

ТАЛЛИНСКИЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

Факультет информационных технологий

Институт вычислительной техники

IAG40LT

Андрей Кашин 124559

МАКЕТНАЯ ПЛАТА НА БАЗЕ AVR

Бакалаврская работа

Владимир Вийес

Ph.D.

Assistant Professor

Таллинн 2015

Autorideklaratsioon

Kinnitan, et olen koostanud antud lõputöö iseseisvalt ning seda ei ole kellegi teiste poolt varem kaitsmisele esitatud. Kõik töö koostamisel kasutatud teiste autorite tööd, olulised seisukohad, kirjandusallikatest ja mujalt pärinevad andmed on töös viidatud.

Andrei Kašin

10.12.2014

Annotatsioon

Tänapäeval ei kujuta me ette elu ilma elektoonika ja robotiteta. Neid kasutatakse kõikjal. Robot tolmuimeja, automaatne muruniiduk, robot käsi tööstustes jne. Robootika on väga huvitav ala mida õppida. Siiski on üsna raske algajal valida, kuidas sellega alustada. Robootikas on väga palju platvorme ja arhitektuure. Paljud platvormid pole taskukohased üksik kasutajatele, vaid seda saavad lubada ainult endale firmad. Hinnad kõiguvad paarist eurost kuni tuhandete eurodeni. Arhitektuuris on ka keerukus aste erinev. Vajadus on teha põhjalik uurimine, mis võib võtta kaua aega, et määratleda kust alustada.

Selleks, et algajatel oleks kergem alustada elektroonika ja robotikaga, selleks peaks olema spetsiaalne algaja arenduskomplekt, kus on kõik vajalikud komponendid arendusplaadi peal. See tähendab, et kasutaja on fokuseeritud tarkvaraarenduse protsessile, aga riistvara on juba varakult valmis.

Arenduskomplekt peab vastama järgmistele nõuetele. Arenduskomplekti hind peab olema piisavalt madal, et seda saaks endale lubada üksik isik. Isegi, kui platvorm on kergesti õpitav, paljud invidiidid ei võimalda endale arenduskomplekti, selle asemel üritavad leida alternatiivset võimalust.

Teine nõudmine on lihtsus. Liiga keeruline arhitektuur, nõuab liiga palju aega, et sellega alustada. Sellepärast peab olema arhitektuur nii kerge kui võimalik, sellejuures veel dokumenteeritud ja ühiskonnale teada. Hea platvormi näide on Arduino.

Arenduse käigus on risk, et mõni komponent läheb plaadil katki. See viib meid järgmise nõudmiseni: arenduskomplekti plaat peab olema kergesti parandatav. Teistesõnadega BOM peab sisaldama komponente, mis on kättesaadavad Euroopas.

Viimane nõudlus ütleb, et arenduskomplekt peab olema piisavalt pisike, et mahtuda tavalise robotti šassii peale. Paljude kasutajate šassiid ei ole suuremad kui 20x30 cm.

Lõputöö on kirjutatud vene keeles ning sisaldab teksti 43 leheküljel, 5 peatükki, 19 joonist, 4 tabelit.

Abstract

Today we cannot imagine our life without electronics and robotics. These are used everywhere. Robotic vacuum cleaners, automatic lawn mowers, robotic arms at factories, etc. Robotics is very interesting field to study. However, it is quite difficult for beginners to choose how to get started. There are lots of different platforms and architectures. Most of the platforms not affordable for private usage and applicable only for enterprises. Price could differ from few euros to hundreds. The complexity of the architecture is also different. Deep research will be required, which could take long time to define where to start.

In order to help beginners to get started with electronics and robotics there should be special starter kit, where all required components are combined in one development board. Which means that user will be focused on the software development process, the hardware will be prepared beforehand.

This development kit should meet following requirements. Starter kit has to be affordable for individuals, because even if the platform is very easy to study, most of individuals will not purchase the kit, but will try to find alternative one.

The second requirement is simplicity. Too complex architecture will require too much time to get started with. Therefore the architecture should be as simple as possible, has to be well documented and well known to community. Arduino is a good example of such platform.

During the development there is a risk to destroy a component on the board. That brings us to the next requirement – the development board should be easily maintainable. In other words the BOM has to contain components, which are available in Europe.

The last requirement says that development kit should be small enough to fit usual robotic chassis. Most of the consumer chassis are not bigger than 20x30 cm.

This thesis is written in Russian and is 43 pages long, including 5 chapters, 19 figures and 4 tables.

Словарь сокращений

ARM – семейство лицензируемых 32-битных и 64-битных микропроцессорных ядер разработки компании ARM Limited.

AVR – Семейство восьмибитных микроконтроллеров фирмы Atmel.

BOM – Bill of Material. Список используемых в производстве компонентов.

CISC – Complex Instruction Set Computing. Архитектура со сложным набором команд.

EEPROM – Electrically Erasable Programmable Read-Only Memory. Электрически стираемое перепрограммируемое ПЗУ.

GPS – Global Positioning System. Система глобального позиционирования

I2C – Inter-Integrated Circuit. Последовательная шина данных для связи интегральных схем.

IDE –Integrated Development Environment, интегрированная среда разработки.

JTAG – Joint Test Action Group. Интерфейс отладки и тестирования.

LSB – Less Significant Bit. Младший бит.

MCU – Micro Controller Unit. Микроконтроллер.

MSB – Most Significant Bit. Старший бит.

PCB – Printed Circuit Board. Печатная плата.

PIC – микроконтроллеры Гарвардской архитектуры, производимые компанией Microchip Technology Inc.

Pull-up – подтягивающий резистор.

RFID – Radio-frequency Identification. Радиочастотная идентификация.

RISC – Reduced Instruction Set Computing. Архитектура с сокращённым набором команд.

RTC – Real-time Clock. Часы реального времени.

SMD – Surface-mount Device. Компонент поверхностного монтажа.

SMT – Surface-mount Technology. Технология поверхностного монтажа компонентов.

SPI – Serial Peripheral Interface. Последовательный интерфейс.

THT – Through-hole Technology. Технология сквозного монтажа компонентов.

TQFP44 – Thin Quad Flat Package – 44 pin. Корпус микросхемы, 44 ножки.

TWI – Two Wire Interface. Аналог интерфейса I2C.

UART – Universal Asynchronous Receiver-Transmitter. Универсальный асинхронный приёмопередатчик.

USB – Universal Serial Bus. Универсальная последовательная шина.

АЛУ – арифметико-логическое устройство.

АЦП – аналого-цифровой преобразователь.

ДПТ – двигатель постоянного тока.

ОЗУ – оперативное запоминающее устройство.

ПЗУ – постоянное запоминающее устройство.

COM-порт – communicational port. Последовательный порт.

ЦПУ – центральное процессорное устройство.

ШД – шаговый двигатель.

ШИМ – широтно-импульсная модуляция.

Оглавление

1. Введение	11
2. Макетная плата.....	14
2.1. Анализ элементной базы	14
2.1.1. Выбор микроконтроллера	15
2.1.2. Микроконтроллер Atmega16.....	18
2.1.3. Интерфейсы микроконтроллера	19
2.1.4. Выбор микросхемы управления моторами	22
2.1.5. Дополнительное оборудование.....	23
2.1.6. Управление питанием	23
2.1.7. USB порт.....	24
2.1.8. Жидкокристаллический дисплей	25
2.1.9. Семисегментный индикатор	25
2.1.10. Светодиоды.....	26
2.1.11. Громкоговоритель	26
2.1.12. Внешний EEPROM.....	27
2.1.13. Датчик температуры.....	27
2.1.14. Кнопки.....	27
2.2. Схема устройства	28
2.2.1. U1. MCU – Главный контроллер	28
2.2.2. U4. Микросхема управления моторами – DRV8833	29
2.2.3. U2. Регулятор напряжения 5В – LM1117	30

2.2.4.	U3. Контроллер последовательного соединения – PL2303HX	30
2.2.5.	U6. Внешняя энергонезависимая память – EEPROM	30
2.2.6.	U5. Датчик температуры – TEMP SENSOR	31
2.2.7.	Кнопки – BUTTONS	31
2.3.	Дизайн платы	32
3.	Производство макетной платы	34
3.1.	Изготовление печатной платы	34
3.2.	Изготовление шаблона для нанесения паяльной пасты.	35
3.3.	Монтаж компонентов SMT.	35
3.4.	Процесс пайки	35
3.5.	Монтаж компонентов ТНТ	35
3.6.	Готовое изделие	36
3.7.	Тестирование	36
4.	Область применения	38
4.1.	Возможное применение на примере простого робота	38
4.2.	Дополнительные устройства для подключения	41
5.	Заключение	42
	Использованная литература	43

Список рисунков

Рисунок 1. Общая блок-схема макетной платы.	15
Рисунок 2. Блок-схема микроконтроллера Atmega16.....	21
Рисунок 3. Коннектор питания 2.1x5.5мм.	23
Рисунок 4. Коннектор типа "Терминал".	24
Рисунок 5. Коннектор USB типа "Mini B".	24
Рисунок 6. Жидкокристаллический дисплей HD44780.	25
Рисунок 7. Семисегментный индикатор.....	26
Рисунок 8. Светодиоды регулируемой индикации.	26
Рисунок 9. Громкоговоритель.....	27
Рисунок 10. Принципиальная схема подключения микроконтроллера.	28
Рисунок 11. Блок-схема устройства DRV8833.	29
Рисунок 12. Адрес микросхемы AT24C01 на шине I2C.	30
Рисунок 13. Адрес датчика температуры LM75AD на шине I2C.	31
Рисунок 14. Схема включения кнопок.	31
Рисунок 15. Схема печатной платы изделия. Размер 100x90 мм.	32
Рисунок 16. Трехмерная модель печатной платы.	35
Рисунок 17. Готовое изделие.	36
Рисунок 18. Блок-схема примитивного робота.....	39
Рисунок 19. Диаграмма состояний робота.	40

Список таблиц

Таблица 1. Сравнение контроллеров Atmega16 и PIC16F887.....	17
Таблица 2. Методы функционального теста.....	37
Таблица 3. Возможные варианты применения макетной платы.....	38
Таблица 4. Дополнительные комплектующие робота.....	38

1. Введение

Человек с древних времен мечтал создать управляемый механизм, который помогал бы решать повседневные проблемы, защищать от врагов или даже играть музыку. Описание таких созданий встречались еще в древнегреческой мифологии.

Более современным упоминанием робота является «Робот Леонардо» - человекоподобный механизм, чертежи которого были разработаны итальянским художником, ученым и писателем – Леонардо да Винчи. Согласно чертежу, этот механизм должен был воспроизводить некоторые движения человека, такие как движение руками и головой. Сам механизм представлялся как рыцарь в итало-германских доспехах. По этим чертежам исследователь Марио Таддей даже реконструировал робота, после чего представил свою книгу «Роботы Леонардо да Винчи» с описанием реконструкций.

Сам термин «робот», так популярный сегодня в мире, появился благодаря Карелу Чапеку, который впервые использовал этот термин в научно-фантастической пьесе «Россумские Универсальные Роботы» (1920). Действие пьесы разворачивается на фабрике по производству «искусственных людей» - роботов, один к одному похожих на живых людей. А само слово «робот» происходит от слова «robota», что на чешском языке означает тяжелый труд.

И правда, тяжелая и однообразная работа, изнуряющая человека, оказалась под силу механическим роботам. Роботы могут действовать на производстве круглосуточно, не отвлекаясь на перерывы или сон. С каждым днем роботы охватывают все большую часть областей производства, такие как конвейерные линии сборки или упаковки. В современном мире человечество применяет роботов и во многих других отраслях, начиная от военной промышленности и заканчивая медициной.

Сегодня уже никого не удивляет робот-пылесос или робот-газонокосилка. Уже давно ведутся разработки и тестирование роботов-беспилотников в воздухе, на земле, на воде и даже под водой. Воздушные беспилотники делают панорамные аэрофотографии, а роботы-сантехники проводят поиск неполадок и диагностику водопроводных труб. Роботы даже управляют автомобилями и вскоре могут появиться на обычных дорогах. Ну и, конечно, они осваивают космос.

С развитием электроники и робототехники, а также в связи с ростом популярности роботов растет и интерес к ним как к хобби. Сегодня можно приобрести недорогой робот-вертолет и сразу начинать осваивать навыки написания алгоритмов управления воздушным роботом, который также еще называют «дрон».

Однако, начинающий любитель роботостроения на первых этапах сталкивается с рядом проблем. Например, стоимость роботов все еще довольно высока и начинающему любителю стоимость такого робота оказывается не по карману, чтобы купить его для простых экспериментов. Конечно, есть комплекты с невысокой ценой, но низкая стоимость, как правило, означает, что начинающему любителю придется потратить много времени и сил на изготовление и сборку такого устройства, причем могут понадобиться инструменты, которые не всегда есть в наличии, например паяльная станция и химикаты для пайки.

Другим препятствием может быть недостаточное информирование о сопутствующих разработке алгоритма процессах, таких как программирование микроконтроллера, соединение с компьютером, а иногда и выбор подходящих сред разработки (IDE).

Именно в таком случае, когда любитель не знает с чего начать, важно предоставить доступную и простую макетную плату, а также пошаговое руководство с описанием всех процессов разработки. Например, какую среду разработки использовать, каким программатором лучше всего записывать данные в микроконтроллер, как считывать данные на персональном компьютере и т.д.

Зачастую для начинающего любителя бывает интересна только разработка программного обеспечения, а все остальные аспекты, связанные с аппаратной частью и механикой, пользователь хотел бы оставить в стороне. В таком случае использование макетной платы самое простое, быстрое и подходящее решение.

Небольшое исследование рынка электроники показало, что в данный момент не представлены простые макетные платы, подходящие для начинающих любителей робототехники. Есть либо дорогие, около двухсот евро за одну плату, либо недостаточно оснащенные – отсутствует блок управления электродвигателем.

В соответствии с вышеописанной целью необходимо решить следующие задачи:

- Создать макетную плату с использованием простой аппаратной части
- Сделать плату недорогой
- Использовать доступные компоненты, и как следствие – упростить ремонт в случае возникновения неполадок
- Изготовить макетную плату небольших размеров, что позволит легко монтировать ее на шасси разного типа

2. Макетная плата

Разработка макетной платы включает в себя несколько этапов, начиная от подбора подходящих компонентов и заканчивая отправкой чертежей на фабрику по изготовлению печатных плат. Рассмотрим ключевые этапы в хронологическом порядке.

2.1. Анализ элементной базы

Исходя из поставленных задач, для начала необходимо определиться с набором базовых элементов и компонентов макетной платы. Вот базовый список ключевых компонентов:

- Контроллер – для исполнения программы понадобится электронное вычислительное устройство. Для простоты и доступности этого компонента лучше использовать контроллер, так как в его конструкцию уже входят такие части как АЛУ, регистры, ОЗУ и ПЗУ. С учетом того, что макетная плата будет ориентирована на начинающего пользователя, лучше использовать простую архитектуру
- Микросхема управления моторами – большинство механических роботов перемещаются в пространстве, для этого нам понадобится устройство, позволяющее управлять небольшими моторами постоянного тока
- Порты ввода-вывода – у макетной платы обязательно должна быть возможность подключаться к персональному компьютеру или другим таким же устройствам для обмена данными и загрузки программ, также к портам ввода-вывода пользователи будут подключать разнообразное оборудование.
- Компоненты взаимодействия с пользователем – у пользователя должна быть возможность взаимодействовать с устройством во время исполнения программы. Как получать сигналы (свет, звук или текст), так и подавать команды устройству (например, кнопки).
- Дополнительное оборудование – хотя порты ввода-вывода позволяют подключать к себе большой ассортимент разного оборудования, минимальный набор датчиков может быть установлен на саму плату, чтобы пользователь мог попробовать получить информацию от дополнительного

устройства. Для этого можно разместить на макетной плате датчик температуры, имеющий интерфейс I2C [9].

Ниже представлена общая блок-схема макетной платы.



Рисунок 1. Общая блок-схема макетной платы.

2.1.1. Выбор микроконтроллера

Сегодня на рынке представлено огромное количество микроконтроллеров, разного уровня сложности и доступности. Начиная от экземпляров недорогой и популярной архитектуры ARM, предоставляемой сразу несколькими производителями, и заканчивая дорогими промышленными контроллерами высокой степени надежности.

Несмотря на большой ассортимент доступных микроконтроллеров, для использования многие из них не подходят. Так как макетная плата рассчитана в первую очередь на начинающего пользователя, то самое важное качество контроллера – это его простота, поэтому лучше всего использовать простой восьмибитный RISC контроллер. Такой контроллер будет намного легче изучить, что позволит пользователю видеть результаты труда почти сразу, начав практически с нуля. Другой аспект – это цена контроллера. Она не должна быть

слишком высокой, так как высокая цена делает макетную плату непривлекательной для использования.

Если исключить все дорогие, труднодоступные и сложные 32-битные контроллеры, то на выбор остается только два претендента: это контроллеры AVR фирмы Atmel и контроллеры PIC от фирмы Microchip.

Сравнивать контроллеры между собой практически невозможно, в связи с огромным количеством разнообразных моделей. Так например компания Microchip предоставляет на выбор 400 разных 8-битных микроконтроллеров. А фирма Atmel больше 100. Поэтому, чтобы упростить задачу, будем сравнивать наиболее популярные экземпляры небольшого размера. Для установки на макетную плату очень удобен корпус TQFP-44, такой корпус достаточно миниатюрен, и в то же время не слишком мал для замены в случае поломки. От производителя Microchip возьмем для сравнения популярную модель PIC16F887 [2], так как именно такой экземпляр установлен на оригинальной 44-х контактной макетной плате (Microchip 44-pin Demo Board). От компании Atmel возьмем контроллер Atmega16. Это один из наиболее популярных контроллеров среди семейства «megaAVR».

Таблица 1. Сравнение контроллеров Atmega16 и PIC16F887.

Параметр	Atmega16	PIC16A887
Flash (КБ)	16	14
Частота процессора до (МГц)	16	20
ЦПУ (bit)	8	8
Контакты ввода-вывода	32	36
SPI	1	1
I2C	1	1
UART	1	1
JTAG	Да	Нет
АЦП	8	14
Компаратор	1	2
ОЗУ (Б)	1024	368
EEPROM (Б)	512	256
Таймеры	3	3
ШИМ	4	2
Инструкции ЦПУ	131	35

Как видно из представленной выше таблицы, контроллеры Atmega16 и PIC16F887 имеют схожие параметры, поэтому среди них трудно выбрать победителя. С одной стороны у Atmega16 больше памяти, но PIC16F887 работает на большей частоте и имеет больше контактов ввода-вывода. Отдельного внимания требует количество инструкций двух архитектур: у Atmega их в четыре раза больше, чем у PIC. Небольшое число инструкций позволяет их быстро выучить, и через короткое время использовать весь потенциал контроллера. Большое число инструкций позволяет писать более гибкий исходный код, но требует большего времени на освоение. Конечно, это относится к разработке исходного кода на ассемблере. На языке Си эта разница, скорее всего, будет незаметна.

К явному преимуществу контроллера PIC16 можно отнести его невысокую стоимость на рынке электронных компонентов. Причем разница по цене очень заметна: по цене одного Atmega16 можно приобрести несколько PIC16F887.

Однако у контроллера Atmega16 тоже есть одно явное преимущество. А именно – интерфейс JTAG, позволяющий проводить отладку прямо внутри устройства. Инструмент JTAG экономит много времени, если возникает трудновоспроизводимая или трудноулавливаемая ошибка внутри исходного кода, он позволяет просто остановить процесс выполнения программы и затем выполнять ее пошагово, наблюдая за изменениями.

В итоге для использования на макетной плате я выбрал микроконтроллер Atmega16, хотя это не означает, что контроллер PIC16F877 хуже по своим характеристикам. Выбор в пользу Atmel был сделан субъективно, исходя из большего опыта использования AVR, чем PIC.

2.1.2. Микроконтроллер Atmega16

Для более детального изучения микроконтроллера Atmega16 обратимся к перечню технических данных [1], который также частично представлен в приложении.

Согласно перечню технических данных, микроконтроллер обладает рядом важных характеристик, которые следует для начала детально рассмотреть по отдельности.

- Тактовая частота процессора до 16 МГц. Такая высокая частота работы процессора достаточна, чтобы изделие быстро отзывалось на инструкции разработчика. Однако, чем выше частота, тем больше энергопотребление изделия. Поэтому было принято решение использовать частоту 12 МГц. Это не сильно повлияет на отзывчивость системы, но уменьшит энергопотребление.
- 32 регистра общего назначения. Все 32 регистра ядра AVR непосредственно подключены к арифметико-логическому устройству (АЛУ), что дает одной инструкции доступ к двум независимым регистрам за один такт. В результате этого выполнение кода является более эффективным, чем у обычных CISC микроконтроллеров.
- 4 порта ввода-вывода. PORT A служит как аналоговый вход для аналого-цифрового преобразователя, а также служит в качестве порта ввода-вывода, если АЦП не используется. PORT B – это 8-разрядный двунаправленный порт ввода-вывода со встроенными pull-up резисторами. Также этот порт используется для обслуживания специальных функций микроконтроллера.

PORT C – это 8-разрядный двунаправленный порт ввода-вывода со встроенными pull-up резисторами. Также этот порт обслуживает интерфейсы JTAG и I2C. И последний порт - PORT D –это также 8-разрядный двунаправленный порт ввода-вывода со встроенными pull-up резисторами. Также этот порт используется для обслуживания специальных функций микроконтроллера, таких как UART и прочее.

- ШИМ. Внутри микроконтроллера имеется встроенный ШИМ-регулятор, который позволит изменять скорость вращения моторов без необходимости менять напряжение их питания, за счет изменения скважности сигнала.
- Таймеры/Счетчики. Удобный инструмент для измерения количества тактов. Могут вызывать события и быть настроены на некоторое конечное число тактов.
- Часы реального времени. Таймер, работающий на частоте 32768 кГц. Может очень точно отсчитывать секунды, так как 2^{15} срабатываний при такой частоте равны точно одной секунде.
- Аналогово-цифровой преобразователь. Позволяет измерить уровень аналогового сигнала между нулем вольт и напряжением питания, а затем оцифровать его в соответствующее число от 0 до 1023 для дальнейшей обработки. В общем случае можно считать, что АЦП – это вольтметр диапазона 0 – 5В с величиной дискретизации 1024 единицы. В роботостроении АЦП используется в устройствах определения яркости диодов, громкости звука и даже детекторов препятствий. Поэтому его наличие на макетной плате просто необходимо.

2.1.3. Интерфейсы микроконтроллера

Контроллер не всегда работает один, зачастую на устройствах установлено дополнительное оборудование, с которым контроллеру необходимо обмениваться данными. Примером могут служить датчики, внешние запоминающие устройства или другие контроллеры. Также практически всегда контроллер обменивается данными с персональным компьютером или сервером, по крайней мере в момент записи программы.

- TWI – это низкоскоростная последовательная шина данных, позволяющая интегральным схемам обмениваться данными между собой. Также эта шина

известна под названием I2C. В силу того, что конструктивной разницы между этими двумя шинами нет, а имя I2C гораздо более популярно, то в дальнейшем будем использовать имя I2C. Изначально I2C была разработана корпорацией Philips в начале 1980-х годов, но до сих пор остается популярна в мире электроники. Очень удобна в разработке, так как большинство микросхем уже имеют встроенный I2C интерфейс. К тому же подключение дополнительных устройств к шине не требует никаких дополнительных цепей. Также I2C обладает ограничением – к одной шине может быть подключено только 127 устройств. Хотя в нашем случае это будет внешний EEPROM и датчик температуры, поэтому адресного пространства хватит с запасом.

- SPI – высокоскоростная последовательная шина данных, используемая в первую очередь для записи программы в память микроконтроллера. Кроме того, многие устройства обмениваются данными по этой шине. К примеру, карты памяти.
- JTAG – универсальное средство отладки. Позволяет запустить программу в режиме Debug, когда разработчик может остановить выполнение программы в определенном месте и проверить или поменять значения регистров, ячеек памяти, портов ввода-вывода и т.д.
- UART – низкоскоростной, но надежный протокол обмена данными между контроллерами или контроллером и компьютером. Обычно подключается к COM-порту компьютера. Позволяет выводить текстовые сообщения на экран компьютера, а также принимать команды, введенные пользователем.

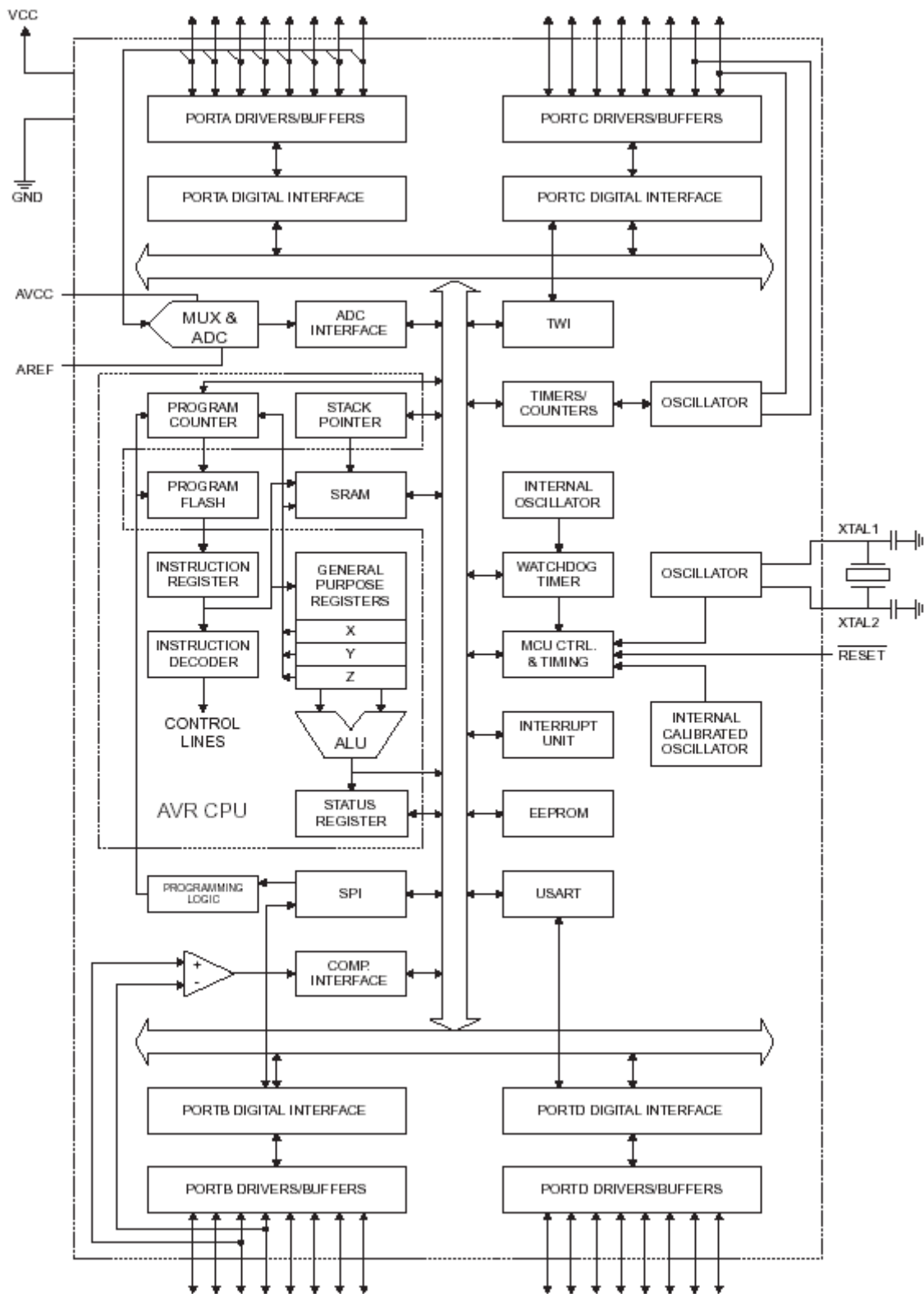


Рисунок 2. Блок-схема микроконтроллера Atmega16.

2.1.4. Выбор микросхемы управления моторами

Робот, как и любой другой механизм, не может быть неподвижным. Поэтому в макетной плате обязательно должно быть установлено устройство управления моторами, чтобы позволить роботу перемещаться в пространстве. Сегодня с уверенностью можно сказать, что самыми распространенными моторами в робототехнике являются двигатели постоянного тока и шаговые двигатели. ДПТ имеют небольшой размер и просты в управлении. ШД требуют более сложной системы управления, но зато имеют высокую точность поворота вала – обычно именуемую шагом.

В связи с ограниченным пространством на плате, необходимо выбрать микросхему, которая будет отвечать следующим требованиям: иметь небольшую цену и монтироваться на поверхности для экономии места, а также иметь как минимум два канала для управления ДПТ.

Ассортимент микросхем управления моторами довольно велик. Но большинство из них не предназначены для поверхностного монтажа. Другие же имеют слишком большую цену. Поэтому, отбросив все дорогие микросхемы, а также микросхемы сквозного монтажа, выбор остановился на микросхеме от производителя Texas Instruments - DRV8833 [3].

Рассмотрим основные характеристики этой микросхемы управления моторами.

- Микросхема управления моторами DRV8833 – это универсальный контроллер, который может управлять двумя двигателями постоянного тока или одним шаговым двигателем.
- Микросхема способна выдерживать номинальный ток нагрузки 1,5 Ампера, или 2 ампера кратковременно. С учетом того, что одна микросхема может управлять двумя моторами, то максимальный ток для двигателей постоянного тока составляет около 500мА на каждый двигатель. Учитывая, что в среднем ток потребления обычного двигателя постоянного тока около 50-100мА, сделаем предположение, что для управления простым роботом этой микросхемы будет хватать с запасом.

В случае, если понадобится управлять моторами, у которых суммарное потребление выше 1А, всегда можно использовать внешние платы контроля ДПТ и ШД.

2.1.5. Дополнительное оборудование

Кроме двух ключевых узлов – контроллера Atmega16 и микросхемы управления моторами DRV8833 – на плате также присутствует дополнительное оборудование, позволяющее отслеживать процессы, протекающие во время выполнения программы, соединяться с компьютером, отображать сообщения и получать команды от пользователя.

2.1.6. Управление питанием

Одной из главных частей макетной платы является блок питания. Как видно из технической документации, напряжение питания контроллера около 5В, в то время как напряжение питания ДПТ может варьироваться в пределах от 6 до 12 Вольт.

Также важно учитывать, что существует несколько способов подачи питания на плату. Это могут быть просто провода, а может быть блок питания с коннектором.

То есть, для универсальности сделаем два возможных варианта подключения.

Во-первых, через обычный коннектор 2,1x5,5мм, который позволяет подключать блоки питания от разных бытовых приборов, входное напряжение питания которых 6 – 9 Вольт. Например, роутеры или USB маршрутизаторы.



Рисунок 3. Коннектор питания 2.1x5.5мм.

Во-вторых, питание можно подать, просто подключив два провода, «плюс» и «минус», напрямую к терминалу входа питания. Такой способ будет удобен в случае подключения нескольких батареек или аккумуляторов.



Рисунок 4. Коннектор типа "Терминал".

2.1.7. USB порт

Для связи с компьютером удобнее всего использовать стандартные средства, которые есть на любом персональном компьютере. Поэтому на макетной плате должен быть установлен коннектор USB. Для экономии пространства будем использовать коннектор типа «Mini B», вариант для поверхностного монтажа. Коннектор такого типа не только занимает мало места на плате, но и обладает сравнительно низкой ценой и высокой доступностью, а также позволяет надежно закрепить себя, так как крепежные контакты имеют большую площадь соприкосновения с поверхностью платы. Это делает коннектор устойчивым к ударам небольшой силы, которые могут произойти во время движения робота.

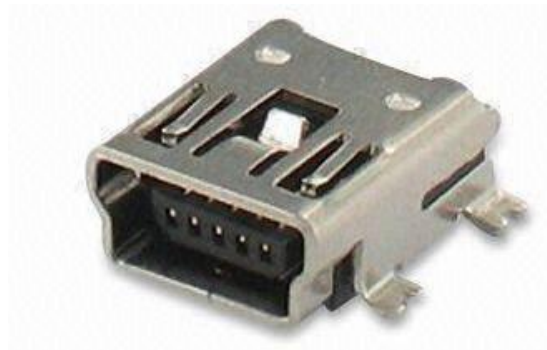


Рисунок 5. Коннектор USB типа "Mini B".

USB – это универсальный порт, и будет использоваться на плате несколькими способами. Во-первых, для связи микроконтроллера с компьютером, а точнее – с виртуальным COM-портом компьютера, который общается с микроконтроллером по интерфейсу UART.

Для соединения макетной платы с компьютером по интерфейсу UART требуется специализированная микросхема. Существует множество разных вариантов таких

микросхем. К ним относятся микросхемы FT232x, CP210x и другие, но для использования в проекте была выбрана микросхема PL2303, так как из всех вышеперечисленных микросхем является самой доступной по цене.

Во-вторых, через USB-порт можно подать питание на микроконтроллер. Однако, при таком подключении двигатели работать не смогут. Это сделано специально, чтобы не вывести из строя порт компьютера большим потреблением энергии моторами.

2.1.8. Жидкокристаллический дисплей

На плате установлен съемный жидкокристаллический дисплей HD44780, имеющий два ряда символов, по 16 знаков в каждом ряду. Такого дисплея достаточно для отображения коротких текстовых сообщений, однако, при использовании метода «бегущая строка», можно выводить на экран и довольно длинные сообщения.



Рисунок 6. Жидкокристаллический дисплей HD44780.

2.1.9. Семисегментный индикатор

Для отображения цифр удобно использовать семисегментные светодиодные индикаторы, поэтому на макетной плате установлен четырехзначный семисегментный индикатор с общим катодом. Этот индикатор имеет десятичные разделители, что позволяет отображать десятичные дроби. Также он обладает специальным разделителем – «двоеточием», позволяющим превратить индикатор в дисплей электронных часов.

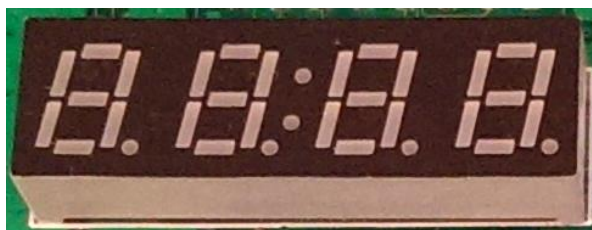


Рисунок 7. Семисегментный индикатор.

2.1.10. Светодиоды

Для примитивной индикации на макетной плате установлено 4 светодиода. Эти диоды могут включаться и выключаться в зависимости от программы.

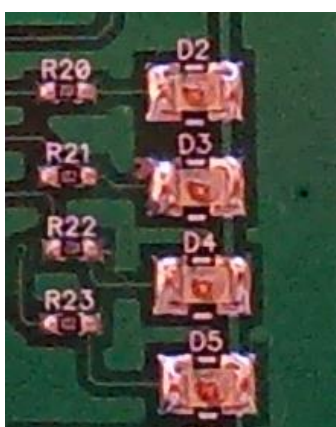


Рисунок 8. Светодиоды регулируемой индикации.

Также на макетной плате присутствуют и другие светодиоды – D1 и D6, но их назначение – оповещение пользователя о проблемах с аппаратным обеспечением.

Так светодиод D6 – это индикатор питания. Он загорается, если на главный контроллер подается напряжение питания.

Светодиод D1 загорается в случае проблем с микросхемой DRV8833. К возможным причинам относятся перегрузка и перегрев компонента.

2.1.11. Громкоговоритель

Иногда для оповещения пользователя о событиях удобно использовать звуковую сигнализацию вместо визуальной. Для этих целей на макетной плате был установлен громкоговоритель. Тип громкоговорителя такой же, как и на материнских платах обычных компьютеров.



Рисунок 9. Громкоговоритель.

2.1.12. Внешний EEPROM

Несмотря на то, что контроллер Atmega16 имеет встроенную энергонезависимую память EEPROM, на макетной плате все же установлена отдельная микросхема памяти этого типа. Это сделано для того, чтобы иметь внешнюю память, отделенную от главного контроллера, так как во время разработки программ центральный микроконтроллер может выйти из строя, а важные данные, сохраненные на во внешней памяти, останутся невредимыми. Также это позволит внешним устройствам третьей стороны записывать данные на макетную плату, используя интерфейс I2C, даже если центральный микроконтроллер не функционирует.

2.1.13. Датчик температуры

Для измерения температуры может быть использована микросхема LM75AD, установленная на макетную плату. Этот датчик удобен тем, что данные о температуре можно сразу получить по интерфейсу I2C, без необходимости создания аналогово-цифрового преобразования и калибровки.

2.1.14. Кнопки

Для двустороннего общения с контроллером необходимо не только получать информацию, но также и отправлять ее. Для отправки примитивных команд контроллеру во время выполнения программы могут служить обычные кнопки. На макетной плате для этих целей установлены две кнопки S2 и S3. По нажатию на эти кнопки контроллер может реагировать определенным образом, заданным в программе.

Кнопка S1 используется только для сброса микроконтроллера и перезапуска выполнения программы. Запрограммировать ее поведение невозможно.

2.2. Схема устройства

Принципиальная схема макетной платы представлена в приложении. Рассмотрим основные составляющие блоки и логические модули.

2.2.1. U1. MCU – Главный контроллер

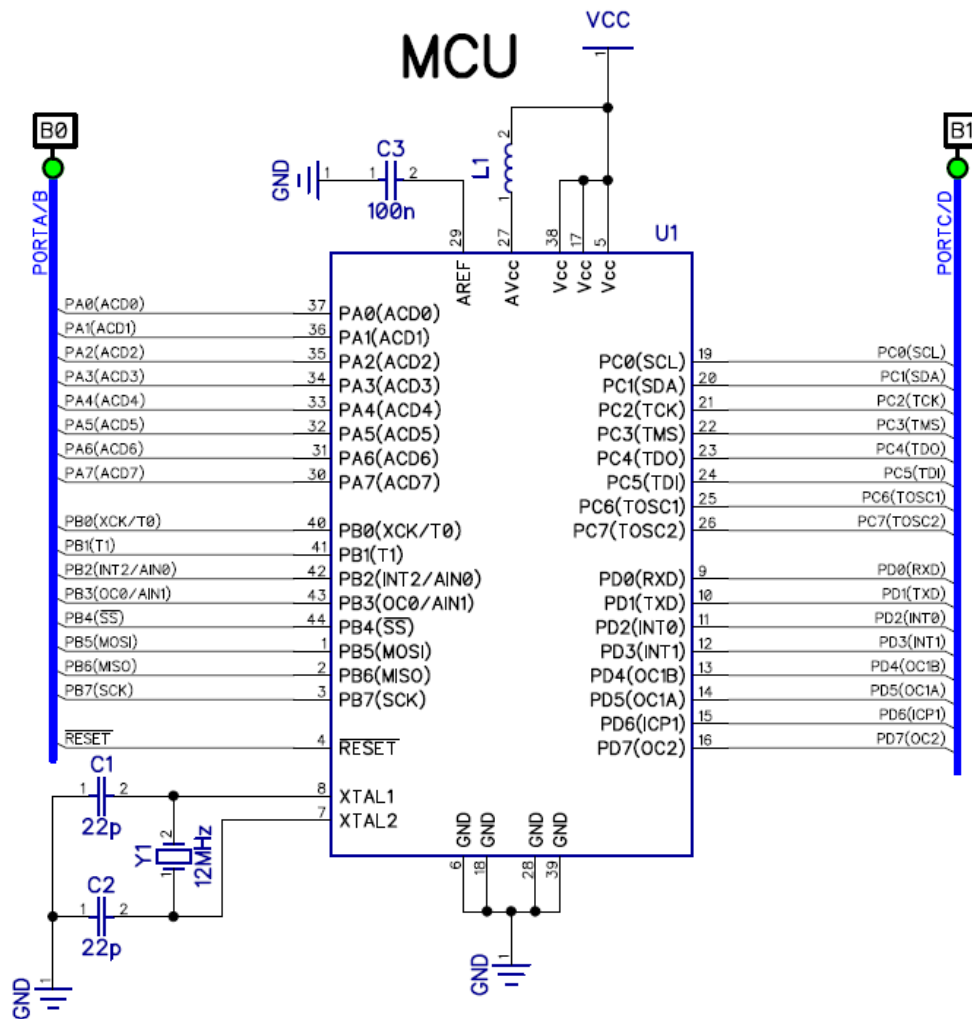


Рисунок 10. Принципиальная схема подключения микроконтроллера.

При подключении микроконтроллера необходимо учитывать некоторые ключевые моменты.

Питание микроконтроллеру подается подключением сигнала VCC, равным 5 вольтам постоянного стабилизированного напряжения. Следует также обратить внимание на 27-ю ножку контроллера, это сигнал AVcc – питание АЦП. Так как аналогово-цифровой преобразователь может реагировать даже на незначительные изменения сигналов, то очень важно подавать на него питание без помех, чтобы не

влиять на результаты замеров. Для этой цели между V_{CC} и AV_{CC} установлен дроссель L1.

Сигнал AREF является входом опорного напряжения АЦП. Он шунтирован конденсатором в 100нФ для уменьшения помех, согласно рекомендации в технической документации (стр. 198)

Также к контроллеру подключен внешний источник тактового сигнала – кварцевый генератор на 12МГц.

2.2.2. U4. Микросхема управления моторами – DRV8833

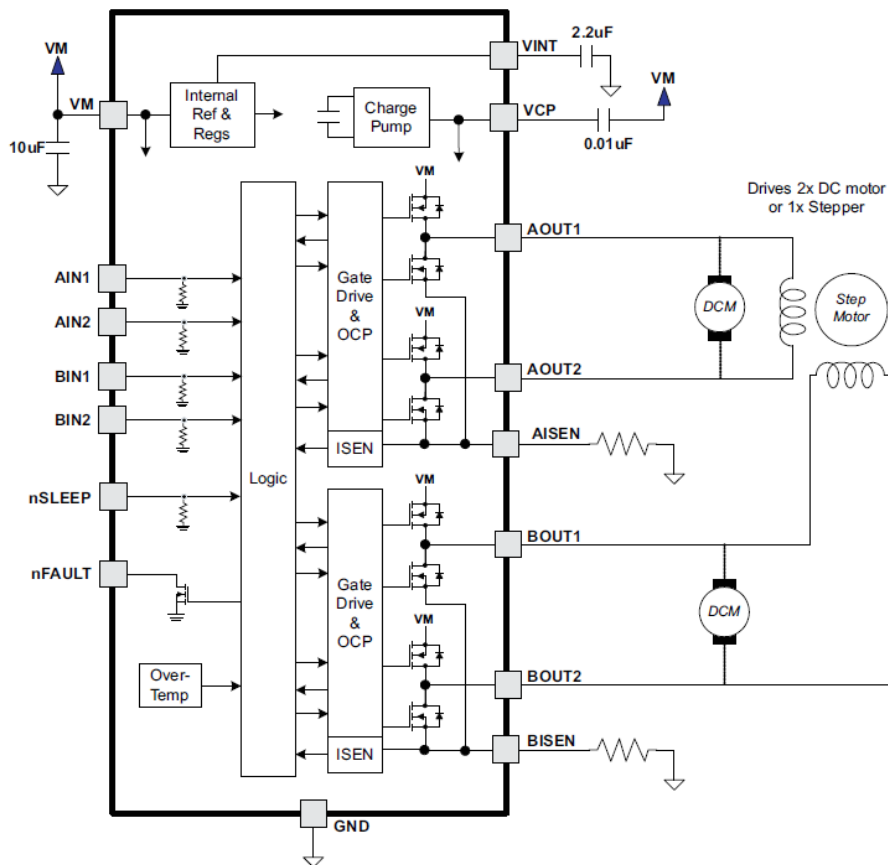


Рисунок 11. Блок-схема устройства DRV8833.

Контроллер имеет четыре входных сигнала для приема сигналов управления и четыре выхода для управления непосредственно двигателями. Вход nSLEEP позволяет переводить контроллер в режим ожидания, когда вся внутренняя логика выключена и потребление энергии сведено к минимуму. Выход nFAULT позволяет

подключить к себе светодиод индикации ошибки. В нашем случае диод D1 загорится, если произойдет превышение потребления тока или перегрев микросхемы.

2.2.3. U2. Регулятор напряжения 5В – LM1117

Для регулирования питанием макетной платы используется микросхема LM1117 [4], модификация для стабилизации напряжения 5 Вольт. Эта микросхема имеет встроенные ограничение по току и защиту от перегрева. Номинальный выходной ток микросхемы равен 800мА. На макетной плате эта микросхема отвечает только за питание контроллера, элементов индикации и других микросхем. Питание на сами моторы будет подаваться в обход регулятора LM1117.

2.2.4. U3. Контроллер последовательного соединения – PL2303HX

Микросхема PL2303HX [5] – это мост между COM-портом компьютера и интерфейсом UART макетной платы. К микросхеме PL2303 подключен внешний кварц на 12МГц. Выводы 1 и 5 подключены к интерфейсу UART, а выводы 15 и 16 – к коннектору USB. Выводы 13 и 14 могут быть подключены к интерфейсу I2C, а настройки микросхемы могут быть сохранены во внешней памяти EEPROM. Однако в первой версии прототипа платы эти выводы остались неподключенными, а микросхема использует настройки «по умолчанию».

2.2.5. U6. Внешняя энергонезависимая память – EEPROM

Кроме внутренней энергонезависимой памяти контроллера на макетной плате установлена дополнительная микросхема памяти EEPROM AT24C01 объемом 1024 бита [6]. Подключена она к последовательному интерфейсу I2C. Адрес последовательной шины этой микросхемы представлен ниже.

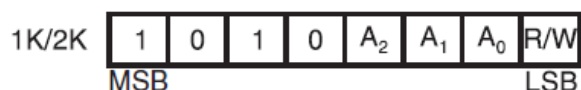


Рисунок 12. Адрес микросхемы AT24C01 на шине I2C.

Выводы микросхемы 1, 2 и 3 являются выводами настройки адреса – A0, A1 и A2 соответственно. На макетной плате все три вывода подключены к логическому нулю, что дает в результате адрес «1010000»

2.2.6. U5. Датчик температуры – TEMP SENSOR

На макетной плате используется датчик температуры LM75AD [7] со встроенным интерфейсом I2C. Как и все устройства на шине I2C у датчика должен быть уникальный адрес. Адрес этой микросхемы представлен ниже

1 = HIGH; 0 = LOW.

MSB				LSB		
1	0	0	1	A2	A1	A0

Рисунок 13. Адрес датчика температуры LM75AD на шине I2C.

Выводы A2, A1 и A0 – выводы настройки адреса – подключены как «001» соответственно, что в результате дает адрес «1001001»

2.2.7. Кнопки – BUTTONS

При подключении кнопок необходимо построить схему так, чтобы избежать такого явления как «дребезг контактов». Этот эффект возникает из-за того, что в момент нажатия кнопки ее контакты успевают несколько раз сомкнуться и разомкнуться, что даст множество ложных срабатываний. Для избегания такой проблемы на плате используются конденсаторы C14 и C15. Их задача сгладить изменение уровня напряжения на сигналах PC6 и PC7 в момент нажатия кнопки.

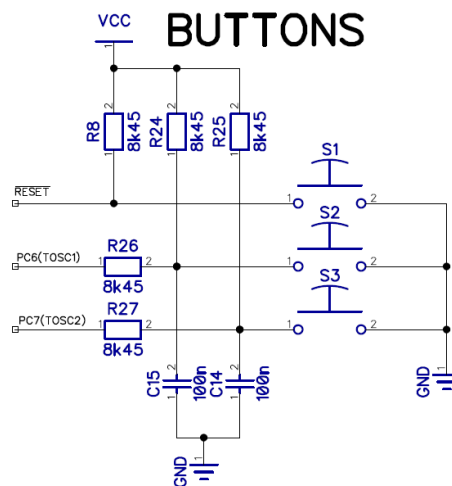


Рисунок 14. Схема включения кнопок.

2.3. Дизайн платы

Когда определены основные компоненты и составлена принципиальная схема устройства, можно переходить к следующему этапу – разработке печатной платы (PCB).

Разработка макетной платы велась в программе DipTrace [10], которая обладает довольно широким выбором моделей разных компонентов. Однако, не все модели оказались в наличии в основной базе. К счастью, программа DipTrace имеет специальный модуль Pattern Editor, в котором можно создать модель компонента. Характеристики корпуса компонента, такие как размер паяльных площадок и расстояние между ними, как правило, указаны в технической документации.

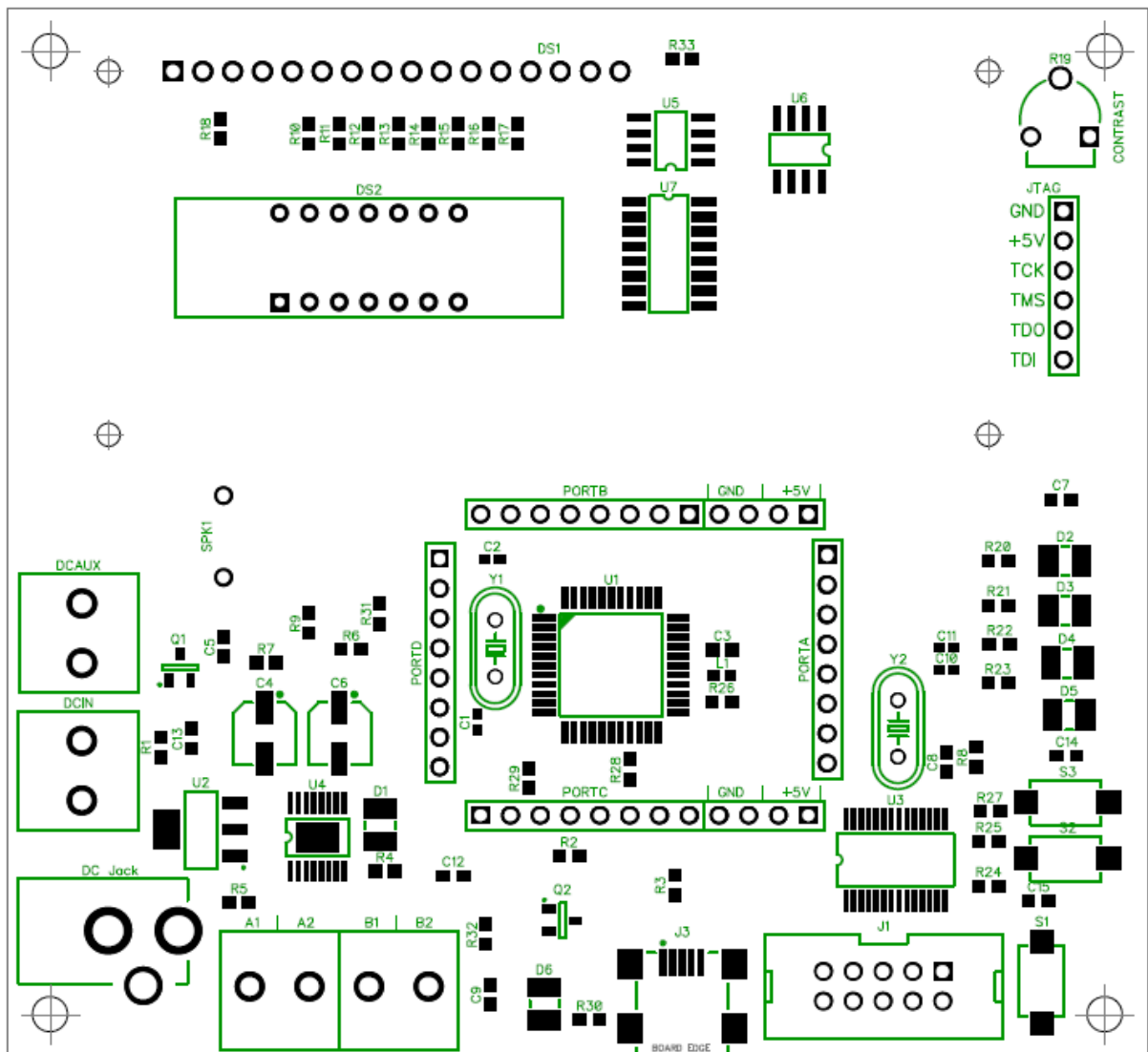


Рисунок 15. Схема печатной платы изделия. Размер 100x90 мм.

Зеленым цветом отмечены контуры компонентов и их имена. Черным цветом обозначаются паяльные площадки компонентов. Серые окружности – отверстия для крепежа.

3. Производство макетной платы

После завершения разработки изделия в электронном виде можно приступать непосредственно к изготовлению печатной платы и сборке компонентов.

Производство макетной платы проводилось частично в домашних условиях и многие важные этапы были пропущены. Поэтому далее я буду описывать общепринятый процесс производства электронных изделий, разбив его на ключевые этапы.

3.1. Изготовление печатной платы

Изготовить печатную плату можно несколькими способами. Самый быстрый и дешевый, это травление платы в домашних условиях. Конечно, такой способ больше подходит для простых прототипов, а изготовить таким способом двустороннюю печатную плату уже довольно сложно. Второй способ – это изготовление печатной платы при помощи фрезерного станка с программным управлением. Однако, такая печатная плата не защищена лаком и металл будет подвергаться коррозии. Поэтому такой способ также отставим. Третий и самый дорогой способ – это обратиться на специализированный завод по производству печатных плат. Изготовленная таким образом плата будет протестирована, защищена от коррозии и иметь привлекательный вид.

В нашем случае производство печатной платы было заказано у специализированного предприятия Itead Studio.

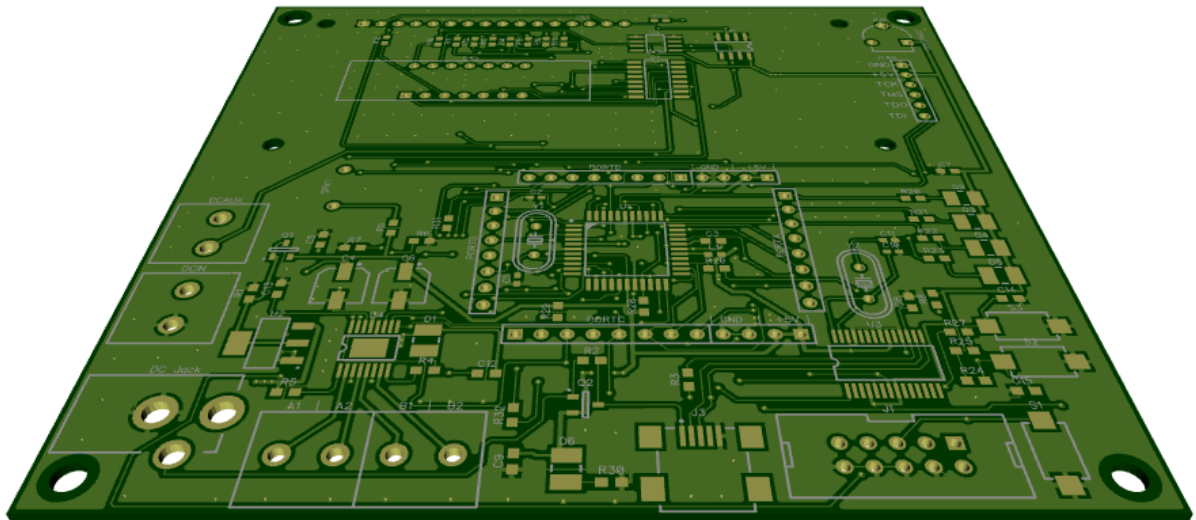


Рисунок 16. Трехмерная модель печатной платы.

3.2. Изготовление шаблона для нанесения паяльной пасты.

Когда печатная плата готова, перед установкой компонентов необходимо нанести на плату паяльную пасту. Для этого используется специальный шаблон, отверстия в котором совпадают с расположением и размерами паяльных площадок.

3.3. Монтаж компонентов SMT.

Компоненты поверхностного монтажа принято монтировать при помощи специальной машины «Pick And Place». Это высокоточное устройство, которое способно захватывать компоненты и размещать их в заданном месте на печатной плате. Компоненты сквозного монтажа на данном этапе не монтируются.

3.4. Процесс пайки

Когда компоненты установлены на свои места, то плата отправляется в специальную печь, где паяльная паста постепенно нагревается до температуры плавления и компоненты припаиваются в печатной плате. Затем идет процесс плавного охлаждения.

3.5. Монтаж компонентов ТНТ

Последним шагом идет установка компонентов сквозного монтажа (ТНТ).

3.6. Готовое изделие

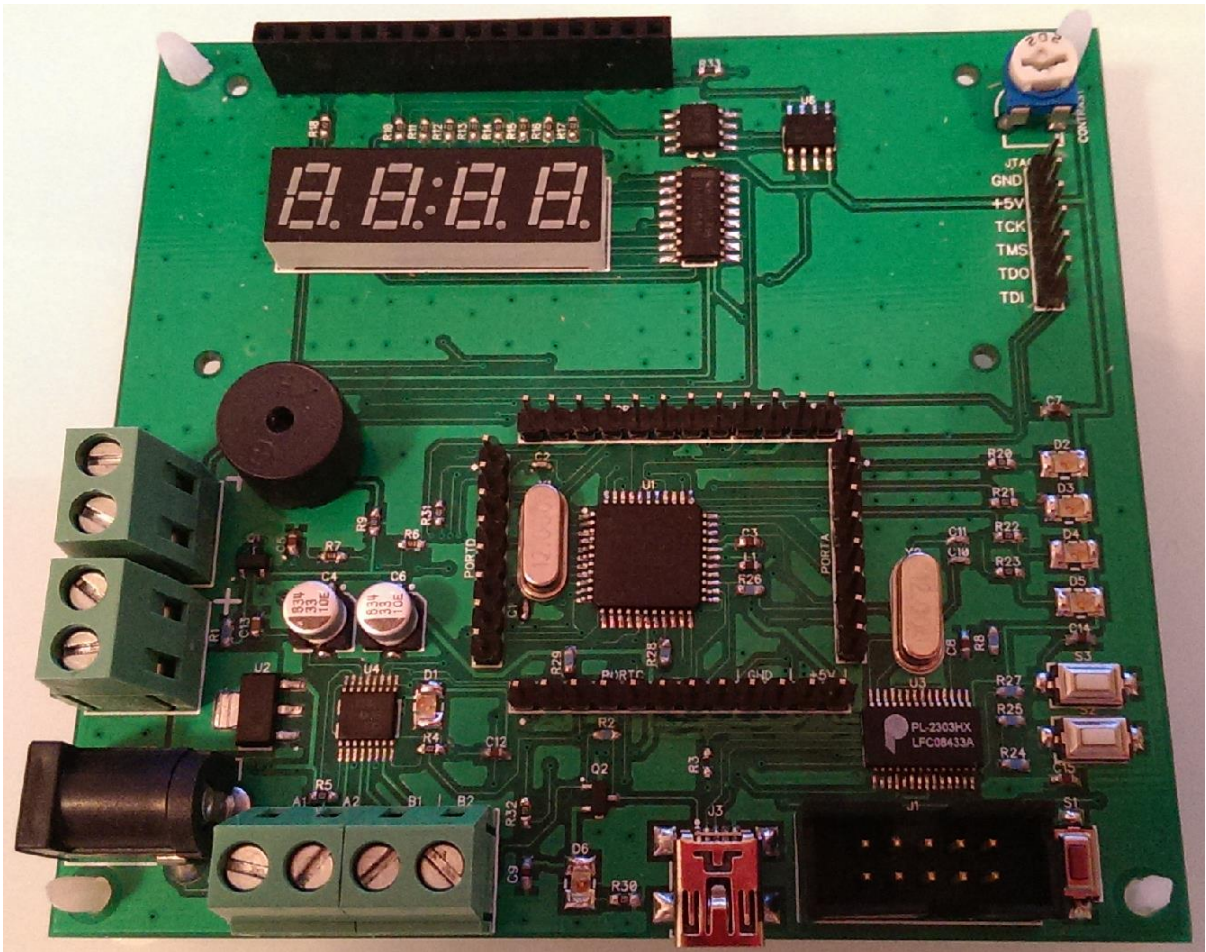


Рисунок 17. Готовое изделие.

3.7. Тестирование

Работоспособность любого готового изделия должна быть проверена, так как в процессе производства могли появиться какие-либо дефекты, например установка неверного компонента или «холодная пайка». Также есть риск, что на плате присутствуют дефектные компоненты.

Чтобы убедиться, что изделие готово к использованию пользователем, его необходимо протестировать. Для этих целей была написана тестовая программа. Конечно, она не проводит тестирование с покрытием 100%, но как минимум показывает, что главный контроллер функционирует и способен записать программу в свою память.

Также вместе с контроллером тест проверяет некоторые другие компоненты. Ниже представлена таблица тестовых методов.

Таблица 2. Методы функционального теста.

Название тестового метода	Тестируемый компонент	Ожидаемый результат
Проверка питания	D1	Диод светится желтым цветом
Проверка светодиодов	D2, D3, D4, D5	Диоды светятся желтым цветом
Проверка семисегментного дисплея	DS2	Все сегменты и разделители светятся красным цветом
Проверка жидкокристаллического дисплея	DS1	На дисплее виден текст с названием изделия
Проверка UART	U3	Изделие каждую секунду посылает на СОМ-порт компьютера сообщение с названием изделия на скорости 9600 бод.
Проверка датчика температуры	U5	На дисплее, вместе с названием, отображается температура в помещении.
Проверка громкоговорителя	SPK1	Раз в секунду плата издает короткий звук.

4. Область применения

Любая макетная плата дает пользователю представление о возможностях той или иной архитектуры. В нашем случае плата может отображать текстовые сообщения, измерять температуру, общаться с компьютером и многое другое. Ниже представлен небольшой список возможных вариантов применения.

Таблица 3. Возможные варианты применения макетной платы.

№	Возможное применение
1	Изготовление примитивного робота.
2	Изготовление устройства регулирования температуры.
3	Регулировка яркости освещения в помещении.
4	Разработку простого «умного дома», умеющего сберегать энергию, если в здании никого нет.
5	Разработка умных автомобильных ворот, которые открываются по звонку мобильного телефона или распознают автомобиль по идентификатору RFID.

4.1. Возможное применение на примере простого робота

Для простого примера соберем робота, который реагирует на препятствия. Для этого необходимо установить некоторые дополнительные детали.

Таблица 4. Дополнительные комплектующие робота.

Название детали	Количество
Инфракрасный светодиод	1
Фототранзистор	1
Резистор 470 Ом	2
Шасси	1
Двигатели	2

Рассмотрим блок-схему примитивного робота. Основной блок – это макетная плата с загруженной в память программой управления. Двигатели управляются контроллером в зависимости от показаний датчиков. Фототранзистор фиксирует отраженный от препятствия свет.



Рисунок 18. Блок-схема примитивного робота.

Макетная плата загружает программу, затем выполняет инициализацию и зажигает инфракрасный светодиод. Затем считывает данные с фототранзистора [8] о наличии препятствия. В зависимости от полученных данных управляет двигателями.

В качестве реакции будет поворот на 90 градусов влево и затем продолжение движения по прямой. Диаграмма состояний приведена ниже.

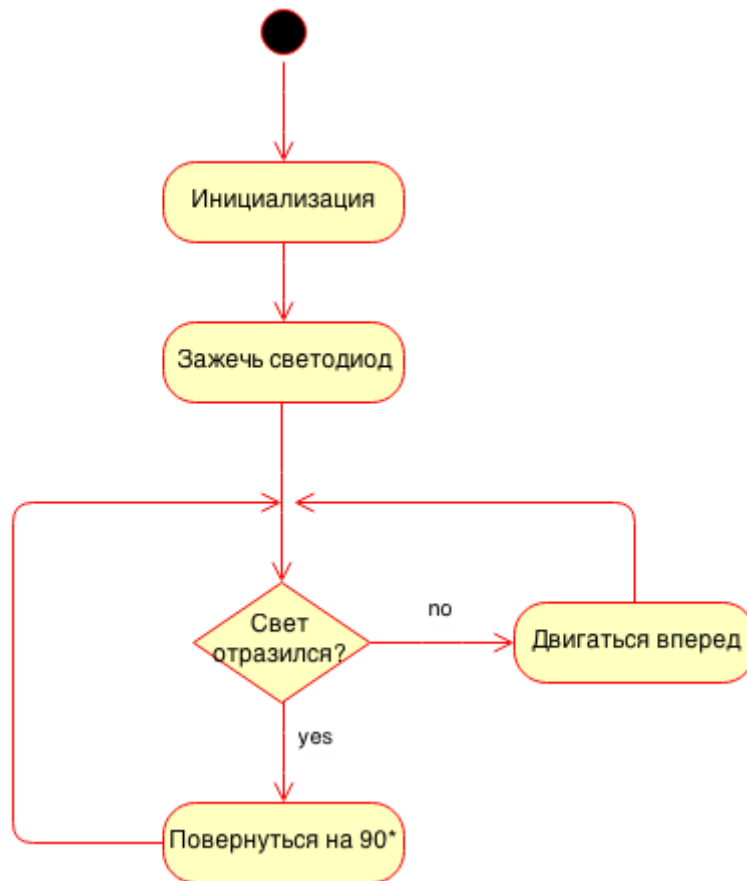


Рисунок 19. Диаграмма состояний робота.

После включения система производит инициализацию, то есть настраивает порты ввода-вывода в определенные состояния. Затем зажигает инфракрасный светодиод. Если свет от диода отразился в фототранзистор, значит впереди препятствие, поэтому совершаем поворот налево на 90 градусов. Снова проверяем наличие препятствия. Если оно отсутствует, то двигаемся вперед и снова выполняем проверку.

4.2. Дополнительные устройства для подключения

Список устройств, которые могут быть использованы для подключения к макетной плате.

- RFID приемник
- Карта Ethernet
- Массив оптических датчиков
- Радиомодуль Bluetooth
- Считыватель карты памяти
- Ультразвуковой датчик
- Датчик утечки природного газа
- Датчик дождя
- Датчик пламени
- Датчик движения
- Датчик ускорения
- Электронный барометр
- Электронный гигрометр
- GPS модуль
- Электронный компас

5. Заключение

В работе было проведено подробное описание процесса разработки макетной платы на основе архитектуры AVR. Начиная от сравнения и выбора элементов, и заканчивая разработкой самой платы в программе DipTrace. С последующим изготовлением прототипов и их тестированием.

Задачей работы было создать недорогую и простую в изучении макетную плату, которая подошла бы начинающим радиолюбителям как основа для разработки робота или другой электроники с использованием микроконтроллера Atmega16. Для этого на плате были установлены микросхемы управления моторами, разные датчики, а также средства обмена данными с пользователем. Главным контроллером на плате был использован 8-битный микроконтроллер Atmega16, имеющий исчерпывающую документацию и широкое сообщество любителей данной архитектуры. Это позволяет новичкам быстро находить информацию, изучать платформу и практически сразу приступить к разработке какого-либо изделия.

Также в задачи был включен пункт сделать макетную плату доступной для обычного радиолюбителя. Итоговая цена прототипа, которая включает в себя цену всех компонентов, печатной платы и стоимость сборки, составила около 20 евро за одну единицу. Такая цена вполне сравнима с ценами на обычные комплекты типа Arduino и PicKit.

После изготовления и тестирования стало очевидно, что некоторые узлы прототипа необходимо доработать, однако в целом макетная плата справляется со своими задачами.

Использованная литература

- [1] Техническое описание микроконтроллера Atmega16
- [2] Техническое описание микроконтроллера PIC16F887
- [3] Техническое описание микросхемы DRV8833
- [4] Техническое описание микросхемы LM1117
- [5] Техническое описание микросхемы PL2303HX
- [6] Техническое описание микросхемы AT24C01
- [7] Техническое описание микросхемы LM75AD
- [8] Техническое описание оптопары CNY70
- [9] Техническое описание интерфейса I2C –
http://www.nxp.com/documents/application_note/AN10216.pdf
- [10] Техническая документация программы DipTrace –
<http://www.diptrace.com/books/tutorial.pdf>
- [11] [WWW] – Электроника для начинающих – <http://www.easyelectronics.ru/>
- [12] Описание макетной платы PinBoard -
<http://shop.easyelectronics.ru/index.php?productID=147>
- [13] Принципиальная схема платы Arduino Uno –
<http://arduino.cc/en/uploads/Main/arduino-uno-schematic.pdf>
- [14] Принципиальная схема платы PisiXBee –
http://www.robotiklubi.ee/_media/projektid/pisi_xbee/skeemid/skeem.png