МИНОБРНАУКИ РОССИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ

ВЫСШЕГО ОБРАЗОВАНИЯ

«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Физический факультет

Кафедра электроники

**???????????????????????????????**

Дипломная работа

Направление 011801 Радиофизика и электроника

Профиль информационные системы и технологии

Зав. кафедрой \_\_\_\_\_\_\_\_ д. ф.-м. н., проф. А.М. Бобрешов \_\_\_.\_\_\_.20\_\_

*подпись*

Обучающийся \_\_\_\_\_\_\_\_ В.А. Сафонов \_\_\_.\_\_\_.20\_\_

*подпись*

Руководитель \_\_\_\_\_\_\_\_ к. геогр. н., доц. И.К.Астанин \_\_\_.\_\_\_.20\_\_

Воронеж 2018

**Содержание**

Введение…………………………………………………………………………...3

1.Теоретическая часть…………………………………………………………….4

1.1.Выбор и обоснование компонент системы…………………………..4

ВВЕДЕНИЕ

В каждом современном доме есть электрическое освещение. Создание концепции умного дома заставило людей задуматься: как должно выглядеть умное освещение? Какие дополнительные функции можно привнести в него?

Так как существует множество компаний разрабатывающие умные дома, то каждая из них ответила на этот вопрос по-своему. Но есть у них и нечто общее. Это высокая стоимость такого оборудования. При том, что себестоимость не сильно возрастает по сравнению с обычным, «не умным», освещением. Это и является одной из основных причин почему я выбрал данную тему. В ходе данной работы будет показано, что на самом деле «умное» освещение - это недорогая и простая система, которая может быть запросто установлена в любом доме.

Для выполнения работы нам потребуются: сами предметы освещения, управляющие элементы, способ управления.

Все эти предметы будут подробно рассмотрены и выбраны в дальнейшем ходе работы.

Также система должна удовлетворять некоторым простым критериям:

1.гибкость (легкость, а также простота использования, возможность настройки)

2.надежность (автоматический контроль над состояниями системы)

3.высокая функциональная возможность и простота в обращении

4.маленькой себестоимостью, большой экономичностью

5.Безопасность использования

**1. Теоретическая часть**

**1.1. Выбор и обоснование компонент системы**

Наиболее распространенной на данным момент является традиционная система центрального освещения с управлением от обычных переключателей. Но они нас не интересуют. Нас интересует «умное» освещение. Попробуем разбить их на отдельные части. Например, по используемым осветительным приборам. Бывают:

1.Лампы накаливания

2.Люминесцентные лампы

3.Галогенные и металлогалоидные источники света

4.Светодиоды и лампы на их основе

5.Световые шнуры «дюралайт» (Duralight)

6. Оптические волокна

Рассмотрим их преимущества и недостатки:

Спектр ламп накаливания наиболее комфортен для зрения, но они лишь около 10% своей мощности расходуют на создание света, а остальную преобразуют в тепловую энергию. Поэтому в помещение со слишком большим количеством лампочек накаливания всегда будет жарко и душно, как в фотосалоне или телевизионной студии. Люминесцентные лампы не нагреваются, но до недавнего времени их неохотно использовали для освещения жилья из-за некомфортного голубовато-белого света, неприятного гула и необходимости дополнительных пусковых устройств для подключения. Современное поколение энергосберегающих люминесцентных ламп свободно от этих недостатков, и они вполне могут использоваться вместо ламп накаливания. Галогенные и металлогалоидные светильники излучают довольно узкий направленный пучок света, а поэтому больше подходят для освещения отдельных зон или подсветки ключевых элементов интерьера — картин, скульптурных групп, колонн, ниш и т.д. Светодиоды прочны, не нагреваются, в их очень широком спектре нет вредных для здоровья человека инфракрасного и ультрафиолетового излучений. Они являются одним из самых экологически чистых источников света. Светодиодные лампы не используют веществ, содержащих [ртуть](https://ru.wikipedia.org/wiki/%D0%A0%D1%82%D1%83%D1%82%D1%8C), поэтому они не представляют опасности в случае выхода из строя или повреждения колбы. Долговечность светодиодов в 80 раз превышает ресурс ламп накаливания. Но светоотдача светодиодов не настолько велика, чтобы использовать их в роли самостоятельных источников света и светят они только в одном направлении, а поэтому в интерьере их используют чаще для декоративной подсветки или в качестве ночника. Световые шнуры «дюралайт» — это декоративные источники света на основе светодиодов, широко использующиеся для светового оформления элементов интерьера или в наружном уличном освещении. Оптические волокна излучают свет только с торцов, но удаляя их на большие расстояния от основного источника света и используя специальные насадки, можно создавать волшебные фантастические световые эффекты.

Рассмотрим еще чуть более подробно светодиодное освещение. У него есть один крупный недостаток, из-за которого, казалось бы, невозможно его использовать. Но эта проблема легко разрешается использование не отдельных светодиодов, а светильников со сменными светодиодными лампами. Эти лампы имеют в своем составе несколько светодиодов, что решает проблему недостаточной светоотдачи, и рассеиватель, который решает проблему узкой направленности. Этот вариант несколько дороже, но за счет низкого энергопотребления и долгого срока службы он быстро окупится. В своей работе я планирую использовать именно этот тип освещения.

Далее рассмотрим возможные типы управляющих устройств.

1.Обычные переключатели

2. [Механические](https://ru.wikipedia.org/wiki/%D0%94%D0%B8%D0%BC%D0%BC%D0%B5%D1%80#%D0%9C%D0%B5%D1%85%D0%B0%D0%BD%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B8%D0%B9_%D0%B4%D0%B8%D0%BC%D0%BC%D0%B5%D1%80) диммеры (на основе переменного резистора)

3. Электронные диммеры (на основе микроконтроллера)

Рассмотрим их плюсы и минусы:

Обычные переключатели обладают жестко ограниченным функционалом: включение и выключение. Этого недостаточно для «умного» освещения. Механические диммеры – это хоть и простое, но безнадежно устаревшее решение, добавляющее к предыдущему только одну функцию – регулировка уровня освещенности. А вот функционал электронных диммеров ограничен только возможностями микроконтроллера, которые, в свою очередь, ограничены только фантазией и навыками разработчика. Именно этот вариант я и буду использовать.

Остается только рассмотреть способы управления освещением:

1.Механическое управление

2.Электронное управление

2.1. Контактное

2.2. Бесконтактное

3. Дистанционное управление

4. Акустическое управление

Изучим их подробнее:

В основе механического диммера [потенциометр](https://ru.wikipedia.org/wiki/%D0%9F%D0%BE%D1%82%D0%B5%D0%BD%D1%86%D0%B8%D0%BE%D0%BC%D0%B5%D1%82%D1%80), подключённый не непосредственно к нагрузке, а передающий сигнал через схему управления на силовой элемент ([реостат](https://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D0%BE%D1%81%D1%82%D0%B0%D1%82), [дроссель](https://ru.wikipedia.org/w/index.php?title=%D0%AD%D0%BB%D0%B5%D0%BA%D1%82%D1%80%D0%BE%D0%BD%D0%BD%D1%8B%D0%B9_%D0%B4%D1%80%D0%BE%D1%81%D1%81%D0%B5%D0%BB%D1%8C&action=edit&redlink=1), [тиристор](https://ru.wikipedia.org/wiki/%D0%A2%D0%B8%D1%80%D0%B8%D1%81%D1%82%D0%BE%D1%80)). Электронное управление устроено на использовании всевозможных датчиков. Отсюда и деление на контактный и бесконтактный. Если первый устроен на использовании, например, сенсорного управления, то второй все различные датчики движения и т.п. В дистанционном управлении используются всеразличные пульты дистанционного управления, использующие какой-либо канал связи, или излучающие инфракрасные или радио волны. Акустическое управление основано на реакции прибора на различные громкие звуки или на голосовые команды. Т.е. его можно рассматривать как голосовое управление. Но это довольно дорогая и сложная система. В своей работе я буду использовать дистанционное управление. Конкретный выбор и схема управления будет рассмотрены позже. Хотя в одном приборе и могут одновременно сосуществовать несколько различных способов управления, в своем проекте я не нахожу это целесообразным.

В конце данного пункта уже можно в общем виде обрисовать мою систему. Это система на основе светодиодных светильников с микроконтроллерным управлением, которые, в свою очередь, будут управляться дистанционно. Все эти компоненты и их реализации будут более подробно рассмотрены в отдельных пунктах далее.

**1.2 Выбор микроконтроллера**

Микроконтроллер - это самостоятельная компьютерная система, которая содержит процессор, вспомогательные схемы и устройства ввода-вывода данных, размещенные в общем корпусе.

Впервые микроконтроллеры появились в 1971 году, как и микропроцессоры общего назначения. Создатели микроконтроллеров разработали остроумную идею - совместить процессор, ПЗУ, оперативную память и периферию, поместив это внутри одного корпуса, по внешнему виду похожего на обыкновенную микросхему. С того момента производство микроконтроллеров из года в год во много раз превосходит производство процессоров, а потребность в них не уменьшается.

Но по-настоящему история микроконтроллеров начинается с выпуска компанией intel микроконтроллера intel8051 в 1980 году. Он является микроконтроллером второго поколения и является устройством с большим объемом ROM и EPROM памяти, новыми портами ввода-вывода и возможностью добавления внешней памяти. К тому же, он имел максимальную тактовую частоту 20MHz, то есть был достаточно быстрым. 8051 микроконтроллеры в скором времени начали производиться огромным количеством компаний в различных модификациях. Некоторые из этих модификаций производятся и в наше время. Микроконтроллеры с 8051 архитектурой имею следующие типовые характеристики:

* Тактовая частота 24 MHz
* Командный цикл в 12 тактов
* Объем памяти программ 4К
* Объем RAM 128байт
* Число линий ввода-вывода – 32
* Два 8\16 разрядных таймера
* Множество внутренних и внешних прерывателей
* Программируемый последовательный порт
* Интерфейс с внешней памятью и объемом до 128К

К фирмам производителям 8051 микроконтроллера относятся такие гиганты, как intel, Atmel, Dallas Semiconductor, Philips Semiconductor.

Микроконтроллеры AVR производят десятки компаний, при этом изготавливают как 32-битные современные микроконтроллеры, так и 16-битные, и даже 8-битные (такие, как i8051 и аналоги). В каждом семействе нередко можно повстречать практически одинаковые модификации, отличающиеся скоростью работы их процессора и объемом содержащейся памяти. Дело состоит в том, что микроконтроллеры используются предпочтительно во встроенных системах: в станках, в бытовой технике и автоматике, в любых автоматических устройствах - там, где необходима не столько мощность процессора, сколько, баланс между стоимостью и необходимой функциональностью.

Поэтому самые старые разновидности микроконтроллеров применяются до сих пор - они все еще многое могут: от автоматического управления дверями, включения/отключения освещения или полива газонов до управления автоматической системой «умный дом». Так же и существуют и наиболее мощные микроконтроллеры, выполняющие сотни миллионов команд в секунду и управляющие всевозможной периферией. У таких микроконтроллеров и предназначения соответствующие. Разработчик, таким образом, вначале оценивает задачу, а уж затем подбирает под нее подходящие элементы.

Сегодня существует намного более 200 разновидностей микроконтроллеров, совместимых с i8051, производимых двумя десятками фирм, и огромное количество микроконтроллеров иных типов. Большой популярностью у конструкторов пользуются 8-битные PIC-контроллеры компании Microchip Technology и AVR компании Atmel, 16-битные MSP430 компании TI, а также 32-битные микроконтроллеры с архитектурой ARM, разработку которой ведет компания ARM Limited и продаёт другим компаниям лицензии для их производства.

**1.2.1 Типовая структура микроконтроллера.**

Микроконтроллер (рис. 1) содержит в своем составе микропроцессорное ядро с развитой системой команд, память программ IROM,

регистровый файл данных RRAM, а также набор программируемых

интерфейсных схем, выполняющих ряд важных функций внутри МК

и обеспечивающих связь с внешней средой. Большинство из перечисленных блоков являются обязательными компонентами любого

МК, некоторые, например IROM, в структуре конкретного МК могут отсутствовать. Состав и назначение интегрированных на кристалл

ПУ определяется областью преимущественного применения МК

и может широко варьироваться в зависимости от типа МК. В целях

повышения гибкости использования аппаратных средств, интегрированных на кристалле, эти средства делаются многофункциональными с программной настройкой на тот или иной режим работы.

При инициализации МК информация о типе настройки заносится

в специальные регистры и в процессе дальнейшей работы МК

обычно остается неизменной. В большинстве случаев в число встроенных ПУ входят таймеры/счетчики, последовательные и параллельные порты

ввода - вывода, контроллер прерывания, многоканальный АЦП, сторожевой таймер и пр.

**1.2.3 Микроконтроллеры Motorola.**

Motorola 6800 был одним из первых доступных 8-разрядных микропроцессоров. Несмотря на то, что его архитектуре уже много лет, он до сих пор жизнеспособен и используется в качестве процессорного ядра для микроконтроллеров.

Компания Motorola создала весьма полный функциональный ряд микроконтроллеров 68НС05, развитием которого являются микроконтроллеры семейства 68НС08. Наиболее широкие функциональные возможности обеспечивают микроконтроллеры семейства 68HCI1, которые имеют несколько другую архитектуру и реализуют расширенный набор функций, среди которых, например, имеется команда деления. Семейства микроконтроллеров 68НСхх имеют архитектуру Фон-Неймана или Принстонскую, а их ядром является CISC-процессор (компьютер со сложным набором команд). Это делает семейства 68НСхх достаточно уникальными среди микроконтроллеров, но для реализации широкого круга приложений, от пейджеров до автомобильных контроллеров, их архитектура оказывается наиболее подходящей.

Имея Принстонскую архитектуру, 68НСхх могут выполнять некоторые приложения более эффективно, чем другие микроконтроллеры. Микроконтроллеры HCl I, как их чаще всего называют, имеют не очень много отличий, но обеспечивают существенно лучшие показатели по производительности.

На 2001 год насчитывалось уже более 180 моделей 68НС05 (без учета разнообразия типов корпусов), которые обладают различными характеристиками и функциональными особенностями возможностей, предоставляемых семействами 68НСхх, поистине поражает.

Для лучшего представления о назначении отдельных микроконтроллеров, семейство НС05 разделено на серии, каждая из которых маркируется буквенным кодом. Система буквенных обозначений также используется для маркировки микроконтроллеров 68НС08 и 68HCU, однако одинаковые буквенные индексы в обозначении приборов из разных семейств соответствуют различным функциональным возможностям. Многие из этих микроконтроллеров разработаны не для общего пользования, а для специального применения. Это означает, что вы можно найти микроконтроллер, который полностью соответствует требованиям, но недоступен для использования, так как поставляется только по специальным заказам. Наилучшим способом выбора микроконтроллеров из семейств 68НС\х является выбор прибора, удовлетворяющего требованиям, из каталогов дистрибьюторов продукции компании Motorola, вместо непосредственного поиска по фирменной документации.

**1.2.4 Микроконтроллеры 0851.**

Одновременно с созданием в 70-х годах первых микропроцессоров компания INTEL вела также разработку микроконтроллеров. Усилия по разработке микроконтроллеров воплотились в архитектуру 8051, которая впервые была представлена в 1980 году и стала одной из наиболее популярных микроконтроллерных архитектур. Микроконтроллер второго поколения 8051 представляет собой законченное устройство с большим объемом встроенной памяти программ (ROM и EPROM) и данных RAM, улучшенными портами ввода-вывода и возможностью подключения внешней памяти Первые микроконтроллеры 8051 были достаточно быстрыми, имея максимальную тактовую частоту 20MHz. В то время как популярность большинства других типов микроконтроллеров определяется объемом их продаж, 8051 снискали другую славу.

Микроконтроллеры с архитектурой 8051 производятся наибольшим числом компаний. В этом заключается их несомненное преимущество, так как каждый производитель производил усовершенствования 8051 с целью расширения функциональных возможностей или повышения скорости. Благодаря этому архитектура 8051 стала весьма привлекательной для реализации широкого диапазона приложений.

Микроконтроллеры 8051 имеют следующие типовые характеристики:

* тактовая частота - 24 МГц;
* объем памяти программ - 4К;
* командный цикл - 12 тактов;
* объем памяти данных RAM - 128байт;
* число линий ввода-вывода - 32;
* два 8/16-разрядных таймера;
* множество внутренних и внешних источников прерываний;
* программируемый последовательный порт;
* интерфейс с внешней памятью объемом до 128К.

Многие производители провели усовершенствования и расширили возможности выпускаемых ими микроконтроллеров данной архитектуры.

**1.2.5 Микроконтроллеры PIC**

Микроконтроллеры семейства PICMicro относятся к числу доминирующих и наиболее эффективных микроконтроллеров [2]. По такому параметру как рабочая скорость, которая определяется количеством команд, выполняемых за один такт.

Особенностью данного семейства является большое количество новых моделей выпускаемых фирмой Microchip - свыше 40 за 1997 год. Эти микроконтроллеры широко известны под именем PIC.

Архитектура PIC была разработана достаточно давно. Первоначально она была предложена отделением компании General Instrument, которое занималось производством микросхем. Позже это отделение выделилось в качестве самостоятельной фирмы Microchip, которая начала проводить агрессивную компанию по продвижению на рынок микроконтроллеров PICMicro и серии микросхем памяти EEPROM с последовательным доступом.

Различают 3 модели семейства PICMicro:

**Младшие модели семейства PICMicro**

Младшие модели серии 16С5х реализуют нижний уровень возможностей микроконтроллеров семейства PICMicro. Они выполняют подмножество команд, реализуемых старшими моделями, и программно совместимы с ними. Не рекомендуется использовать эти устройства для разработки новых приложений для PICMicro. Недостаток прерываний, малый объем доступной памяти программ и RAM, отсутствие внутрисистемного программирования и портов ввода/вывода с расширенными функциями являются причинами трудностей, которые возникают при реализации приложений на различных моделях микроконтроллеров этого семейства.

Решающим фактором перехода от младших моделей к моделям среднего уровня послужило то обстоятельство, что последние продаются по аналогичной и даже меньшей цене и в таких же корпусах, как и первые.

Рассмотрим младшие модели микроконтроллеров с восьмью выводами.

PICMicro серии 12С5хх. Обладая 6-ю линиями ввода/вывода и объемом памяти программ 512 или 1024 команд, эти устройства используются для создания небольших и простых приложений. Микроконтроллеры этой серии, возможно, являются наилучшими простыми интерфейсными устройствами.

**Средние модели семейства PICMicro**

Модели среднего уровня образуют законченную линию микроконтроллеров со множеством различных функций. Они имеют такую же базовую архитектуру, как микроконтроллеры младших моделей, но существенно отличаются возможностями обработки прерываний.

В моделях среднего уровня реализован следующий набор «стандартных» свойств:

количество уровней прерываний (подпрограмм) - 8;

объем памяти программ - 512-8 К команд;

регистровый файл (RAM) - 36-192 байт;

число источников прерываний - 4-12;

число таймеров - 1-3;

возможность внутрисхемного программирования;

напряжение питания - 2,0 - 5,0 В;

тип корпуса - пластиковый ОТР, керамический с окном,S01C, ОFР;

число линий ввода/вывода - 13-33;

тактовая частота - от 0 до 4 МГц или до 20 МГц;

тип памяти программ - масочно-программируемая, ЕРROM, EEPROM/Flash;

возможности АЦП - Компараторы напряжения, ЦАП с резистивным делителем, АЦП интегрирующего типа, ШИМ выходы;

последовательные порты - SPI. I2C, асинхронный.

другие порты - Порог прямого подключения ЖКИ, параллельный ведомый порт.

Широкий диапазон возможностей, включая внутрисистемное программирование, дает этим микроконтроллером значительные преимущества, позволяй использовать один и тот же программный код при реализации разнообразных приложений.

**Старшие модели семейства PICMicro**

Старшие модели микроконтроллеров PICMicro используют 16-битные команды и могут адресовать 64К слов памяти. Так как каждое слово содержит 16 бит, то старшие модели серии 17Схх могут адресовать по 128 Кбайт памяти программ и памяти данных.

Для обеспечения чтения/записи памяти программ в серии 17Схх используется модифицированное ядро центрального процессора. Это ядро позволяет различным командам обращаться ко всем регистрам процессора. Это повышает гибкость микроконтроллера и позволяет использовать команды в широком спектре приложений. В то время как в микроконтроллерах среднего уровня используется только один вектор прерывания, в старших моделях PICMicro имеется несколько векторов прерываний. Старшие модели PICMicro разрабатывались, в основном, для взаимодействия с другими цифровыми устройствами. По этой причине в микроконтроллерах серии 1 7Схх нет АЦП и устройств, обеспечивающих непосредственное подключение датчиков, которые имеются в моделях среднего уровня.

**1.2.7 Микроконтроллеры AVR**

Микроконтроллеры AVR фирмы Amtel открывают одно из новых направлений в области разработки и архитектуры микроконтроллеров. Структура процессора AVR представляется собой RISC-архитектуру с повышенной производительностью и пониженным энергопотреблением Гарвардского типа.

Одним из основных достоинств этого контроллера является быстрое выполнение команд – он выполняет команду за один такт. АVR имеет, наиболее разносторонний по своим возможностям процессор из всех микроконтроллеров. Это означает, что при разработке приложений надо

потратить немного больше времени на планирование размещения данных в

памяти и регистрах, чем для других микроконтроллеров. Но благодаря своей

разносторонности AVR очень прост в программировании на языке «Ассемблер» так и для языков высокого уровня, например С#.

Главным преимуществом AVR является наличие памяти EEPROM для хранения программ с возможностью программирования в системе, а также расширенный набор команд с возможностью выполнения большинства команд за один машинный цикл.

**1.2.8 Микроконтроллеры STM**

Компания STMicroelectronics одной из первых вывела на рынок семейство микроконтроллеров на ядре ARM Cortex-M3 и на сегодняшний день по праву занимает лидирующее место среди производителей микроконтроллеров на этом ядре. Все началось в 2007 году с двух семейств — Performance Line (STM32F103) и Access Line (STM32F101). Компания постоянно работает как над расширением номенклатуры семейства, так и над улучшением характеристик, не забывая при этом также пополнять программную составляющую продукта. На сегодняшний момент [STM32](http://catalog.compel.ru/mcu/list?BRAND=ST)уже состоит из 10 линеек для всевозможных применений — микроконтроллеры с высокой производительностью, недорогие микроконтроллеры общего применения, микроконтроллеры с ультранизким энергопотреблением, микроконтроллеры со встроенным радиомодулем для беспроводных решений, и все это — на одном ядре ARM Cortex-M3! Нельзя не отметить pin-to-pin и программную совместимость по всем линейкам.

**1.3 Выбор языка программирования**

Языки программирования микроконтроллеров по своей структуре мало отличаются от классических языков для компьютеров. Единственным отличием становится ориентированность на работу со встроенными периферийными устройствами. Архитектура микроконтроллеров требует, например, наличия битово-ориентированных команд. Последнее позволяют выполнять работу с отдельными линиями портов ввода/вывода или флагами регистров. Подобные команды отсутствуют в большинстве крупных архитектур. Даже ядро ARM, активно применяемое в микроконтроллерах, не содержит битовых команд, вследствие чего разработчикам пришлось создавать специальные методы битового доступа.

**1.3.1 Ассемблер**

Ассемблер является  языком самого низкого уровня. При этом он позволяет наиболее полно раскрыть все возможности микроконтроллеров и получить максимальное быстродействие и компактный код. В некоторых случаях альтернативы ассемблеру нет, но, тем не менее, он имеет множество недостатков. Несмотря на получаемую компактность машинного кода, программа, написанная на языке Ассемблер, громоздка и труднопонимаема. Для ее создания требуется отличное знание архитектуры  и системы команд микроконтроллеров.

Ассемблер отлично подходит для программирования микроконтроллеров, имеющих ограниченные ресурсы, например 8-ми битных моделей с малым объемом памяти. Для больших программ и тем более 32-разрядных контроллеров, лучше использовать другие языки,  отличающиеся более высоким уровнем. Это позволит создавать более сложные и при этом понятные программы.

**1.3.2 Pascal**

Язык Pascal еще более удобен для восприятия и изучения. Тем не менее, он не имеет такого распространения как C/C++, особенно при программировании микроконтроллеров. Некоторые отдельные фирмы поддерживают данный язык, с целью упрощения перехода на контроллеры с больших ПК. В частности вариант языка под названием MicroPASCAL входит в состав поставки отладочных средств фирмы Mikroelektronika.

**1.3.3 BASIC**

Старинный язык первоначального обучения программированию, в настоящее время в основном сохранился в виде реализации Visual BASIC от Microsoft. Используется он и для программирования микроконтроллеров. Реализаций этого языка гораздо больше, чем того же Pascal. Связано это в первую очередь с простотой языка. BASIC часто выбирают разработчики программно-аппаратных платформ, нацеленных на упрощенную разработку электронных устройств. Можно назвать такие проекты,  как PICAXE, Amicus18, microBASIC и некоторые другие. Недостатком BASIC является плохая структурированность кода. Этот язык не стоит выбирать для первоначального изучения с целью дальнейшего перехода на С/С++.  Программирование микроконтроллеров на BASIC можно рекомендовать любителям, не нацеленным на создание, в основном, простых устройств.

**1.3.4 С/С++**

Язык программирования С/С++, относится к языкам более высокого уровня, по сравнению с  Ассемблером. Программа на этом языке лучше понятна человеку. Достоинством С/С++ является огромное число программных средств и библиотек, позволяющих просто создавать необходимый код. Фактически, С/С++ сегодня стал основным языком разработки управляющих программ. Компиляторы данного языка реализованы практически для всех моделей микроконтроллеров. Стандартный язык дает возможность переноса программ с одной платформы на другую. Теоретически, используя разные компиляторы, можно преобразовать любую программу в команды микроконтроллера нужного типа. На практике дополнительно требуется учитывать архитектуру микроконтроллера каждого типа.

Язык С/С++ имеет достаточно сложную для изучения структуру. Получаемый программный код конкретной задачи, имеет больший объем, чем код той же задачи, реализованной на Ассемблере. Тем не менее язык С/С++ следует признать единственным правильным выбором для профессионального программирования микроконтроллеров.

**1.3.5 Визуальные языки**

В отличие от классических языков программирования, визуальные языки позволяют разрабатывать программы в виде изображений . Среди таких языков можно выделить FlowCODE или Scratah. Достоинством визуальных языков является хорошо воспринимаемая структура алгоритма. Это позволяет просто разобраться в его функционировании любому человеку, знающему основные символы языка. Перевод структурных схем в команды микроконтроллера, как правило, выполняется не сразу. Вначале алгоритм транслируется в команды ассемблера или какого-либо языка высокого уровня. Только затем, все преобразуется в машинный код. Такая схема, несмотря на свою сложность, позволяет использовать наиболее удобные компиляторы разных разработчиков.

Еще одним достоинством визуального программирования становится простота изучения. Недостатком визуального подхода является громоздкость исходных материалов. Тем не менее, подобные языки программирования нашли очень большое распространение для решения специальных задач.

**1.3.6 Выбор языка программирования**

Выбор того или иного языка программирования зависит от множества факторов. В первую очередь необходимо определиться с типом решаемых задач и необходимым качеством кода. Если не требуется разработка объемных и сложных программ, то можно использовать практически любой язык. Для обеспечения компактности кода подойдет Ассемблер, а если ставятся серьезные задачи, то альтернативы С/С++ практически нет. Также необходимо учитывать доступность компилятора. В некоторых случаях, реализация языка может вообще отсутствовать, или предлагаться за солидные деньги. В итоге самым универсальным решением можно назвать связки Ассемблер и  C/C++.