM, производительность до 20 MIPS (на частоте 20 МГц), 1 Кбайт Flash-памяти для хранения программ, аналого-цифровой преобразователь (ADC).

Конец ознакомительного фрагмента.
Приобрести книгу можно
в интернет-магазине
«Электронный универс»
e-Univers.ru