

Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Саратовский государственный технический университет  
имени Гагарина Ю.А.»

Филиал федерального государственного бюджетного образовательного  
учреждения высшего образования  
«Саратовский государственный технический университет  
имени Гагарина Ю.А.» в г. Петровске

УТВЕРЖДАЮ  
Директор филиала СГТУ  
имени Гагарина Ю.А. в г. Петровске  
Е.А. Бесшапошникова  
«30» июня 2021 г.



## МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ВЫПОЛНЕНИЮ ЛАБОРАТОРНЫХ РАБОТ


по дисциплине

МДК 03.01. «Методы технического обслуживания и ремонта установок для  
аддитивного производства»

направление подготовки

15.02.09 «Аддитивные технологии»

Методические указания рассмотрены  
на заседании предметной (цикловой) комиссии  
общепрофессиональных дисциплин,  
профессиональных модулей специальностей  
технического профиля  
«14» июня 2021 года, протокол №13

Председатель ПЦК  /Т.А.Лескина/

Петровск 2021

### **Пояснительная записка**

Методические указания по выполнению лабораторных работ подготовлены на основе рабочей программы учебной дисциплины МДК 03.01. «Методы технического обслуживания и ремонта установок для аддитивного производства», разработанной на основе ФГОС СПО по специальности 15.02.09 «Аддитивные технологии» и соответствующих общих (ОК) и профессиональных (ПК) компетенций:

ОК 01. Понимать сущность и социальную значимость своей будущей профессии, проявлять к ней устойчивый интерес.

ОК 02. Организовывать собственную деятельность, выбирать типовые методы и способы выполнения профессиональных задач, оценивать их эффективность и качество.

ОК 03. Принимать решения в стандартных и нестандартных ситуациях и нести за них ответственность.

ОК 04. Осуществлять поиск и использование информации, необходимой для эффективного выполнения профессиональных задач, профессионального и личностного развития

ОК 05. Использовать информационно-коммуникационные технологии в профессиональной деятельности.

ОК 08. Самостоятельно определять задачи профессионального и личностного развития, заниматься самообразованием, осознанно планировать повышение квалификации.

ОК 09. Ориентироваться в условиях смены технологий в профессиональной деятельности.

ПК 3.1. Диагностировать неисправности установок для аддитивного производства

ПК 3.2. Организовывать и осуществлять техническое обслуживание и текущий ремонт механических элементов установок для аддитивного производства

ПК 3.3. Заменять неисправные электронные, электронно-оптические, оптические и прочие функциональные элементы установок для аддитивного производства и проводить их регулировку

При выполнении лабораторных работ студент должен *знать*:

- физические принципы работы, конструкцию, технические характеристики, правила технического обслуживания установок для аддитивного производства;

- элементы систем автоматики, основные характеристики и принципы их применения в аддитивных установках и вспомогательном оборудовании;

- классификацию и назначение электроприводов, физические процессы в электроприводах;

- выбор элементов схемы электроснабжения и защиты;

- технологию ремонта установок для аддитивного производства, вспомогательного оборудования и пускорегулирующей аппаратуры;

- действующую нормативно-техническую документацию по

специальности;

- правила сдачи оборудования в ремонт и приема после ремонта;
- порядок проведения стандартных и сертифицированных испытаний;
- пути и средства повышения долговечности оборудования

При выполнении лабораторных работ студент должен *уметь*:

- проводить анализ неисправностей электрооборудования;
- подбирать технологическое оборудование для ремонта и эксплуатации аддитивных установок и вспомогательных электромеханических, электротехнических, электронных и оптических устройств и систем, определять оптимальные варианты его использования;
- организовывать и выполнять наладку, регулировку и проверку установок для аддитивного производства;
- осуществлять метрологическую поверку изделий;
- производить диагностику оборудования и определение его ресурсов;
- прогнозировать отказы и обнаруживать дефекты аддитивных установок, осуществлять технический контроль при их эксплуатации;
- эффективно использовать материалы и оборудование;
- заполнять маршрутно-технологическую документацию на эксплуатацию и обслуживание аддитивных установок.

Содержание лабораторных работ определено рабочей программой и тематическим планированием, соответствует теоретическому материалу изучаемых разделов учебной дисциплины.

Объем лабораторных работ по дисциплине определяется учебным планом по данной специальности.

Продолжительность лабораторной работы – 2 академических часа. Перед проведением лабораторной работы преподавателем организуется инструктаж, а по ее окончании – обсуждение итогов.

Комплект методических указаний по выполнению лабораторных работ по дисциплине МДК 03.01. «Методы технического обслуживания и ремонта установок для аддитивного производства» содержит 10 лабораторных занятий.

**Перечень лабораторных работ  
по дисциплине МДК 03.01. «Методы технического обслуживания и  
ремонта установок для аддитивного производства»**

**ЛАБОРАТОРНАЯ РАБОТА № 1.**

Тема: Работа с программатором

**ЛАБОРАТОРНАЯ РАБОТА № 2.**

Тема: Подключение к программатору кабелей и адаптеров

**ЛАБОРАТОРНАЯ РАБОТА № 3.**

Тема: Измерители влажности

**ЛАБОРАТОРНАЯ РАБОТА № 4.**

Тема: Измерители мощности

**ЛАБОРАТОРНАЯ РАБОТА № 5.**

Тема: Измерители параметров электробезопасности, токовые клещи, кабель-тестеры, калибраторы портативные, мегаомметры и омметры, измерители шума и вибрации

**ЛАБОРАТОРНАЯ РАБОТА № 6.**

Тема: Измерители параметров электробезопасности, токовые клещи, кабель-тестеры, калибраторы портативные, мегаомметры и омметры, измерители шума и вибрации

**ЛАБОРАТОРНАЯ РАБОТА № 7.**

Тема: Подбор контроллера

**ЛАБОРАТОРНАЯ РАБОТА № 8.**

Тема: Программирование контроллера G-кодом

**ЛАБОРАТОРНАЯ РАБОТА № 9.**

Тема: Настройка в программном обеспечении Marlin

**ЛАБОРАТОРНАЯ РАБОТА № 10.**

Тема: Тестирование контроллера

# ЛАБОРАТОРНАЯ РАБОТА № 1

## Тема: Работа с программатором

**Цель работы:** научиться производить программирование микросхем с помощью программатора.

**Оборудование:** ПК, программатор CH341A, микросхема ПЗУ 25L8005, инструкции по выполнению работы

### Справочный материал:

Программаторы - это устройства, предназначенные для программирования микросхем памяти (EPROM, EEPROM, FLASH), внутренней памяти микроконтроллеров и микросхем программируемой логики (PLD).

Его основная часть – это микросхема CH341A. Рядом с ней располагается кварцевый резонатор на 12 МГц, а также стабилизатор напряжения AMS1117, который выдаёт 3,3 вольта. По бокам от USB разъёма, которым программатор подключается к компьютеру, располагаются светодиодные индикаторы: сверху – индикатор питания (POWER), а снизу – индикатор обмена данными между ПК и программатором (RUN). Он включается, когда программатор считывает данные из программируемой микросхемы и когда происходит запись.



Для подключения программируемых микросхем установлена 16-контактная DIP панель с нулевым усилием (ZIF), которая промаркирована TFXTDOL. С обеих сторон панели располагаются две группы контактов. Их назначение описано на нижней стороне программатора



По названиям выводов понятно, что верхняя гребёнка предназначена для обмена по интерфейсу SPI, а нижняя – по UART. Также тут имеется площадка для пайки, на которую можно припаять программируемую микросхему.

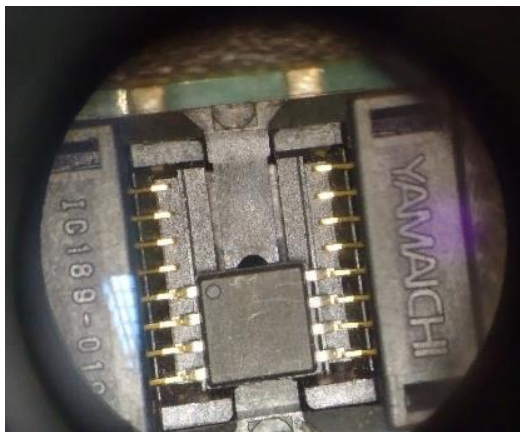
С помощью перемычки, которая по умолчанию установлена между контактами 1 и 2, можно менять режим работы программатора. Так, если перемычка установлена между контактами 1 и 2, программатор работает в параллельном режиме и определяется в диспетчере устройств Windows как параллельный порт (USB-EPP/I2C), а если между контактами 2 и 3 – в последовательном режиме и определяется в диспетчере устройств как COM-порт.

Программатор предназначен для чтения и записи данных в микросхемы flash-памяти серий 24 и 25.

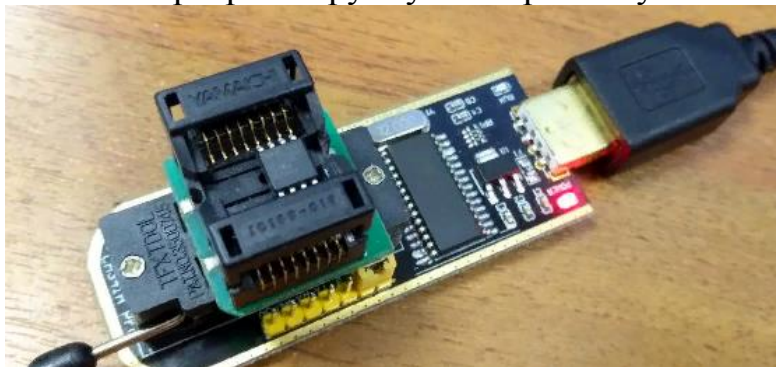
### Порядок выполнения работы:

#### Задание 1. Произвести работу с программатором

1. Установим программируемую микросхему в DIP-панель и зажмём с помощью специального рычага. Первая ножка микросхемы flash-памяти обозначена на корпусе точкой.



2. На нижней стороне программатора, как мы уже видели, отмечено, как необходимо располагать программируемую микросхему



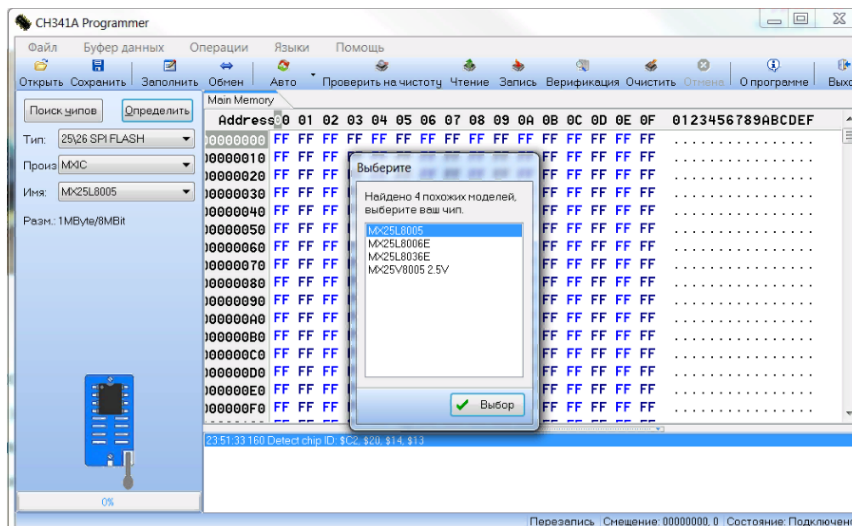
3. Будьте предельно внимательны при установке программируемой микросхемы. Если её неправильно (и неудачно) подключить, можно вывести из строя или микросхему, или сам программатор. Явным признаком неправильного подключения микросхемы может служить сильный разогрев частей программатора или программируемой микросхемы.

4. Подключим программатор к компьютеру, перемычка установлена между контактами 1 и 2. При первом запуске программатора необходимо установить драйверы.

5. После установки драйвера запустим программу **CH341A Programmer**. Программа автоматически определит, что программатор подключён. В правом нижнем углу в статусной строке программы появится надпись, оповещающая о том, что программа нашла программатор: *Состояние: Подключено*

6. Если программа не определила программатор, статусная строка отобразит соответствующее предупреждение. Нажмите кнопку «Определить». Программа, если сможет, покажет наиболее подходящие варианты. Также вы можете выбрать тип микросхемы вручную, нажав кнопку «Поиск чипов».





7. После того, как чип выбран, нажмите кнопку «Чтение». Программа прочитает и отобразит содержимое чипа в шестнадцатеричном формате (а также в виде текстовых символов в кодировке ASCII).

8. Для записи данных в ПЗУ нужно ввести в поле представления данных в 16-ном формате необходимый массив байтов, а затем нажать кнопку «Запись». Программа начнёт запись данных в микросхему флеш-памяти. Светодиодный индикатор RUN на программаторе загорится оранжевым цветом. По завершении записи программа проверит успешность записи, сверив переданный массив с содержимым в памяти микросхемы, а индикатор погаснет. Можно убедиться в том, что данные успешно записаны, отключив программатор от компьютера, а затем подключив его и заново считав содержимое ПЗУ.

9. Можно сохранить считанный из ПЗУ массив данных в файл. Для этого нужно просто нажать кнопку «Сохранить» и указать желаемое имя файла. А можно, наоборот, загрузить в память данные из файла, нажав кнопку «Открыть».

10. Расширения файлов для хранения данных ПЗУ – \*.bin, \*.hex и \*.rom.

### **Оформление отчета:**

Отчет должен содержать:

- 1) формулировку цели работы;
- 2) конспект практической части;
- 3) анализ полученных результатов.

### **Контрольные вопросы:**

1. Что такое программатор?
2. На какие группы по функциональным возможностям можно разделить программаторы?
3. Перечислите основные функциональные возможности программаторов.

## ЛАБОРАТОРНАЯ РАБОТА № 2.

### Тема: Подключение к программатору кабелей и адаптеров

**Цель работы:** получение начальных сведений о работе с интегрированной средой разработки, внутрисхемной отладке программ и программировании внутренней памяти микроконтроллеров K1886BE61Y; ознакомление с интерфейсом графической оболочки DEVCPP.EXE, ознакомление со структурой директорий среды разработки

**Оборудование:** ПК, отладочная плата Eval17 на основе микроконтроллеров K1886BE61Y, инструкции по выполнению работы

### Справочный материал:

Отладочный комплект для микроконтроллера K1886BE61Y предназначен для демонстрации функционирования данных микроконтроллеров и их основных периферийных модулей, для начального обучения программированию микроконтроллеров с помощью прилагаемой демонстрационной программы; а также для отладки собственных проектов с применением установленных на плате блоков и возможностью макетирования дополнительной схемы на монтажном поле платы. Выводы микроконтроллера, используемые в собственных проектах, отсоединяются с помощью легко удаляемых перемычек. Программирование памяти микроконтроллера K1886BE61Y осуществляется с помощью внутрисхемного программатора.

Для демонстрации функционирования плата Eval17 подключается к COM порту ПК или к интерфейсу RS-232 дополнительного внешнего устройства, например к аналогичной демонстрационно-отладочной плате Eval17. Подключение производится с помощью прилагаемого нуль-модемного кабеля. Питание платы осуществляется от адаптера стабилизированного напряжения 5 В или от адаптера 12 В.

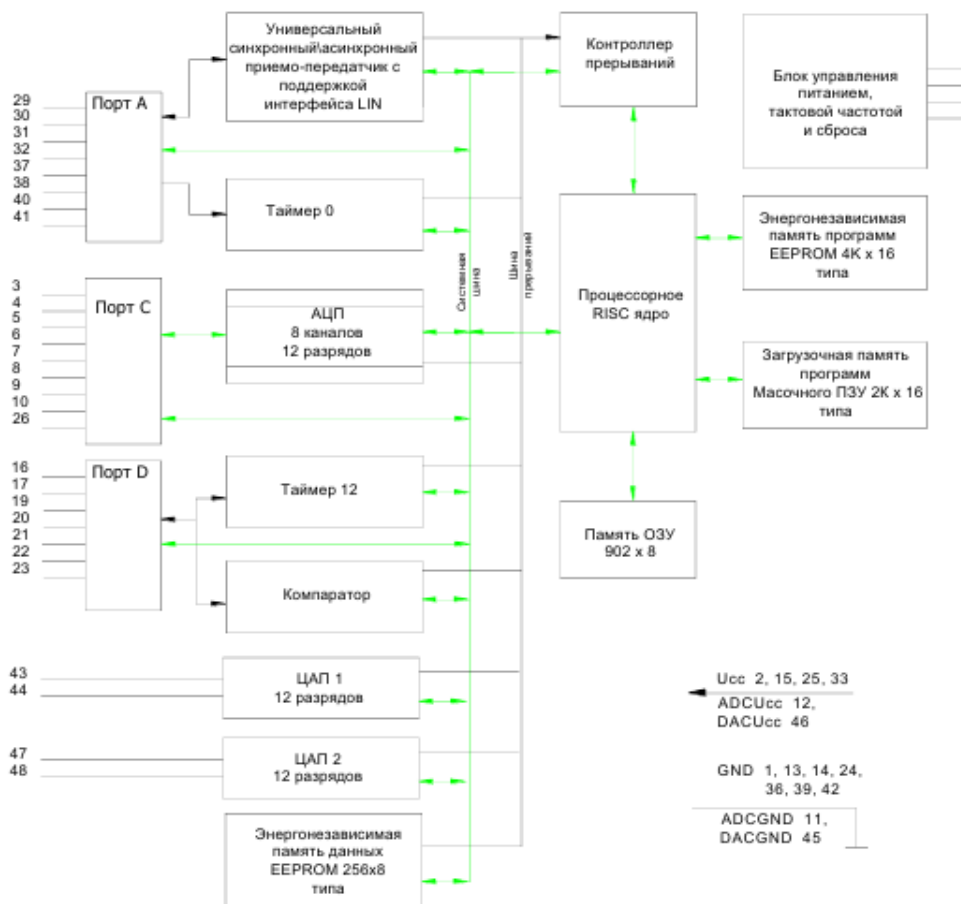
Для демонстрации функционирования используется прилагаемое программное обеспечение: демонстрационная программа, прошиваемая в память микроконтроллера K1886BE61Y, и демонстрационная программа для ПК.

На демонстрационно-отладочной плате Eval17 установлены следующие блоки и компоненты:





## Схематичное устройство платы:



## Порядок выполнения работы:

### Задание 1. Загрузить тестовую программу на микроконтроллер.

Выполнить этап трансляции описания, получив файлы .bin, .hex. Убедиться, что на индикаторе, размещенном на плате, высветились значения каналов. Проверить все каналы.

#### 1. Создание проекта и его структуры.

Запустите приложение DEVCPP.EXE от имени администратора (ПКМ – Запуск от имени администратора). Создайте новый проект: File – New – Project. В качестве компилятора выберите MPASM. Укажите имя проекта и место его хранения.


*Примечание.* Длина адреса не должна превышать 62 символов, адрес должен состоять только из латинских букв.

Скопируйте в папку, где хранится проект, компилятор MPASMWIN.EXE и заголовочный файл 1886VE6M.INC.

Откройте Project – Project Options – Compilers и в первом списке выберите MPASM. Выберите генерацию инструкций для микроконтроллера 1886BE6.

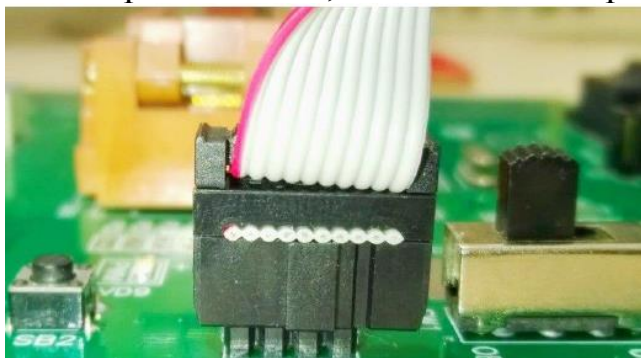
Закройте окно Project Options нажатием ОК.

Откройте Tools – Compiler Options. Во второй строке выберите адрес хранения компилятора MPASMWIN.EXE. Закройте окно Compiler Options нажатием ОК.

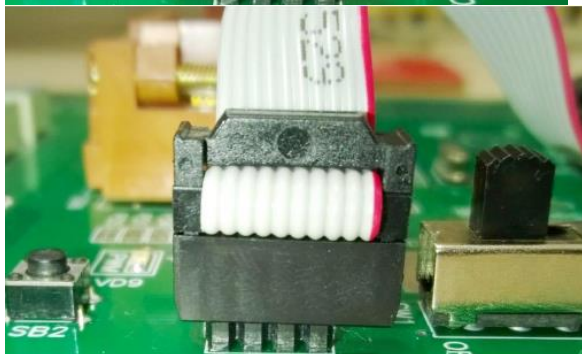
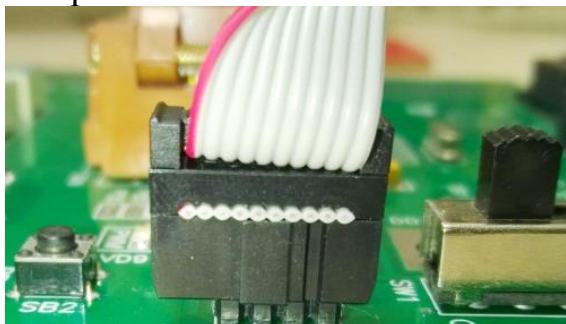
2. Подготовка файла на языке Assembler. Для подготовки файла на языке Assembler скопируйте в окно проекта текст из тестовой программы, находящейся по адресу: D:\LW\IDE1886\Examples\Eval17\_VE6\Project1.asm
3. Компиляция. Проект может быть скомпилирован через меню Execute – Compile или с помощью сочетания клавиш Ctrl+F9 либо через значок Compile  на панели быстрого доступа. В директории проекта должны появиться файлы с расширениями .bin и .hex. При возникновении ошибки «все права защищены» следует проверить директории хранения компиляторов.
4. Подключение программатора к плате.

**Внимание!** Переключатель SW1 должен находиться в положении «OFF» до того, как USB-программатор будет подключен, в противном случае возможен выход микросхемы из строя.

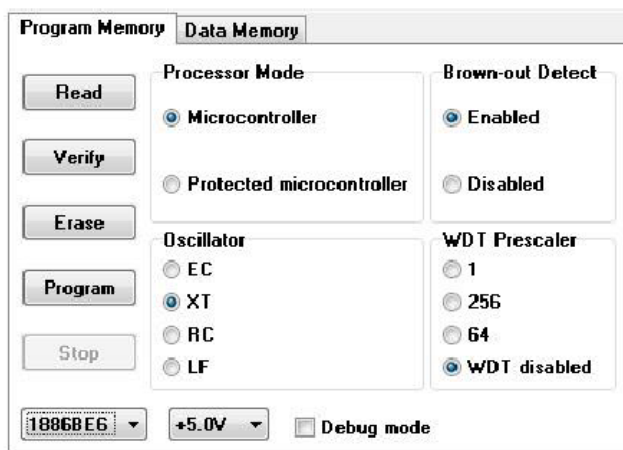
Подключите плату к ПК с помощью USB-программатора. При этом красный (розовый) провод на шине USB-программатора должен соединиться с выводом №1 разъема XP1, как показано на рисунке.



**Обратите внимание!** На рисунках ниже показаны примеры некорректного подключения разъема программатора: на 1 рисунке красный провод «висит» слева от разъема, а на 2 рисунке красный провод «висит» справа от разъема.



5. Загрузка программы в микроконтроллер. Перейдите с основного окна проекта на вкладку Programming и далее – на вкладку Program Memory.



В окне Program Memory:

- в разделе Processor Mode выберите Microcontroller;
- в разделе Oscillator выберите XT;
- в разделе Brown-out Detect выберите Enabled;
- в разделе WDT Prescaler выберите WDT disabled.
- установите напряжение +5.0V;
- в списке микроконтроллеров выберите 1886BE6.

Далее необходимо загрузить программу в память микроконтроллера.

Для успешной загрузки любой программы на микроконтроллер K1886BE61Y сначала необходимо стереть прежние инструкции, записанные в его память, нажав клавишу Erase, затем загрузить новые инструкции нажатием клавиши Program.

После того как новая программа будет успешно загружена в память микроконтроллера, следует отключить USB-программатор.

Подключите питание 5 В на вход XP2. Переключатель 3 на микросхеме поставьте в положение «ON».

6. Получение результата. На индикаторе, размещенном на плате, должны высветиться значения каналов, а именно АЦП2, АЦП3, АЦП4, АЦП5 и четыре свободных канала. Подбрав нужные значения подстроечных резисторов R22 и R24, получите на выходе АЦП2 и АЦП3 значения 700 и 900 соответственно. С помощью клавиатуры введите на вход ШИМ1 (PWM1) значение 500, на вход ШИМ2(PWM2) – 600, на вход ЦАП1 (DAC1) – 700, на вход ЦАП2(DAC2) – 800.

### **Оформление отчета:**

- 1) формулировку цели работы;
- 2) конспект практической части;
- 3) анализ полученных результатов.

### **Контрольные вопросы:**

1. С какой целью производится стирание памяти микроконтроллера?
2. С какой целью производится верификация памяти микроконтроллера?
3. Что общего и какие различия у файлов с разрешением .bin и .hex?
4. Сохраняются ли в памяти микроконтроллера значения, введенные на каналы при перезапуске, при перезагрузке, при перепрограммировании?

### ЛАБОРАТОРНАЯ РАБОТА № 3.

#### Тема: Измерители влажности

**Цель работы:** получение сведений о параметрах воздушной среды; изучение устройства и принципа работы метеометра; приобретение практических навыков по определению параметров воздушной среды с помощью метеометра.

**Оборудование:** метеометр, инструкции по выполнению работы

#### Справочный материал:

Основными параметрами воздушной среды являются температура окружающей среды, относительная влажность воздуха, атмосферное давление и скорость воздушного потока.

Оптимальным диапазоном относительной влажности воздуха считается 30 – 60 %.

Прибор контроля параметров воздушной среды – метеометр – предназначен для измерения атмосферного давления, относительной влажности, температуры воздуха, скорости воздушного потока, которую можно измерять как на открытых пространствах, так и в вентиляционных трубопроводах.

Метеометр состоит из блока электроники и сменного измерительного щупа. Составные части метеометра предназначены для эксплуатации в следующих условиях:

- блок электроники при температуре от  $-20$  до  $60$  °С и относительной влажности окружающего воздуха до 95 % при температуре  $35$  °С;
- щуп измерительный Щ-1 для измерения давления, относительной влажности, температуры и скорости воздушного потока при температуре от  $-40$  до  $85$  °С и относительной влажности окружающего воздуха до 98 % при температуре  $35$  °С.

#### Порядок выполнения работы:

**Задание 1.** Ознакомиться с устройством и принципом работы метеометра

**Задание 2.** Настроить прибор и выполнить измерения параметров воздушной среды: температуры, влажности, давления и скорости воздушного потока. Результаты записать в таблицу

Параметры воздушной среды	Температура воздуха	Относительная влажность воздуха	Относительное атмосферное давление	Скорость воздушного потока
Фактические значения параметров				

**Задание 3.** Оценить погрешности результатов измерений.

**Задание 4.** Сделать выводы по работе.

#### Оформление отчета:

1. Наименование работы, цели.
2. Таблица результатов измерений.
3. Выводы.

**Контрольные вопросы:**

1. Перечислите основные параметры воздушной среды.
2. Какие значения параметров воздушной среды создают комфортные ощущения?
3. Какие внешние условия нужно соблюдать при использовании метеометра



## ЛАБОРАТОРНАЯ РАБОТА № 4.

### Тема: Измерители мощности

**Цель работы:** изучить методы измерения активной мощности и энергии в трехфазной электрической цепи

**Оборудование:** ваттметр, инструкции по выполнению работы

#### Справочный материал:

Активная мощность симметричной трехфазной электрической цепи (ЭЦ) определяется по формуле:  $P = 3 \cdot U_{\text{ф}} \cdot I_{\text{ф}} \cdot \cos\varphi = 3 \cdot U_{\text{Л}} \cdot I_{\text{Л}} \cdot \cos\varphi$ ,

где  $U_{\text{ф}}$  – фазные напряжения, В;

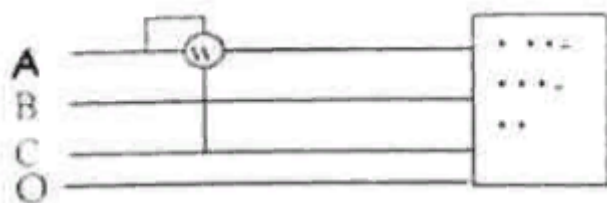
$I_{\text{ф}}$  – фазные токи, А;

$U_{\text{Л}}$  – линейное напряжение, В;

$I_{\text{Л}}$  – линейный ток, А;

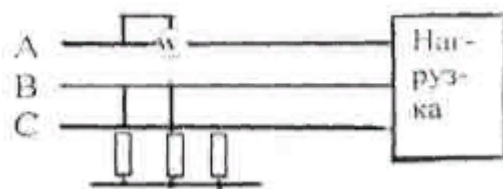
$\varphi$  – фазовый сдвиг между током и напряжением в фазах нагрузки.

В симметричной трехфазной 4-х проводной электрической цепи можно использовать один ваттметр, утроенное показание которого дает значение потребляемой мощности.

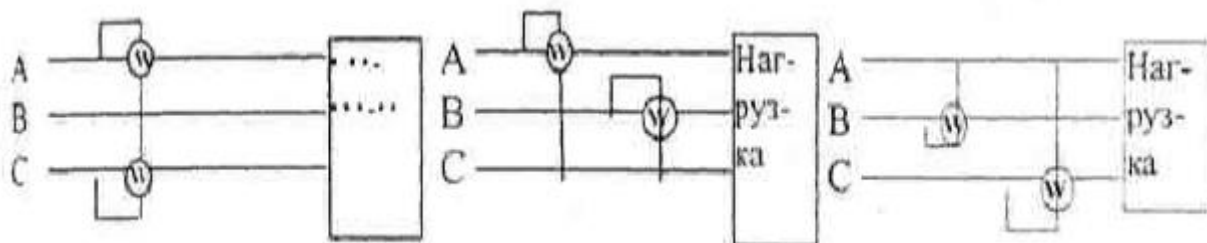


При несимметричной нагрузке применяют три ваттметра включенные аналогично во все три фазы.

В симметричной трехфазной 3-х проводной электрической цепи используется искусственная нулевая точка



При несимметричной трехфазной 3-х проводной электрической цепи можно использовать метод двух ваттметров.

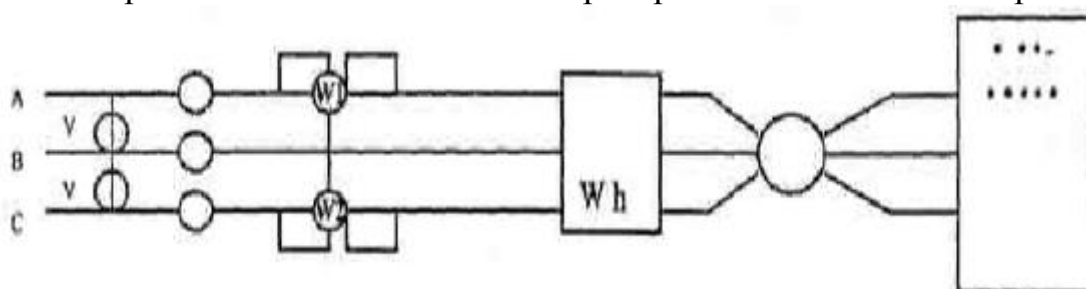


При этом активная мощность в симметричной и несимметричной трехфазной 3-х проводной цепи будет равна сумме показаний ваттметров.

#### Порядок выполнения работы:

**Задание 1.** Ознакомиться с приборами, используемыми в работе, записать их данные.

**Задание 2.** Собрать схему, приведенную на рисунке, обратив особое внимание на правильность включения генераторных зажимов ваттметров.



Измерительная схема

**Задание 3.** При симметричной активной нагрузке определить активную мощность

- 1) по формуле  $P = \sqrt{3} \cdot U \cdot I \cdot \cos\varphi$
- 2) по показаниям двух ваттметров;
- 3) по показаниям счетчика и секундомера;
- 4) по векторной диаграмме;
- 5) подсчитать погрешности мощности при всех способах измерений;
- 6) сравнить между собой результаты измерений всеми способами (составить таблицу).

**Задание 4.** Создать активно-индуктивную симметричную нагрузку (например, при  $\cos\varphi = 0,5$ ) и для нее:

- 1) определить активную мощность всеми способами, указанными в п.3;
- 2) по векторной диаграмме найти показания каждого ваттметра;
- 3) сравнить результаты измерения всеми способами и оценить их погрешности;
- 4) сравнить результаты измерений при  $\cos\varphi > 0,5$  и при  $\cos\varphi < 0,5$ .

**Задание 5.** Повторить п. 4 при симметричной активно-емкостной нагрузке.

**Задание 6.** Повторить измерения п. 4 при любом несимметричном режиме.

#### Оформление отчета:

- 1) формулировку цели работы;
- 2) конспект практической части;
- 3) анализ полученных результатов.

#### Контрольные вопросы:

1. Какие существуют способы измерения мощности?
2. Как можно измерить активную и реактивную мощность в 3-х фазных электрических цепях одним однофазным счетчиком?
3. Можно ли применить искусственную нулевую точку при измерении активной энергии и активной мощности одним однофазным счетчиком?
4. Каковы особенности включения ваттметров, однофазных и многофазных счетчиков?

## **ЛАБОРАТОРНАЯ РАБОТА № 5.**

**Тема: Измерители параметров электробезопасности, токовые клещи, кабель-тестеры, калибраторы портативные, мегаомметры и омметры, измерители шума и вибрации**

**Цель работы:** научиться работать с измерителями параметров электробезопасности, применять в работе токовые клещи, кабель-тестеры, калибраторы портативные, мегаомметры и омметры, измерители шума и вибрации

**Оборудование:** измерители параметров электробезопасности, токовые клещи, кабель-тестеры, калибраторы портативные, мегаомметры и омметры, измерители шума и вибрации, инструкции по выполнению работы

### **Справочный материал:**

Измерители параметров электробезопасности сочетают в себе функционал многих измерительных устройств вплоть до мегаомметров и поэтому могут быть использованы для определения целого спектра параметров, влияющих на безопасность электроустановок или сетей. Это такие показатели как напряжение или частота в сети, заземляющий контур, поляризация, коэффициент абсорбции, полное сопротивление цепи или заземления, ток короткого замыкания и другие. Поэтому при их помощи проводятся комплексные испытания изоляционных покрытий кабелей, защитных цепей, освещенности и других объектов в целях обнаружения и локализации дефектов и их последующего устранения.

Клещи токоизмерительные представляют собой прибор, основным назначением которого является измерение электрического тока без разрыва электрической цепи и нарушения ее функционирования. В соответствии с измеряемыми величинами электроизмерительные клещи делятся на амперметры, вольтметры, ваттметры, фазометры, ампервольтметры.

К самым распространенным относятся клещевые амперметры для измерения переменного тока, получившие название токоизмерительных клещей. С их помощью можно быстро измерить ток в проводнике, не разрывая и не отключая электрическую цепь. Электроизмерительные клещи могут применяться в электроустановках до 10000В.

Измеритель шума и вибрации ИШВ-1 представляет собой комбинированный прибор, предназначенный для измерения интенсивности шума, вибрации и анализа спектра. Он позволяет измерять: шум от 30 до 140 дБ относительно порогового значения Па в диапазоне частот 0..12500 Гц; виброскорости от 7 до 130 дБ относительно порогового значения м/с в диапазоне частот - 10...2800 Гц.

Измерение шума основано на принципе преобразования звуковых и механических колебаний исследуемых объектов в пропорциональные им электрические сигналы. Прибор питается от сети напряжением 220 В или от элементов типа А 373.

## Порядок выполнения работы:

**Задание 1.** Изучить конструкцию токовых клещей и произвести с их помощью измерения

1. В состав токовых клещей любой модификации входят следующие основные части: клещи-магнитопровод, переключатель диапазонов и функций, дисплей, выходные разъемы, кнопка фиксации измерений.

Переключатель может быть установлен в одно из положений режимов измерений:

1. DCV – постоянное напряжение;
2. ACV – переменное напряжение;
3. DCA – постоянный ток;
4. ACA – переменный ток;
5.  $\Omega$  - сопротивление
6. значок диода – проверка диода
7. значок сигнала – прозвонка с зуммером

Три входных разъема прибора имеют защиту от перегрузки. При подключении прибора черный провод щупов подсоединяется к разъему COM, а красный – к разъему V $\Omega$ . Третий разъем, обозначенный как EXT, применяется для подключения измерителя изоляции.

2. Порядок измерения тока. Переключатель пределов устанавливается в положение, соответствующее необходимому диапазону измерения переменного тока. Токовые клещи подключаются к измеряемому проводнику. Если на дисплее наблюдается только значение «1», то необходимо переключатель пределов установить на более высокое значение, так как возникла перегрузка.

3. Порядок измерения напряжения. Красный провод щупа подсоединить к разъему V $\Omega$ , а черный – к COM. Переключатель пределов установить в положение, соответствующее измеряемому диапазону. Щупы подсоединить к измеряемой нагрузке или источнику напряжения. На экране прибора будет наблюдаться измеряемое напряжение, а также его полярность. Если на экране наблюдается только значение «1», то переключатель пределов необходимо переключить на более высокое значение, т.к. возникла перегрузка.

4. Порядок измерения сопротивления. Щупы прибора так же, как и при измерении напряжения. Переключатель диапазонов установить на диапазон « $\Omega$ ». Если прибор используется для прозвонки, то переключатель нужно установить в соответствующее положение. Если сопротивление измеряемого участка схемы меньше 50 Ом, то будет звучать сигнал зуммера.



## Задание 2. Определить уровень шума и вибрации

1. Микрофон подключите к разъему «Вход», включите питание, установите его в положение «Контр». При этом индикаторная лампа начинает мигать, а стрелка индикатора устанавливается против сектора «Батарея». Тумблер ставится в положение «Звук». Переключатель «Род работы» поставьте в положение «Медленно», переключите «Род измерения» в положение «Лин».
2. При камере суммарного звукового давления во всех октавных полосах переключатель ставится в положение «Фильтр». Переключатели «Децibel 1» и «Децibel II» ставятся в крайнее правое положение (90 и 40).
3. Если при измерении стрелка индикатора прибора располагается в левой части шкалы, то она выводится в правую часть (правее 0) изменением положения переключателя «Децibel 1» и «Децibel II» при положении переключателя род измерения «Лин». Отсчет показаний производится суммированием показателей переключателей «Децibel 1» и «Децibel II» стрелочного прибора с учетом коэффициента поправки на чувствительность микрофона К (для данной настройки).
4. Отсчет при измерении шума в октавных полосах производится суммированием показателей «Децibel 1», «Децibel II» и стрелочного прибора. После проверки измерений измерительный прибор отключите. Измерения шума проводятся на лабораторной установке, состоящей из канала, в котором установлен источник шума. Микрофон устанавливается в гнездах канала на расстоянии 350, 1000 и 1600 мм от источника шума.
5. Методика определения спектрального и общего уровня звука. Для измерения шумов микрофон шумомера нужно ориентировать в направлении наибольшего воздействия шума. При измерении шума внутри помещений микрофон должен быть удален от пола, стен и источника шума не менее чем на 1,25 м. При измерении шума на открытом воздухе следует устанавливать микрофон не ниже 1,25 м от земли и не более 3,5 м от стен зданий и других крупногабаритных предметов. В случае измерения шума на рабочих местах операторов микрофон устанавливается на уровне их уха. При измерении шума внутри помещения, в которое шум проникает извне, должны быть открыты окна и двери. При измерении шума, создаваемого каким-либо устройством, необходимо, чтобы окружающий шум был, по крайней мере, на 10 дБ ниже уровня шума от данного устройства.
6. Определение изменения уровня шума в зависимости от расстояния до источника и вида звукоизолирующего материала:
  - Подготовить шумомер и стенд к измерениям, как было указано выше.
  - Провести измерение уровней шума в зависимости от расстояния до источника на стенде и занести результаты измерений в таблице 1

Источник шума	Расстояние до источника шума, м	Звонок
0,35	1,0	1,6

- Построить график изменения шума на расстоянии.
- Поместить на пути распространения шума поочередно экраны из дерева и пенопласта. Микрофон в этом случае находится на расстоянии 1,0 м от



источника шума. Замеры выполнить по октавным полосам на частотах. По результатам замеров построить графики в системе координат дБ – Гц и оценить изолирующие свойства дерева и пенопласта на разных среднегеометрических частотах октавных полос

### **Оформление отчета:**

- 1) формулировку цели работы;
- 2) конспект практической части;
- 3) анализ полученных результатов.

### **Контрольные вопросы:**

1. Что такое измерители параметров электробезопасности?
2. В каких единицах измеряется шум, и их физическая сущность?
3. Расскажите о принципе действия и отсчета показаний шумомера.
4. Назовите частотный диапазон звука. Что он показывает?

## ЛАБОРАТОРНАЯ РАБОТА № 6.

**Тема: Измерители параметров электробезопасности, токовые клещи, кабель-тестеры, калибраторы портативные, мегаомметры и омметры, измерители шума и вибрации**

**Цель работы:** научиться работать с измерителями параметров электробезопасности, применять в работе токовые клещи, кабель-тестеры, калибраторы портативные, мегаомметры и омметры, измерители шума и вибрации

**Оборудование:** измерители параметров электробезопасности, токовые клещи, кабель-тестеры, калибраторы портативные, мегаомметры и омметры, измерители шума и вибрации, инструкции по выполнению работы

### **Справочный материал:**

Мегаомметры — удобные и функциональные приборы для измерения сопротивления изоляции, позволяют не только выполнить точные замеры, но и убедиться в целостности изоляционного материала. Прибор позволяет замерять большие значения в сопротивлении цепей, изоляционных материалах, двигателях, телекоммуникационных установках и других видах техники, а основным назначением является определение безопасности эксплуатации проверяемых объектов.

### **Порядок выполнения работы:**

**Задание 1.** Изучить принцип работы мегаомметра.

1. Источником постоянного тока мегаомметра служит генератор постоянного тока напряжением до 2500В с ручным приводом. Мегаомметр имеет три зажима: Л – линия, З – земля, Э – экран.

Пояснение символов и знаков, нанесенных на мегаомметре:

- регулятор нуля;
- условное обозначение изменяемой величины;
- обозначение класса точности;
- прибор для использования с горизонтальным циферблатом;
- цепь постоянного тока;
- отрицательный зажим «ГХ»;
- испытательное напряжение 5,2 кВ;
- магнитоэлектрический прибор с подвижной катушкой и с электронным устройством в измерительной цепи
- оборудование, защищенное двойной или усиленной изоляцией;
- категория монтажа (категория перенапряжения) II;
- высокое напряжение;
- магнитная индукция 0,2 мТ;
- положения переключателя выходного напряжения ЭС0202/1-Г (ЭС0202/2-Г);
- товарный знак изготовителя;
- знак соответствия России;

Класс точности, выраженный в виде относительной погрешности по ГОСТ 8.401-80, 15. Пределы допускаемых значений основной относительной погрешности равны  $\pm 15\%$  от измеряемого значения.

Пределы допускаемых значений дополнительной погрешности мегомметров, вызванной протеканием в измерительной цепи токов промышленной частоты 50 мкА для ЭС0202/1-Г и 500 мкА для ЭС0202/2-Г, не должны превышать пределов основной относительной погрешности.

Время установления показаний не превышает 15 с.

Режим работы мегомметра прерывистый: измерение – мин, пауза – 2 мин.

Питание мегомметра осуществляется от встроенного электромеханического генератора.

Скорость вращения рукоятки генератора должна быть (120...144) оборотов в минуту.

Мегомметры сохраняют работоспособность при температуре окружающего воздуха от минус 30 до плюс 50 С и относительной влажности 90% при температуре плюс 30 С.

Рабочее положение – горизонтальное расположение плоскости шкалы.

2. Установить переключатель измерительных напряжений в нужное положение, а переключатель диапазонов в положение «1».

3. При вращении рукоятки генератора начинает светиться индикатор «ВН», что свидетельствует о наличии выходного напряжения на клеммах прибора.

4. Убедившись в отсутствии напряжения на объекте, подключите объект к гнездам «ГХ». При необходимости экранировки, для уменьшения влияния токов утечки, экран объекта подсоединить к гнезду «Э».

5. Для проведения измерений вращать рукоятку генератора со скоростью 120-144 оборотов в минуту.

6. После установления стрелочного указателя, сделайте отчет значения измеренного сопротивления. При необходимости перейдите на другой диапазон.

7. По окончании измерений установите переключатели мегомметра в среднее положение.

8. Провести замер сопротивления изоляции лабораторного оборудования.

## **Задание 2. Определить неисправности кабельных линий**

Измерение сопротивления изоляции жил кабеля. Выявление характера повреждения жил кабеля. Измерение сопротивления петли фаза—нуль. Анализ установленной защиты на надежность срабатывания при коротких замыканиях.

1. Для выполнения измерений кабель необходимо отключить от сети, проверить отсутствие напряжения и наложить заземление.

2. После измерения сопротивления изоляции нельзя касаться жил кабеля, так как кабель в качестве конденсатора может быть заряжен высоким напряжением от мегомметра. Поэтому после каждого измерения кабель должен быть разряжен с помощью штанги с заземленным проводником.

3. Измерение цепи фаза—нуль прибором М 417 выполнять только в присутствии и под наблюдением преподавателя. Прибор подключать при отключенном напряжении сети после проверки отсутствия напряжения.

### **Оформление отчета:**

- 1) формулировку цели работы;
- 2) конспект практической части;
- 3) анализ полученных результатов.

### **Контрольные вопросы:**

1. Для чего предназначен мегаомметр? Требования, предъявляемые к работе с мегаомметром.
2. В каких случаях запрещено пользоваться мегаомметром?
3. В каких единицах измеряется сопротивление изоляции.
4. Назначение клеммы Э – экран.
5. Какой основной элемент мегаомметра?

## ЛАБОРАТОРНАЯ РАБОТА № 7.

### Тема: Подбор контроллера

**Цель работы:** научиться производить подбор контроллеров для схемы RepRap 3D-принтера

**Оборудование:** 3D-принтер, контроллеры для схемы RepRap, инструкции по выполнению работы

#### **Справочный материал:**

Плата контроллера является сердцем любого 3D-принтера и отвечает за управление такими компонентами, как экструдер, нагретый слой, двигатели и датчики. Быстро развивающиеся платы контроллеров 3D-принтеров расширяются, предлагая невероятные возможности.

#### **Порядок выполнения работы:**

**Задание 1.** Рассмотреть имеющиеся контроллеры для 3D-принтера

##### 1. Плата STEVAL-3DP001V1 для управления 3D-принтером

STEVAL-3DP001V1 представляет собой полностью завершенное комплексное решение производства компании STMicroelectronics, поставляемое в виде платы с поддержкой необходимого программного обеспечения. STEVAL-3DP001V1 позволяет управлять разными моделями современных 3D-принтеров, включая требующие повышенной вычислительной мощности модели типа Delta. Эта система управления идеально подходит как для новичков, только открывающих для себя мир 3D-печати, так и для опытных специалистов. Созданная на основе 32-разрядного микроконтроллера на ядре ARM Cortex M4 автономная система управления может использоваться как с программным интерфейсом, так и с пользовательскими прошивками.

Плата STEVAL-3DP001V1, представленная на рисунке 1, разработана для управления многоосевыми 3D-принтерами (поддержка до 6 двигателей) с использованием 1...3 экструдеров и 1...3 зон нагрева стола.



Использование беспроводного интерфейса Wi-Fi позволяет управлять 3D-принтером с планшета или смартфона. При этом можно работать с такими распространенными программными инструментами для 3D-печати как Pronterface. Интерфейс USB доступен через виртуальный COM-порт, а также разъемы USB OTG и USB A.

Основные характеристики STEVAL-3DP001V1:

##### 1. Совместимость с новейшим поколением 3D-принтеров:

- поддержка нескольких экструдеров (до 3 шт.): 12 В/ до 8 А, контроль температуры и управление вентилятором охлаждения, LED-индикатор,
- упрощенное подключение разнообразной периферии,



- повышенная выходная мощность для высокотемпературной камеры и нескольких зон нагрева стола: 12 В или 24 В/ до 20 А, контроль температуры, LED-индикатор;

2.Использование полевых транзисторов (MOSFET) со сверхнизким сопротивлением открытого канала ( $R_{ds(on)}$ ), обеспечивающее повышенный выходной ток при незначительных размерах платы;

3.Программная прошивка с открытым исходным кодом;

4.Поддержка периферийных устройств:

- встроенные модули USB и micro SD,
- встроенный модуль Wi-Fi для связи с сетью и сервером,
- внешний-ЖК дисплей/ клавиатура;

5.Драйвер управления шаговыми двигателями ST SPIN L6474, обладающий уникальными характеристиками по выходному току и защите (до 6 осей, ток в фазе до 3 А, микрошаговый цифровой конечный ограничитель);

6.Встроенная система отладки STLINK/V2;

7.Соответствие требованиям RoHS по содержанию вредных и опасных веществ.

К плате можно подключить до трех зон подогрева с отдельным управлением и общим потребляемым током до 20 А. Для оперативной наладки и упрощенного контроля работы на плате предусмотрена светодиодная индикация. По свечению многочисленных светодиодов можно судить о рабочем состоянии платы и ее взаимодействии с компьютером и периферийным оборудованием.

Для подключения источников питания, шаговых двигателей, нагревателей и вентиляторов на плате предусмотрены винтовые зажимные клеммы. 40-контактный разъемный соединитель позволяет подключать концевые датчики по осям перемещения, сигналы обратной связи от термодатчиков, сигналы управления, интерфейсы. Напряжение питания составляет 5 В и 3,3 В.

## 2. Комплект TIDM-PRINTERPACK для управления 3D-принтером

Для непосредственного управления исполнительными механизмами 3D-принтера компания TI предлагает простое типовое решение в виде платы (Boosterpack, по терминологии компании) TIDM-PRINTERPACK, которая совместно с контроллерной платой MSP-EXP430F5529LP из семейства LaunchPad образует полноценную систему управления. Платы соединяются друг с другом через два 20-контактных разъема с гнездами на PRINTERPACK и длинными штырями на LaunchPad.

Особенностью этого предложения является то, что в действительности комплект TIDM-PRINTERPACK доступен в виде набора электронных компонентов (BOM- Bill of Materials) и свободно скачиваемых с сайта компании программного обеспечения и файлов для изготовления печатной платы.



Эта небольшая плата работает под управлением платы LaunchPad, которая создана на основе MSP430F5529 - 16-разрядного микроконтроллера со сверхнизким энергопотреблением, имеющего 128 кбайт Flash и 8 кбайт RAM-памяти, 12-разрядный АЦП и таймеры. Также имеется несколько интегрированных последовательных интерфейсов, включая порт USB.

Основные характеристики TIDM-PRINTERPACK:

- четыре драйвера DRV8825 для управления экструдером и шаговыми двигателями по осям X, Y и Z;
- защита двигателей от превышения тока;
- три N-канальных полевых транзистора CSD18534KCS для управления вентилятором, нагревательными элементами экструдера и зоны печати;
- протестированная совместимость с разнообразным программным обеспечением, включая Pronterface и Repetier;
- слот для карты microSD, позволяющий вести печать без подключения к компьютеру;
- протестированное решение для создания системы управления 3D-печатью со встроенным программным обеспечением, руководством пользователя и файлами для изготовления печатной платы.

Удобное расположение разъемов и готовое встроенное программное обеспечение позволят разработчику быстро создать недорогой 3D-принтер. Несмотря на то, что изначально плата создавалась как недорогой контроллер для управления 3D-принтером, наличие четырех драйверов шаговых двигателей на базе DRV8825 позволяет применить данное решение в любых системах управления шаговыми двигателями. В процессе тестирования TIDM-PRINTERPACK использовались запускаемые на компьютере программы для 3D-печати Pronterface и Repetier-Host.

### 3. Комплект TIDA-00405 для управления 3D-принтером

Другим типовым решением компании TI для непосредственного управления процессом 3D-печати является комплект TIDA-00405, представляющий собой, как и в случае TIDM-PRINTERPACK, плату BoosterPack. TIDA-00405, работающую под управлением материнской платы MSP430F5529 из семейства LaunchPad.



Плата TIDA-00405 также предназначена для управления работой экструдера и механизма перемещения по трем осям с использованием шаговых двигателей. Она лишь незначительно отличается от платы PRINTERPACK за счет использования менее мощных полевых транзисторов и драйверов шаговых

двигателей. Кроме того, если в TIDA-00405 предлагается использовать датчики Холла DRV5033 для контроля конечных позиций, то для платы PRINTERPACK на этот счет нет прямых указаний. Хотя, в обоих случаях датчики конечного положения не включены в перечень предлагаемых комплектующих.

В качестве драйверов осевых шаговых двигателей и экструдера здесь выбраны DRV8846 с пиковым выходным током 1,4 А. Три МОП-транзистора CSD18534Q5A с максимальным током 50 А использованы для контролируемого питания нагревателей стола и экструдера, а также вентилятора охлаждения.

Наиболее заметным отличием TIDA-00405 является наличие в комплекте готовой печатной платы устройства, тогда как в PRINTERPACK ее предлагается изготовить на основе имеющихся конструкторских файлов.

Основные характеристики TIDA-00405:

- Система управления, обеспечивающая управление шаговыми двигателями 3D-принтера, имеющая выходы для подключения нагревателей, входы для датчиков ограничения перемещения и слот для карт памяти SD;
- Система управления прецизионными шаговыми двигателями на основе драйверов DRV8846;
- Бесконтактные концевые выключатели на основе датчиков Холла, не подверженные механическому износу и работоспособные при загрязнении;
- Электропитание нагревателей через CSD18534Q5A с низким сопротивлением открытого канала  $R_{ds(ON)}$  7,8 мОм;
- Питание платы от одного источника 12 В;
- Система 3D-печати на основе TIDA-00405, прошедшая полный цикл испытаний в лабораторных условиях.

**Задание 2.** Подобрать контроллер, который подойдет к 3D-принтеру, находящемуся в кабинете

#### **Оформление отчета:**

- 1) формулировку цели работы;
- 2) конспект практической части;
- 3) анализ полученных результатов.

## **ЛАБОРАТОРНАЯ РАБОТА № 8.**

### **Тема: Программирование контроллера G-кодом**

**Цель работы:** научиться программировать контроллеры с помощью языка G-кода

**Оборудование:** ПК, контроллер, программное обеспечение для программирования G-кодом, инструкции по выполнению работы

#### **Справочный материал:**

G-code — условное наименование языка программирования устройств с числовым программным управлением (ЧПУ) и написанного на этом языке кода. Был создан компанией Electronic Industries Alliance в начале 1960-х. Программа, написанная с использованием G-code, имеет жесткую и последовательную структуру. В отличие от других языков программирования, в G-code нет циклов, функций и логических команд, т.е. оборудование выполняет ровно то, что подготовила программа, либо пользователь ручным набором.

Основные команды языка начинаются с буквы G (отсюда и его название), это например:

- перемещение рабочих органов оборудования с заданной скоростью (линейное и круговое,
- выполнение типовых команд (таких, как запуск вентиляторов, нагревателей и т.д.),
- управление параметрами, системами координат (абсолютное или относительное исчисление).

Для 3D-принтеров применяются программы, называемые слайсерами. Такие слайсеры, как Cura, Simplify3D и множество подобных, помогают автоматизировать процесс написания кода управления, пользователю остается лишь использовать полученный с их помощью код. Возможная ручная корректировка сводится к коррекции одного или двух параметров, или поиску лишней команды. Теперь не требуется писать 100-200 строк кода, достаточно лишь сгенерировать код в слайсере и, если требуется, внести небольшие правки

Для ручного внесения правок в G-code рекомендуем использовать бесплатную программу Notepad++. С её помощью можно найти вредоносную команду, пометить все её повторения в коде и удалить их.

#### **Порядок выполнения работы:**

**Задание 1.** Рассмотреть типичную команду G-кода

G1 X-9.2 Y-5.42 Z0.5 E0.0377

В этой кодировке содержится следующая информация:

- G1 — перемещаться по прямой;
- Координата X — -9,2 мм;
- Координата Y — -5,42 мм;
- Координата Z — 0,5 мм;
- Экструзия — 0,0377 мм.

## **Задание 2. Изучить основные команды G-кода**

### **1.G28 — вернуться в исходное положение**

Эта команда сообщает 3D-принтеру о необходимости вернуться в нулевую точку. С этой команды начинается работа 3D-принтера, а также этой командой заканчивается печать. Печатающая головка перемещается в дальний угол печатной камеры, чтобы пользователь мог легко извлечь деталь.

Пример: G28.

### **2.G1 — линейное движение**

С этой команды начинается около 95% строк в файле для печати. Команда G1 задает и направление перемещения печатающей головки. В этой же строке может содержаться команда E, которая указывает, сколько филамента (в миллиметрах) необходимо протолкнуть в сопло. Также в строке можно указать команду F, которая задает скорость движения в миллиметрах в минуту.

Пример: G1 X30 E10 F1800 — протолкнуть 10 мм филамента в экструдер, пока печатающая головка перемещается на 30 мм по координате X со скоростью 1800 мм/мин.

### **3.G92 — установить текущее положение**

Команда задает текущее положение осей. Одно из наиболее распространенных применений команды — это ось E (положение филамента). Если переопределить текущее положение нити, то все будущие команды будут определяться по новому значению. Обычно это делается в начале каждого слоя.

Пример: G92 E0 — установить текущее положение нити в качестве нулевого.

### **4.M104 и M109 — температура экструдера**

Команды M104 и M109 задают температурные значения в градусах Цельсия (S) для экструдера (экструдеров). При использовании команды M104 3D-принтер может производить другие действия в процессе нагрева. Команда M109 указывает принтеру не предпринимать других действий, пока не будет достигнута заданная температура. При использовании 3D-принтера с двумя экструдерами используются команды T0 для установления температуры правого экструдера и T1 — для левого.

Пример: M104 S190 T0 — начать разогревать правый экструдер до температуры 190 °C.

### **5.M140 и M190 — температура рабочего стола**

Эти команды указывают на необходимость нагреть рабочий стол до заданной температуры в градусах Цельсия (S). Аналогично примеру выше, команда M140 будет выполняться 3D-принтером одновременно с другими процессами, а команда M190 указывает на необходимость ожидания, пока рабочий стол не будет нагрет до заданной температуры.

Пример: M140 S50 — разогревать рабочий стол до 50 °C.

### **6.M106 — скорость вращения кулера**

Эта команда задает скорость вращения кулера, который охлаждает изделие. Скорость вращения (S) устанавливается в диапазоне значений от 0 (выключен) до 255 (максимальная скорость).



### Задание 3. Рассмотреть пример G-кода:

#### Задание 4. Написать программу для контроллера на языке G-кода

### Оформление отчета:

- 1) формулировку цели работы;
- 2) код программы;
- 3) анализ полученных результатов.

## ЛАБОРАТОРНАЯ РАБОТА № 9.

### Тема: Настройка в программном обеспечении Marlin

**Цель работы:** научиться настраивать программное обеспечение Marlin и использовать в 3D-принтере.

**Оборудование:** ПК, 3D-принтер, программное обеспечение Marlin, инструкции по выполнению работы

#### Справочный материал:

Данная прошивка является одной из самых популярных, в том числе, потому что разработчики регулярно добавляют в нее новые возможности: автоматическая регулировка зазора, датчик окончания прутка и многое другое. Кроме того, эта прошивка абсолютно бесплатная, и ее можно скачать с официального сайта.

#### Порядок выполнения работы:

##### Задание 1. Установка Arduino IDE

1. После того, как вы скачали прошивку, нужно ее отредактировать и в дальнейшем записать на микроконтроллер платы управления (Arduino mega 2560 ). Для этих целей понадобится программа Arduino IDE, скачать которую можно бесплатно с официального сайта Arduino.

2. Данная программа Arduino IDE регулярно обновляется и возможен такой вариант, что при заливки прошивки на плату, с новыми версиями Arduino IDE могут возникнуть проблемы, а именно будут появляться ошибки, и вы не сможете записать прошивку в микроконтроллер. Поэтому, при возникновении проблем, попробуйте скачать более старую версию программы, например версию 1.6.0

Arduino 1.6.x, 1.5.x BETA

These packages are no longer supported by the development team.

1.6.13	 Windows Windows Installer	MAC OS X	Linux 32 Bit Linux 64 Bit Linux ARM	Source code on Github
1.6.12	Windows Windows Installer	MAC OS X	Linux 32 Bit Linux 64 Bit Linux ARM	Source code on Github
1.6.11	Windows Windows Installer	MAC OS X	Linux 32 Bit Linux 64 Bit Linux ARM	Source code on Github
1.6.10	Windows Windows Installer	MAC OS X	Linux 32 Bit Linux 64 Bit Linux ARM	Source code on Github
1.6.9	Windows Windows Installer	MAC OS X	Linux 32 Bit Linux 64 Bit Linux ARM	Source code on Github

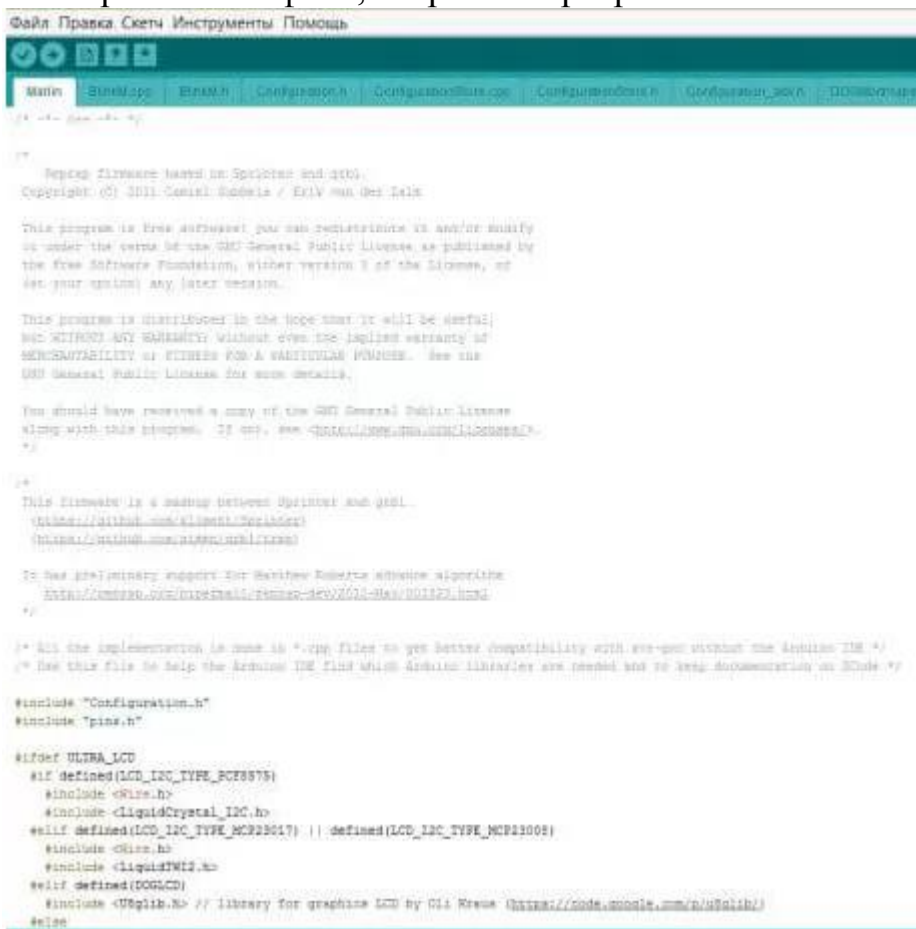
3. Нажмите на Windows Installer, и вас перекинут на другую страницу, где необходимо нажать на кнопку JUST DOWNLOAD, далее начнется скачивание файла. Установите программу и приступите к следующему шагу.

##### Задание 2. Отредактировать прошивку Marlin

1.Откройте папку с прошивкой "Marlin", найдите файл "Marlin" с расширением .ino

Имя	Дата изменения	Тип	Размер
example_configurations	01.12.2016 8:35	Папка с файлами	
scripts	01.12.2016 8:35	Папка с файлами	
Marlin	01.12.2016 8:35	Arduino file	2 КБ
BlinkM	01.12.2016 8:35	C/C++ Header File	1 КБ
boards	01.12.2016 8:35	C/C++ Header File	4 КБ
cardreader	01.12.2016 8:35	C/C++ Header File	4 КБ
Configuration	01.12.2016 8:35	C/C++ Header File	36 КБ
Configuration_adv	01.12.2016 8:35	C/C++ Header File	22 КБ
ConfigurationStore	01.12.2016 8:35	C/C++ Header File	1 КБ
Default_Version	01.12.2016 8:35	C/C++ Header File	1 КБ
dogm_font_data_marlin	01.12.2016 8:35	C/C++ Header File	16 КБ
dogm_lcd_implementation	01.12.2016 8:35	C/C++ Header File	20 КБ

2. Откройте этот файл, откроется программа Arduino IDE



```
/*
  RepRap firmware based on Sprinter and Arduino
  Copyright (C) 2011 Conall Madden / EriV on the left

  This program is free software: you can redistribute it and/or modify
  it under the terms of the GNU General Public License as published by
  the Free Software Foundation, either version 3 of the License, or
  at your option) any later version.

  This program is distributed in the hope that it will be useful,
  but WITHOUT ANY WARRANTY without even the implied warranty of
  MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
  GNU General Public License for more details.

  You should have received a copy of the GNU General Public License
  along with this program. If not, see <http://www.gnu.org/licenses/>.
  */

/*
  This firmware is a mashup between Sprinter and g201...
  (http://github.com/kluge/RepRap)
  (http://github.com/kluge/RepRap)


  To see preliminary support for Marlin's firmware algorithm
  (http://code.google.com/p/repap-firmware-eriv/2012-05-01/2012-05-01)
  */

/* All the implementation is done in *.cpp files to get better compatibility with the Arduino IDE */
/* See this file to help the Arduino IDE find which Arduino libraries are needed and to keep documentation on board */

#include "Configuration.h"
#include "pins.h"

#ifdef ULTRA_LCD
  #if defined(LCD_I2C_TYPE_PCF8575)
    #include <Wire.h>
    #include <LiquidCrystal_I2C.h>
  #elif defined(LCD_I2C_TYPE_MCP23017) || defined(LCD_I2C_TYPE_MCP23008)
    #include <Wire.h>
    #include <LiquidCrystal.h>
  #elif defined(DOGLCD)
    #include <Uglib.h> // library for graphics LCD by Olaf Krawinkel (http://code.google.com/p/uglib/)
  #endif
#endif
```

3. Вверху окна программы находится много вкладок, в каждой из которых располагаются куски кода, от которых и зависит работа 3D принтера. Вам потребуется только несколько основных вкладок. Первая и основная вкладка это "Configuration.h"

The image is a screenshot of the Marlin IDE interface. At the top, there is a menu bar with 'Файл', 'Правка', 'Скетч', 'Инструменты', and 'Помощь'. Below the menu bar is a toolbar with icons for saving, undo, redo, and other functions. A tab bar shows several open files: 'Marlin', 'BlinkM.cpp', 'BlinkM.h', 'Configuration.h' (which is the active file), 'ConfigurationStore.cpp', 'ConfigurationStore.h', 'Configuration\_adv.h', and 'DOOMBanner.h'. The main editor area displays the contents of 'Configuration.h'. The code is in C++ and includes various preprocessor directives and comments. It starts with '#ifndef CONFIGURATION\_H' and '#define CONFIGURATION\_H'. It includes 'boards.h'. There are comments explaining the purpose of the file and where to find advanced settings. It also includes sections for 'DELTA Printer' and 'SCARA Printer' configurations, with comments about replacing files in specific directories. At the bottom, there is a section for version information, including '#ifdef USE\_AUTOMATIC\_VERSIONING' and '#define STRING\_SPLASH\_LINE1 SHORT\_BUILD\_VERSION'.

```
#ifndef CONFIGURATION_H
#define CONFIGURATION_H

#include "boards.h"

// This configuration file contains the basic settings.
// Advanced settings can be found in Configuration_adv.h
// BASIC SETTINGS: select your board type, temperature sensor type, axis scaling, and endstop configuration

//=====
//----- DELTA Printer -----
//=====
// For a Delta printer replace the configuration files with the files in the
// example_configurations/delta directory.
//

//=====
//----- SCARA Printer -----
//=====
// For a Delta printer replace the configuration files with the files in the
// example_configurations/SCARA directory.
//

// @section info

#ifdef USE_AUTOMATIC_VERSIONING
#include "_Version.h"
#else
#include "Default_Version.h"
#endif

// User-specified version info of this build to display in [Pronterface, etc] terminal window during
// startup. Implementation of an idea by Prof Braino to inform user that any changes made to this
// build by the user have been successfully uploaded into firmware.
#define STRING_SPLASH_LINE1 SHORT_BUILD_VERSION // will be shown during bootup in line 1
//#define STRING_SPLASH_LINE2 STRING_DISTRIBUTION_DATE // will be shown during bootup in line 2
```

Этот конфигурационный файл, который содержит основные настройки. Именно в этой вкладке необходимо произвести основные изменения.

Все изменения в прошивке проведите по порядку сверху вниз. Эти изменения затронут основные участки кода, и они необходимы для начального запуска вашего 3D принтера.

### Задание 3. Устанавливаем необходимую скорость в бодах

1. Первое, что необходимо поменять - скорость в бодах. По умолчанию скорость стоит 250000 (47 строчка кода)

Marlin	BlinkM.cpp	BlinkM.h	Configuration.h	ConfigurationStore.cpp	ConfigurationStore.h	Configuration_adv.h
--------	------------	----------	-----------------	------------------------	----------------------	---------------------

```

// User-specified version info of this build to display in [Pronterface, etc] terminal window during
// startup. Implementation of an idea by Prof Braino to inform user that any changes made to this
// build by the user have been successfully uploaded into firmware.
#define STRING_SPLASH_LINE1 SHORT_BUILD_VERSION // will be shown during bootup in line 1
// #define STRING_SPLASH_LINE2 STRING_DISTRIBUTION_DATE // will be shown during bootup in line 2

#define STRING_VERSION_CONFIG_H __DATE__ " " __TIME__ // build date and time
#define STRING_CONFIG_H_AUTHOR "(none, default config)" // Who made the changes.

// SERIAL_PORT selects which serial port should be used for communication with the host.
// This allows the connection of wireless adapters (for instance) to non-default port pins.
// Serial port 0 is still used by the Arduino bootloader regardless of this setting.
#define SERIAL_PORT 0

// This determines the communication speed of the printer
#define BAUDRATE 250000

// This enables the serial port associated to the Bluetooth interface
// #define BTENABLED // Enable BT interface on AT90USB devices

// The following define selects which electronics board you have.
// Please choose the name from boards.h that matches your setup
#ifndef MOTHERBOARD
#define MOTHERBOARD BOARD_ULTIMAKER
#endif

```

2. Для каждой платы производитель рекомендует свои скорости, поэтому для связки Arduino mega 2560 и Ramps 1.4 необходимо поставить 115200, то есть участок кода у нас должен принять следующий вид:

// This determines the communication speed of the printer #define BAUDRATE 115200

Если вы используете плату Gen V1.4, то скорость должна быть 250000.

#### Задание 4. Выбираем управляющую плату

1. После установки скорости в бодах, необходимо указать используемую плату управления (55 строчка кода).

```

#ifndef MOTHERBOARD #define MOTHERBOARD BOARD_ULTIMAKER #endif

// This enables the serial port associated to the Bluetooth interface
// #define BTENABLED // Enable BT interface on AT90USB devices

// The following define selects which electronics board you have.
// Please choose the name from boards.h that matches your setup
#ifndef MOTHERBOARD
#define MOTHERBOARD BOARD_ULTIMAKER
#endif

// Define this to set a custom name for your generic Mendel,
// #define CUSTOM_MENDEL_NAME "This Mendel"

// Define this to set a unique identifier for this printer, (Used by some programs to differentiate between machines)
// You can use an online service to generate a random UUID. (eg http://www.uuidgenerator.net/version4)
// #define MACHINE_UUID "00000000-0000-0000-0000-000000000000"

```

2. По умолчанию стоит плата 3D принтера Ultimaker - BOARD\_ULTIMAKER, поэтому необходимо поменять плату. Весь список плат находится во вкладке "BOARDS\_H"





Там предоставлен огромный список различных плат, но вам необходимы только следующие:

```
#define BOARD_RAMPS_13_EFB 33 // RAMPS 1.3 / 1.4 (Power outputs: Extruder, Fan, Bed)
```

```
#define BOARD_RAMPS_13_EEB 34 // RAMPS 1.3 / 1.4 (Power outputs: Extruder0, Extruder1, Bed)
```

```
#define BOARD_RAMPS_13_EFF 35 // RAMPS 1.3 / 1.4 (Power outputs: Extruder, Fan, Fan)
```

```
#define BOARD_RAMPS_13_EEF 36 // RAMPS 1.3 / 1.4 (Power outputs: Extruder0, Extruder1, Fan)
```

Эти платы относятся к Arduino mega 2560 и Ramps 1.4. В зависимости от модификации вашего 3D принтера, необходимо выбрать соответствующую плату. Например, стандартная связка 1 экструдер + обдув рабочей области + нагревательный стол соответствует плате BOARD\_RAMPS\_13\_EFB

3. Название платы необходимо скопировать и заменить на вкладке "Configuration.h", меняем следующие строчки:

```
// The following define selects which electronics board you have.  
// Please choose the name from boards.h that matches your setup  
#ifndef MOTHERBOARD  
#define MOTHERBOARD BOARD_RAMPS_13_EFB  
#endif
```

#### 4. Меняем название 3D принтера

При настройке обязательно придумайте название своему 3D принтеру и укажите это в прошивке. Название принтера отображается на его LCD дисплее, такая возможность точно предусмотрена на таком дисплее.

Найдите строчки: (59 строчка)

```
// #define CUSTOM_MENDEL_NAME "This Mendel"
```

перед #define стоят "//" - это означает, что данные строчки не используются в коде, а служат в качестве пояснений. Чтобы активировать данную строчку, необходимо раскомментировать строку, уберите // перед строчкой.

Измените название по умолчанию "This Mendel" на ваше название 3D принтера, например, "P3Steel". Получаем следующие:

```
// Define this to set a custom name for your generic Mendel,  
#define CUSTOM_MENDEL_NAME "P3Steel"
```

#### Задание 5. Выбираем датчик температуры стола и экструдера

1. Выше были указаны настройки прошивки для 1 экструдера и нагревательного стола, то есть в 3D принтере присутствуют два нагревательных элемента, температуры которых необходимо регулировать. Контроль температуры производится с помощью датчиков температуры - термисторов.

2. Существует большое количество различных термисторов с различными характеристиками, поэтому в прошивке необходимо указать какой именно термистор стоит у вас. Это нужно, чтобы в дальнейшем принтер показывал верную температуру. В прошивке найдите список поддерживаемых термисторов:

```
//// Temperature sensor settings: // -2 is thermocouple with MAX6675 (only for  
sensor 0) // -1 is thermocouple with AD595 // 0 is not used // 1 is 100k thermistor -  
best choice for EPCOS 100k (4.7k pullup) // 2 is 200k thermistor - ATC Semitec  
204GT-2 (4.7k pullup) // 3 is Mendel-parts thermistor (4.7k pullup) // 4 is 10k  
thermistor !! do not use it for a hotend. It gives bad resolution at high temp. !! // 5 is  
100K thermistor - ATC Semitec 104GT-2 (Used in ParCan & J-Head) (4.7k pullup) //  
6 is 100k EPCOS - Not as accurate as table 1 (created using a fluke thermocouple)  
(4.7k pullup) // 7 is 100k Honeywell thermistor 135-104LAG-J01 (4.7k pullup) // 71  
is 100k Honeywell thermistor 135-104LAF-J01 (4.7k pullup) // 8 is 100k 0603 SMD  
Vishay NTCS0603E3104FXT (4.7k pullup) // 9 is 100k GE Sensing AL03006-  
58.2K-97-G1 (4.7k pullup) // 10 is 100k RS thermistor 198-961 (4.7k pullup) // 11 is  
100k beta 3950 1% thermistor (4.7k pullup) // 12 is 100k 0603 SMD Vishay  
NTCS0603E3104FXT (4.7k pullup) (calibrated for Makibox hot bed) // 13 is 100k  
Hisens 3950 1% up to 300°C for hotend "Simple ONE " & "Hotend "All In ONE" //  
20 is the PT100 circuit found in the Ultimainboard V2.x // 60 is 100k Maker's Tool  
Works Kapton Bed Thermistor beta=3950 // // 1k ohm pullup tables - This is not  
normal, you would have to have changed out your 4.7k for 1k // (but gives greater  
accuracy and more stable PID) // 51 is 100k thermistor - EPCOS (1k pullup) // 52 is  
200k thermistor - ATC Semitec 204GT-2 (1k pullup) // 55 is 100k thermistor - ATC  
Semitec 104GT-2 (Used in ParCan & J-Head) (1k pullup) // // 1047 is Pt1000 with
```



4k7 pullup // 1010 is Pt1000 with 1k pullup (non standard) // 147 is Pt100 with 4k7 pullup // 110 is Pt100 with 1k pullup (non standard)

3.В списке найдите свой, запомните цифру слева. Как правило, многие используют китайский термистор 100 кОм, для него подходит термистор под номером "1".

// 1 is 100k thermistor - best choice for EPCOS 100k (4.7k pullup)

Внесите изменения в нужном месте (строчки 115-118)

```
#define TEMP_SENSOR_0 -1 #define TEMP_SENSOR_1 -1 #define  
TEMP_SENSOR_2 0 #define TEMP_SENSOR_BED 0
```

4.По умолчанию в прошивке активированы два первых термистора:

TEMP\_SENSOR\_0 - отвечает за термистор первого экструдера

TEMP\_SENSOR\_1 - отвечает за термистор второго экструдера

TEMP\_SENSOR\_BED - отвечает за термистор стола

5.Поменяйте строчки и получите следующее:

```
// 51 is 100k thermistor - EPCOS (1k pullup)  
// 52 is 200k thermistor - ATC Semitec 204GT-2 (1k pullup)  
// 55 is 100k thermistor - ATC Semitec 104GT-2 (Used in ParCan & J-Head) (1k pullup)  
//  
// 1047 is Pt1000 with 4k7 pullup  
// 1010 is Pt1000 with 1k pullup (non standard)  
// 147 is Pt100 with 4k7 pullup  
// 110 is Pt100 with 1k pullup (non standard)  
  
#define TEMP_SENSOR_0 1  
#define TEMP_SENSOR_1 0  
#define TEMP_SENSOR_2 0  
#define TEMP_SENSOR_BED 1  
  
// This makes temp sensor 1 a redundant sensor for sensor 0. If the temperatures difference between these sensors is too high the print will be aborted.  
// #define TEMP_SENSOR_1_AS_REDUNDANT  
#define MAX_REDUNDANT_TEMP_SENSOR_DIFF 10  
  
// Actual temperature must be close to target for this long before M109 returns success  
#define TEMP_RESIDENCY_TIME 10 // (seconds)  
#define TEMP_HYSTERESIS 3 // (degC) range of +/- temperatures considered "close" to the target one  
#define TEMP_WINDOW 1 // (degC) Window around target to start the residency timer x degC early.
```

TEMP\_SENSOR\_1 и TEMP\_SENSOR\_2 не используются, поэтому напротив них ставим "0" нули.

6. Ограничение максимальной температуры

Для ограничения максимальной температуры необходимы следующие строчки (140-143)

```
#define HEATER_0_MAXTEMP 275 #define HEATER_1_MAXTEMP 275 #define  
HEATER_2_MAXTEMP 275 #define BED_MAXTEMP 150
```

Числа стоящие справа, а именно 275 и 150 - это максимальные температуры экструдера и нагревательного стола соответственно.

Когда температура превышает максимальный Temp, ваш нагреватель будет выключен. Эта функция существует для того, чтобы защитить ваш экструдер от случайного перегрева. Если вы используете хотенд с тефлоном внутри, то рекомендуем ограничить температурой 260°.

7.Ограничение минимальной температуры

Также в прошивке по умолчанию стоит ограничение минимальной температуры экструдера в 170°. Это означает что, если температура экструдера будет ниже 170°, то двигатель экструдера не будет вращаться и пластик не будет подаваться. Защита от проталкивания не прогретого пластика (строчка 230).

```
#define EXTRUDE_MINTEMP 170
```

Если хотите отключить данную функцию, то перед строчкой поставьте "//"

## 8. Настройка концевых выключателей. Настройка логики работы концевиков

В первую очередь на что нужно обратить внимание - это какие концевики вы используете и какой у них принцип работы. В прошивке необходимо правильно указать логику работы концевиков. Найдите следующие строчки (301-306)

```
const bool X_MIN_ENDSTOP_INVERTING = true; // set to true to invert the logic
of the endstop. const bool Y_MIN_ENDSTOP_INVERTING = true; // set to true to
invert the logic of the endstop. const bool Z_MIN_ENDSTOP_INVERTING = true;
// set to true to invert the logic of the endstop. const bool
X_MAX_ENDSTOP_INVERTING = true; // set to true to invert the logic of the
endstop. const bool Y_MAX_ENDSTOP_INVERTING = true; // set to true to invert
the logic of the endstop. const bool Z_MAX_ENDSTOP_INVERTING = true; // set
to true to invert the logic of the endstop.
```

Если механические концевики, то при срабатывании цепь замыкается, напротив каждой строчки соответствующей оси поставьте значения "true". Если вы используете оптические концевики, то при срабатывании цепь размыкается, напротив каждой строчки соответствующей оси поставьте значения "false".

По умолчанию в прошивке напротив каждого концевика стоят значения "true", что соответствуют механическим концевикам.

После настройки работу концевиков можно проверить командой M119 в консоли. В ответ должен прийти текст:

```
x_min:      open      -      концевик      не      сработал;
x_min: TRIGGERED - концевик сработал.
```

## 9. Установка положения "HOME" - дом

В прошивке поддерживаются 3 пары концевиков: для каждой оси X, Y и Z по два концевика min и max. Как правило, ставятся концевики только для минимального положения каждой оси, а максимальное задается в прошивке.

Положение дом (начальное положение), будет находиться в минимальных положениях концевиков и это задается в прошивке: (строчки 337-339)

```
#define X_HOME_DIR -1 #define Y_HOME_DIR -1 #define Z_HOME_DIR -1
1=MAX, -1=MIN
```

## 10. Изменения направления вращения двигателей

При сборке 3D принтера, а именно при подключение шаговых двигателей к плате, возможна такая ситуация: когда вы все настроили и подключили, при нажатии "home" (дом), каретка одной из осей едет в другую сторону (не к концевикам), тогда необходимо перевернуть коннектор шагового двигателя на 180° или поменять значения в прошивке:

```
#define INVERT_X_DIR true // for Mendel set to false, for Orca set to true #define
INVERT_Y_DIR false // for Mendel set to true, for Orca set to false #define
INVERT_Z_DIR true // for Mendel set to false, for Orca set to true #define
INVERT_E0_DIR false // for direct drive extruder v9 set to true, for geared extruder
set to false #define INVERT_E1_DIR false // for direct drive extruder v9 set to true,
```

for geared extruder set to false #define INVERT\_E2\_DIR false // for direct drive extruder v9 set to true, for geared extruder set to false

Например, если каретка оси Y в другую сторону, то необходимо найти строчку

```
#define INVERT_Y_DIR false // for Mendel set to true, for Orca set to false
```

и поменять "false" на "true". И так с каждой осью и экструдером.

## 11. Установка габаритов перемещения

Чтобы 3D принтер определял рабочую область, необходимо указать ее размеры в прошивке: (строчки 345-350)

```
#define X_MAX_POS 205 #define X_MIN_POS 0 #define Y_MAX_POS 205  
#define Y_MIN_POS 0 #define Z_MAX_POS 200 #define Z_MIN_POS 0
```

Напротив каждой строчки укажите соответствующие габариты, по умолчанию рабочая область задана 205x205x200 мм

## 12. Настройка шагов перемещения по осям

Указание количества шагов шаговых двигателей - одна из главных настроек прошивки (строчка 490):

```
#define
```

```
DEFAULT_AXIS_STEPS_PER_UNIT {78.7402,78.7402,200.0*8/3,760*1.1} //  
default steps per unit for Ultimaker
```

```
#define DEFAULT_AXIS_STEPS_PER_UNIT {78.7402,78.7402,200.0*8/3,760*1.1} // default steps per unit for Ultimaker
```

X	Y	Z	E0
---	---	---	----

В скобках через запятую для каждой оси указывается количество шагов, который должен сделать шаговый двигатель, чтобы каретка проехала 1 мм. Откуда взять данные значения? Можно рассчитать или возьмите уже известные.

Расчет осей X и Y (ремни). По всем осям стоят шаговые двигатели 200 шагов на оборот, 16 микро-шагов на шаг (это устанавливается перемычками на плате).

По осям X и Y стоит приводной ремень GT2 с шагом 2 мм и шкивы с 20 зубьями. Получается:

$$(200*16)/(2.0*20)=80$$

Столько шагов должен сделать шаговый двигатель, чтобы ось X и Y проехала ровно 1 мм.

Если у вас зубчатый шкив Gt2 с шагом 2 мм и с количеством зубьев 20, то формула такая:

$$(200*16)/(2.0*16)=100$$

Расчет оси Z (ходовой винт). По оси Z могут стоять:

- Шпилька M8 с шагом резьбы 1,25 мм, тогда формула:  $200*16/1.25=2560$
- Шпилька M5 с шагом резьбы 0.8 мм, тогда формула:  $200*16/0.8=4000$
- Трапецеидальный винт диаметром 8 мм с шагом 1 мм и заходностью 1, тогда формула:  $200*16/1=3200$
- Трапецеидальный винт диаметром 8 мм с шагом 2 мм и заходностью 1, тогда формула:  $200*16/2=1600$

- Трапецеидальный винт диаметром 8 мм с шагом 2 мм и заходностью 4, тогда формула:  $200 \cdot 16/2 \cdot 4 = 400$

В Pruse i3 Steel используются шпильки M5, тогда получается число 4000.

### 13. Расчет экструдера

Настройка подачи экструдера зависит от коэффициента редукции и диаметра подающей шестерни. Количество шагов, который должен сделать шаговый двигатель экструдера, чтобы продавить пластик на 1 мм подбирается экспериментально после первой заливки прошивки в 3D принтер.

Открутите сопло и уменьшите ограничение минимальной температуры сопла до 5°:

```
#define EXTRUDE_MINTEMP 5
```

Теперь экструдер будет работать при холодном сопле. Не меняя настроек экструдера, нажмите прогнать пластик на 100 мм. Измерьте длину прутка прошедшего через экструдер линейкой или штангенциркулем.

Подбирая настройку экструдера добейтесь точной цифры на разумной длине прутка, например 200 мм. После настройки верните ограничения минимальной температуры:

```
#define EXTRUDE_MINTEMP 170
```

### 14. Ограничение максимальной скорости перемещения по осям

```
#define DEFAULT_MAX_FEEDRATE {500, 500, 5, 25} // (mm/sec)
```

По умолчанию стоят скорости 500, 500, 5, 25 мм/с на оси X, Y, Z и экструдер соответственно. Рекомендуем понизить скорость с 500 до 200.

Настройка ускорения перемещений по осям

Еще одной из важных настроек является задание ускорений для различных осей, так как из-за некорректной настройки этого момента часто бывают проблемы при печати, а именно смещение слоев по причине пропуска шагов двигателя. Если поставить слишком большие ускорения, то будут пропуски. По умолчанию в прошивке стоят следующие значения:

```
#define DEFAULT_MAX_ACCELERATION {9000,9000,100,10000} // X, Y, Z, E
maximum start speed for accelerated moves. E default values are good for Skeinforge
40+, for older versions raise them a lot. #define DEFAULT_ACCELERATION 3000
// X, Y, Z and E max acceleration in mm/s^2 for printing moves #define
DEFAULT_RETRACT_ACCELERATION 3000 // X, Y, Z and E max acceleration
in mm/s^2 for retracts
```

Для осей X и Y стоят ускорения 9000 мм/с<sup>2</sup> - это очень много.

Для первичной настройки установите не более 1000 и для DEFAULT\_ACCELERATION поставьте 1500, вместо 3000.

```
#define DEFAULT_AXIS_STEPS_PER_UNIT {100,100,4000,100} // default steps per unit for Ultimaker
#define DEFAULT_MAX_FEEDRATE {500, 500, 5, 25} // (mm/sec)
#define DEFAULT_MAX_ACCELERATION {1000,1000,100,10000} // X, Y, Z, E maximum start speed for accelerated moves. E default values are good for Skeinforge 40+, for older versions raise them a lot.
#define DEFAULT_ACCELERATION 1500 // X, Y, Z and E max acceleration in mm/s^2 for printing moves
#define DEFAULT_RETRACT_ACCELERATION 1500 // X, Y, Z and E max acceleration in mm/s^2 for retracts
```

### 15. Активация дисплея

Последнее, что остается сделать - это активировать нужный вам дисплей. Один из самых популярных дисплеев, это RepRapDiscount Smart Controller. Найдите и раскомментируйте следующие строчки:



```
#define ULTRA_LCD #define SDSUPPORT #define ULTIPANEL #define
REPRAP_DISCOUNT_SMART_CONTROLLER
```

Перед этими строчками, не должны стоять "//". Должно получиться следующее:

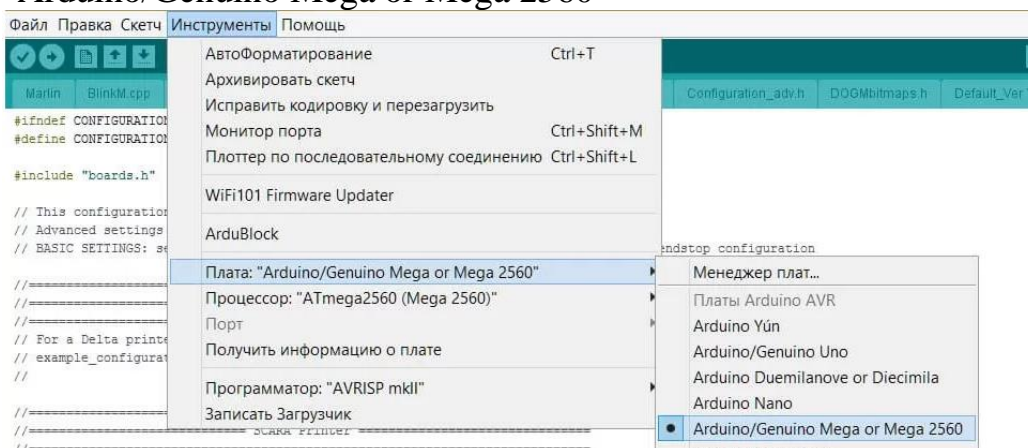
```
//LCD and SD support
#define ULTRA_LCD //general LCD support, also 16x2
// #define DOGLCD // Support for SPI LCD 128x64 (Controller ST7565R graphic Display Family)
#define SDSUPPORT // Enable SD Card Support in Hardware Console
// #define SDSLOW // Use slower SD transfer mode (not normally needed - uncomment if you're getting volume init error)
// #define SD_CHECK_AND_RETRY // Use CRC checks and retries on the SD communication
// #define ENCODER_PULSES_PER_STEP 1 // Increase if you have a high resolution encoder
// #define ENCODER_STEPS_PER_MENU_ITEM 5 // Set according to ENCODER_PULSES_PER_STEP or your liking
// #define ULTIMAKERCONTROLLER //as available from the Ultimaker online store.
#define ULTIPANEL //the UltiPanel as on Thingiverse
// #define LCD_FEEDBACK_FREQUENCY_HZ 1000 // this is the tone frequency the buzzer plays when on UI feedback. ie Screen Click
// #define LCD_FEEDBACK_FREQUENCY_DURATION_MS 100 // the duration the buzzer plays the UI feedback sound. ie Screen Click

// The MaKr3d MaKr-Panel with graphic controller and SD support
// http://reprap.org/wiki/MaKr3d_MaKrPanel
// #define MAKRPANEL

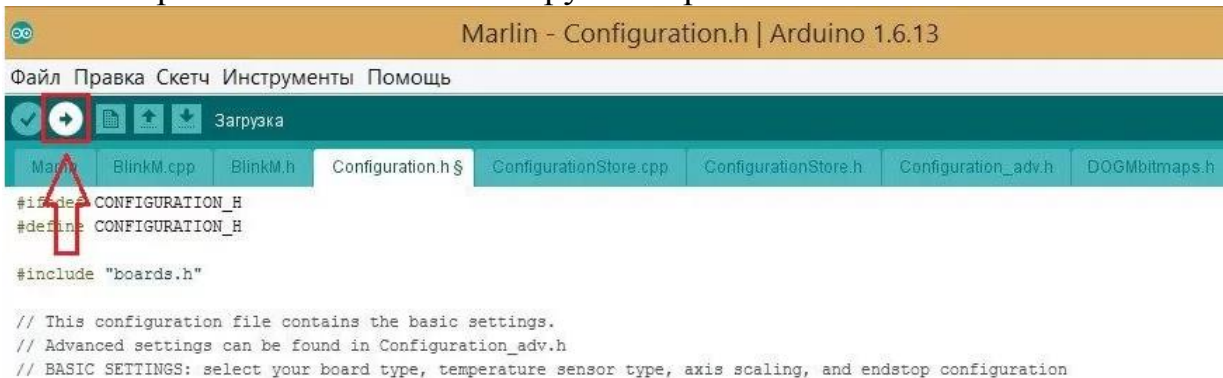
// The RepRapDiscount Smart Controller (white PCB)
// http://reprap.org/wiki/RepRapDiscount_Smart_Controller
#define REPRAP_DISCOUNT_SMART_CONTROLLER
```

## Задание 6. Заливка прошивки

1. После всех основных изменений прошивки, можно ее заливать. В программе Arduino IDE зайдите во вкладку "Инструменты" - "Плата" и выберите "Arduino/Genuino Mega or Mega 2560"



2. И там же нужно выставить верный COM порт вашего 3D принтера. Для заливки прошивки нажимаем на круг со стрелкой.



3. Прогресс заливки прошивки отображается индикатором, а после успешного завершения на экране появятся подтверждающие сообщение.

4. Далее можете пробовать запускать 3D-принтер

## ЛАБОРАТОРНАЯ РАБОТА № 10.

### Тема: Тестирование контроллера

**Цель работы:** изучить методику и порядок работы при тестировании контроллера для 3D-принтера

**Оборудование:** ПК, 3D-принтер, контроллер, инструкции по выполнению работы

### Порядок выполнения работы:

**Задание 1.** Проверка работоспособности контроллера

1. Протестировать контроллер. Результаты тестирования представить в виде таблицы:

№ п/п	Наименование пункта проверки	Значения проверяемого параметра	
		Должно быть	Полученное значение

### Оформление отчета:

- 1) Название работы
- 2) Цель работы
- 3) Перечень оборудования
- 4) Результаты тестирования в виде таблицы;
- 5) Вывод по работе.

## **Информационное обеспечение обучения по дисциплине**

### **Печатные издания**

#### **Основные учебные издания:**

1. Кравченко, Е. Г. Аддитивные технологии в машиностроении: учебное пособие для СПО / Е. Г. Кравченко, А. С. Верещагина, В. Ю. Верещагин. — Саратов: Профобразование, 2021. — 139 с. — ISBN 978-5-4488-1193-7. — Текст: электронный // Электронный ресурс цифровой образовательной среды СПО PROФобразование: [сайт]. — URL: <https://profspo.ru/books/105721>
2. Штейнбах, О. Л. Инженерная и компьютерная графика. AutoCAD: учебное пособие для СПО / О. Л. Штейнбах, О. В. Диль. — Саратов: Профобразование, 2021. — 131 с. — ISBN 978-5-4488-1175-3. — Текст: электронный // Электронный ресурс цифровой образовательной среды СПО PROФобразование: [сайт]. — URL: <https://profspo.ru/books/106615>

#### **Дополнительные учебные издания:**

3. Забелин, Л. Ю. Компьютерная графика и 3D-моделирование: учебное пособие для СПО / Л. Ю. Забелин, О. Л. Штейнбах, О. В. Диль. — Саратов: Профобразование, 2021. — 258 с. — ISBN 978-5-4488-1188-3. — Текст: электронный // Электронный ресурс цифровой образовательной среды СПО PROФобразование: [сайт]. — URL: <https://profspo.ru/books/106619>
4. Штейнбах, О. Л. Компьютерная графика. Проектирование в среде AutoCAD: учебное пособие для СПО / О. Л. Штейнбах, О. В. Диль. — Саратов: Профобразование, 2021. — 100 с. — ISBN 978-5-4488-1179-1. — Текст: электронный // Электронный ресурс цифровой образовательной среды СПО PROФобразование: [сайт]. — URL: <https://profspo.ru/books/106620>