

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ БЮДЖЕТНОЕ
УЧРЕЖДЕНИЕ
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«ДОНСКОЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»
ДГТУ

Кафедра «Приборостроение»

**ИНТЕГРИРОВАННАЯ СРЕДА ПРОЕКТИРОВАНИЯ УПРАВЛЯЮЩИХ
ПРОГРАММ ДЛЯ МИКРОКОНТРОЛЛЕРОВ PIC – MPLAB IDE**

Методические указания к лабораторной работе по дисциплинам
«Автоматизированные системы сбора и обработки измерительной
информации», «Микропроцессорная техника»

Ростов-на-Дону 2012

УДК 004.382.7

Составители: к.т.н., доц. А.В. Литвин

к.т.н., доц. К.А. Мороз

ст. преподаватель И.Н. Нестеренко

ст. преподаватель В.Н. Сыроватка

Формирование интервалов времени: Метод. указания к лабораторной работе по дисциплинам «Автоматизированные системы сбора и обработки измерительной информации» и «Микропроцессорная техника». – Ростов н/Д: Издательский центр ДГТУ. 2012. – 16 с.

В методических указаниях приводятся цель работы, краткое описание интегрированной среды проектирования MPLAB IDE, содержание и порядок выполнения лабораторной работы, приведен порядок создания проекта, приведены примеры использования возможностей по отладке встроенных в среду MPLAB IDE. Предназначены для студентов очной и заочной форм обучения специальностей «Приборостроение» и «Стандартизация и сертификация».

Печатается по решению методической комиссии факультета «Приборостроение и техническое регулирование»

Рецензент к.т.н., доцент П. С. Обухов

Научный редактор к.т.н., проф. В.Н. Ананченко

© Издательский центр ДГТУ, 2012

1. Цель работы. Изучение интегрированной среды проектирования MPLAB IDE v8.56. Получение практических навыков работы в среде на примере создания проекта устройства на микроконтроллере (МК) PIC16F628A.

2. Краткие сведения из теории.

MPLAB это набор программ, которые позволяют программисту максимально эффективно составлять, отлаживать и оптимизировать текст программы, а также, после завершения процесса отладки , создавать HEX - файл программы. (HEX – архивный файл, содержащий текст программы в машинном коде, предназначен для непосредственного программирования микроконтроллера).

Среда MPLAB создана только для микроконтроллеров PIC. Для микроконтроллеров других видов имеются свои среды проектирования. Работа по созданию текста программы или работа по изменению текста готовой программы всегда начинается с создания *проекта*. При этом создается файл проекта с расширением MCP. Изменять текст программы, производить ее оптимизацию и отладку лучше всего в проекте. В этом случае, можно пользоваться полным набором возможностей MPLAB.

Среда содержит специальный текстовый редактор для создания файла с текстом программы на ассемблере. Это часть программы MPASM входящей в пакет MPLAB. Также редактор осуществляет ряд сервисных функций по отладке программы. Файлы редактора имеют расширение ASM.

Совместно с текстовым редактором работает программа - симулятор. Она воссоздает работу устройства без применения реального микроконтроллера и является средством отладки программы. Этот "виртуальный", физически не существующий микроконтроллер, создается программными средствами MPLAB, конкретно для каждого типа PIC. С помощью симулятора можно проверить работу программы во всех возможных режимах (как в целом, так и по частям), отследить ее временные характеристики, найти ошибки и проверить их устранение. Программа - симулятор, покажет: по каким адресам находятся те или иные регистры и какие изменения происходят с их содержимым, посчитает количество машинных циклов на выбранном участке программы, симулирует ту или иную ситуацию и т.д.

3. Порядок выполнения работы

3.1. Создание каталога. Перед началом работы с программой необходимо создать каталог (папку), в котором будут размещаться

файлы проекта. Откройте папку PIC на диске C, и создайте в ней папку с именем primer1.

3.2. Настройка пакета MPLAB.

Запустите MPLAB. (Пуск/Программы/ Microchip/ MPLAB IDE v8.56/ MPLAB IDE);

Выберите тип микроконтроллера. Для этого в главном меню программы выберите пункт Configure, а в нем – пункт Select Device... Откроется окно Select Device, в котором в пункте Device необходимо выбрать тип МК PIC16F628A. Нажмите Ок.

Остальные настройки производятся по ходу работы программы.

3.3. Создание проекта.

В главном меню программы выберите пункт Project, а в нем – пункт Project Wizard. Откроется диалог Project Wizard по созданию проекта. Нажмите Далее.

На первом шаге предлагается выбрать тип микроконтроллера, на котором будет создаваться проект. Выберите МК PIC16F628A. Нажмите Далее.

На втором шаге предлагается выбрать язык программирования. Проверьте, чтобы в пункте Active Toolsuite было выбрано Microchip MPASM Toolsuite. В пункте Toolsuite Contents перечислен состав программ для поддержки данного языка. Их всего три: MPASM Assembler, MPLINK Object Linker, MPLIB Librarian. Каждая программа имеет свое место расположения, которое указано в строке Location. Если положение программы не указано, или ошибочно, то возле соответствующей программы в окне Toolsuite Contents появится красный крестик. В этом случае, необходимо сначала выбрать данную программу в окне Toolsuite Contents, затем нажать на кнопку Browse... и найти путь к программе. После того как путь будет корректным и не будет ни одного крестика, нажмите Далее.

На третьем шаге предлагается ввести имя проекта. Введите в поле Project Name имя primer1. В поле Project Directory необходимо ввести путь к папке, в которой будет сохранен данный проект. Для этого, нажмите Browse... и найдите папку primer1 в папке PIC на диске C. Щелкните на папке primer1. Нажмите Ок. Путь (C:\pic\primer1) окажется в строке Project Directory. Нажмите Далее.

- На четвертом шаге в проект можно добавить любой внешний файл из других проектов, тест программы и т.д. Сейчас у Вас нет таких файлов, поэтому нажмите Далее. В процессе лабораторной работы Вы научитесь добавлять внешние файлы на любом этапе проектирования.

В появившемся окне Вы видите итог настройки проекта, нажмите Готово. Проект создан. Закройте окно Output.

В процессе работы над проектом возможно изменение типа используемого микроконтроллера. Для этого необходимо в главном меню программы выбрать пункт Configure, а в нем – пункт Select Device... Откроется окно Select Device, в котором в пункте Device можно выбрать необходимый тип МК и нажать Ok.

Проект создан. Далее, необходимо создать файл, в котором будет находиться программа, написанная на языке ассемблера. Для это создайте новый файл (меню File/New). Данный файл необходимо сохранить (меню File/Save As...) по адресу C:\PIC\primer1\ с именем primer1. При этом, в строке 'Тип файла' должно быть указано Assembly Source Files (*.asm,...). При этом при сохранении к имени файла необходимо дописать расширение, а именно имя файла.asm. Теперь у созданного текстового файла есть имя и место положения. Однако этот файл пока не является файлом проекта. Чтобы он им стал, необходимо прикрепить его к проекту. Для этого, в окне проекта с заголовком primer1.mcs, выберите строку Source Files. Правой кнопкой мыши активируйте ниспадающее меню, в котором выберите пункт Add Files... Появится окно добавления файлов в проект (Add Files to Project), в котором найдите прежде сохраненный файл с именем primer1. Важно чтобы в строке Тип файлов был выбран тип (*.asm). Выделите файл с именем primer1, нажмите Открыть. Теперь файл находится в составе проекта, и в нем можно напечатать программу.

3.4. Загрузка в текстовый редактор текстов программ.

В случае если есть внешний файл с программой, то программу можно не набирать в файл, а скопировать её в файл проекта. Откройте заданный преподаваем внешний файл с программой (например, с помощью меню File/Open, программы MPLAB IDE). Файл имеет расширение asm. Далее, выделите все содержимое открытого внешнего файла, и скопируйте его в окно файла проекта (C:\PIC\primer1\primer1.asm). Закройте внешний файл. Теперь программа первого примера находится в проекте.

3.5. Настройка интерфейса MPLAB.

Далее, для обеспечения минимального удобства работы с программой настройте размер шрифта текста программы. (меню Edit/Properties..., вкладка Text). Нажмите на кнопку Select Font и установите необходимый размер шрифта текста программы. Перейдите на вкладку Editor пункта меню Edit/Properties..... Установите галочку напротив пункта Line Numbers. Теперь, каждая строка в тексте программы будет пронумерована, что существенно облегчит навигацию по программе в процессе отладки. Нажмите Ok.

3.6. Проверка и компиляция файла с программой.

Для проверки и компиляции выберите пункт меню Project/Build All. Если проверка синтаксиса программы прошла успешно, то полоса выполнения операции зеленого цвета. По окончании процедуры появляется окно результатов, в котором указываются диагностические сообщения (Message), и создается файл с архивом прошивки для непосредственного программирования микроконтроллера на программаторе. Файл находится в папке проекта и имеет расширение .hex. В случае если в процессе проверки были найдены ошибки, то полоса выполнения операции окрашивается в красный цвет, а в окне результатов помимо диагностических сообщений (Message), указываются и сообщения об ошибках (Error и Warning). Файл с архивом прошивки не создается, до момента исправления всех ошибок. Выполните проверку и компиляцию программы primer1 и исправьте ошибки.

При сообщениях об ошибках всегда указывается номер строки, в которой она найдена. Номер строки всегда можно узнать по параметру Ln нижней информационной полосы окна программы MPLAB. Также, можно перейти на строку с ошибкой щелкнув левой кнопкой мыши на сообщении об ошибке два раза.

3.7. Работа с симулятором.

Исправление всех синтаксических ошибок не гарантирует корректной работы программы согласно замыслу разработчика. Программу необходимо проверить на соответствие работы алгоритма заданию на проектирование. Для этой цели в составе пакета MPLAB имеется Симулятор. Это программа, которая воссоздает программно работу выбранного микроконтроллера согласно созданной разработчиком программы. Позволяет отслеживать весь процесс выполнения программы, состояние регистров, временные характеристики и т.д.

Для включения симулятора, выберите пункт меню Debugger/Select Tool/MPLAB SIM. При этом появляется панель запуска программы на симуляцию. (Квадратики с символами Run, перемотки и фигурные скобки со стрелками).

Далее, настройте симулятор на тактовую частоту используемую в проекте. В случае примера1 используется кварцевый резонатор на 4 Мгц. Откройте пункт меню Debugger/Settings... Выберите вкладку Osc/Trace и установите значение тактовой частоты генератора. Нажмите Ок.

После успешной проверки и компиляции текста программы на левом поле текста программы устанавливается мерцающий курсор с зеленой стрелкой напротив первой команды программы goto Start. Далее возможно несколько режимов работы симулятора.

Нажмите на кнопку Run (один цветной треугольник в панели запуска программы на симуляцию). Программа начинает выполняться в

непрерывном режиме с максимальной скоростью. О работе симулятора можно судить по бегущей полоске в нижней части окна MPLAB с названием Running... Остановите симуляцию, нажав на кнопку Halt – остановка (две вертикальные черты). Зеленая стрелка остановится на той команде программы, в момент выполнения которой была нажата кнопка остановки. Если повторно нажать кнопку Run работа программы начнется с этой команды. Остановите симуляцию.

Сбросьте текущее положение рабочей точки программы (зеленой стрелки) на начало программы (команда goto Start). Для этого нажмите на кнопку Reset (желтый квадрат со стрелкой).

Нажмите кнопку Animate (Анимация) с двумя цветными треугольниками. Начнется непрерывное выполнение программы в замедленном режиме, зеленая стрелка перемещается по программе. В этом режиме удобно отслеживать состояние регистров, поскольку изменение их содержимого происходит относительно медленно. Скорость анимации возможно менять с помощью пункта меню Debugger/Settings... вкладка Animation / Realtime Update , параметр Animate step time. Остановите симуляцию. Не нажимайте Reset.

Оставшиеся три кнопки предназначены для пошагового режима симуляции. Кнопка Step Into со стрелкой входящей в скобки запускает симулятор на выполнение одной команды. Нажмите на эту кнопку несколько раз, убедитесь, что рабочая точка программы (зеленая стрелка) перемещается.

Кнопка Step Over (выполнение подпрограммы с пропуском) со стрелкой обходящей скобки, предназначена для выполнения шага, но в случае если на пути рабочей точки появится команда call вызывающая подпрограмму, то подпрограмма будет выполнена и рабочая точка установится на команде следующей за командой call. (Визуального перехода в подпрограмму не будет).

Кнопка Step Out (выход из подпрограммы с выполнением) позволяет выйти из текущей подпрограммы на команду, следующую за командой её (подпрограмму) вызвавшей. При этом происходит ускоренное выполнение остатка команд выполнявшейся подпрограммы.

Поэкспериментируйте с кнопками Step Into, Step Over, Step Out, уясните смысл выполняемых ими действий.

На практике все режимы симуляции используются комплексно, подчас один режим сменяется другим, затем третьим и т.д.

3.8. Отслеживание состояния регистров в процессе работы симулятора.

Пакет MPLAB предоставляет широкий набор средств для отслеживания состояния регистров симулируемых микроконтроллеров. Данные программные функции находятся в меню View. Разработчик

может наблюдать за состоянием стека (View/Hardware Stack), где хранятся адреса возврата из подпрограмм. Также можно менять содержимое ячеек стека. (Двойной щелчок мышью на ячейке, и ячейка становится доступна для редактирования). Закройте окно Hardware Stack.

Выбрав строку View / Program Memory программист может наблюдать за состоянием памяти программ в трех различных видах: Orcode Hex – разкодированный файл из заархивированной прошивки (файлы с расширением .hex), видны ячейки памяти с числами, с шестнадцатеричным содержимым; Machine – программа в машинных кодах, в которой вместо символов указаны их числовые (адресные) значения; Symbolic – программа в машинных кодах, но с указанием символов переменных, меток и названий подпрограмм. Содержимое любой ячейки Program Memory можно редактировать. Закройте окно Program Memory.

Выбрав строку View / File Registers можно наблюдать за состоянием памяти данных, включая как регистры общего, так и регистры специального назначения. Наблюдение доступно как в режиме hex, так и в режиме Symbolic. Значение любого регистра возможно редактировать, причем значения представлены в шестнадцатеричном, двоичном и десятичном видах.

Представление в виде hex более компактно. В процессе выполнения программы регистр памяти данных, с содержимым которого производится действие выделяется красным цветом. Разместите на экране окно File Registers и окно с программой. Выберите вид hex. Сбросьте программу на начало. Запустите Симулятор в режиме анимации. Пронаблюдайте за изменением содержимого регистра счетчика команд PCL (адрес 02h), и как это содержимое отображается в других банках (по адресам 82h, 102h, 182h). Пронаблюдайте за изменением содержимого ячейки 70h, адрес которой присвоен переменной n. Остановите симуляцию. Сбросьте программу на начало. Как можно заметить, регистры памяти данных сохранили значения оставшиеся после последнего запуска. Поэтому, для обнуления памяти рекомендуется воспользоваться пунктом Debugger/ Clear Memory/ All Memory. После очистки памяти необходимо произвести проверку с компиляцией (Project/Build All). Закройте окно File Registers.

Для отдельного наблюдения регистров специального назначения можно воспользоваться пунктом меню View / Special Function Registers. Где собраны все регистры специального назначения и их значения, содержимое которых удобно изменять в случае необходимости. Закройте окно Special Function Registers.

На практике, использование всех перечисленных средств для отслеживания состояния регистров не всегда удобно в силу большой

численности регистров. Когда четко известно какие регистры необходимо отслеживать, пользуются пунктом меню View / Watch. Данная функция позволяет выбрать из состава регистров специального назначения выбрать необходимые. Выбор осуществляется из ниспадающего меню возле кнопки Add SFR, с последующим нажатием на эту кнопку. Таким же способом можно выбрать необходимые регистры общего назначения. Выбор осуществляется из ниспадающего меню возле кнопки Add Symbol, с последующим нажатием на эту кнопку. Выберите из регистров специального назначения регистры TrisB, PortB. Из регистров общего назначения – переменные p, p1, p2. Сбросьте программу на начало. Запустите анимацию, наблюдайте за изменением содержимого регистров. Остановите анимацию, сбросьте программу на начало.

3.9. Использование точек остановки.

При отладке больших программ, зачастую использование анимации или пошагового режима в масштабах всей программы не эффективно, в силу малой скорости операций. Использование перечисленных режимов дополняется использованием режима непрерывной симуляции. Однако этот режим необходимо остановить в определенной точке программы, с которой необходимо произвести отладку. Для этой цели в тексте программы устанавливают точки остановки. Для этого нужно указать щелчком мыши на команду в программе (укажите на строку 71 программы), а затем щелчком правой кнопки мыши в ниспадающем меню выбрать пункт Set Breakpoint (установить точку остановки). Появится красная точка на поле возле строки, отмеченной мигающим курсором. Аналогичный результат можно получить простым двойным щелчком левой кнопкой мыши. Убрать точку можно или двойным щелчком мыши, или командой ниспадающего меню Disable Breakpoint.

Установите точку остановки на строке 71. Запустите симулятор в режиме непрерывного выполнения (кнопка Run). Стрелка остановится на красной точке. Дальше возможен выбор пошагового или любого другого режима. При повторном нажатии кнопки Run программа выполнится по кругу и опять остановится в отмеченной точке. Поэкспериментируйте с точками остановки и режимами симуляции. Остановите симуляцию, удалите все точки. Сбросьте программу на начало.

3.10. Секундомер.

Для отслеживания временных характеристик выполнения программы в составе MPLAB имеется средство под названием Stopwatch расположенное в пункте меню Debugger. Запустите секундомер. Откроется окно Stopwatch. В окне представлена частота процессора, а также два параметра: Instruction Cycles – количество машинных циклов, а также Time – время выполнения программы

(команд). Каждый параметр разделен на два окна, собственно Stopwatch - секундомер, и общий счетчик машинных циклов и времени – Total Simulated.

Запустите симулятор в режиме анимации. Понаблюдайте как меняется счетчики. Остановите симулятор. Сбросьте программу на начало.

В программе primer1 отследите время выполнения подпрограммы Pause1. Для этого поставьте точки остановки напротив команды call Pause1 строка 111, и - команды decfsz n1,1 строка 112. Запустите симулятор в непрерывном режиме (кнопка Run). После чего, рабочая точка установится на первой точке останова. Обнулите секундомер (кнопка Zero). Повторно запустите симулятор в непрерывном режиме (кнопка Run). Рабочая точка установится на второй точке останова. В окне Stopwatch – Time прочтите значение времени выполнения подпрограммы. Поэкспериментируйте со значением переменной n. Убедитесь что число 199 обеспечивает необходимое время задержки в 1мс.

Удалите точки остановки. Аналогичным образом убедитесь, что интервал времени между переключениями сигналов на линиях порта B, строго соответствует значению в 1 секунду. (точки необходимо поставить, например на линиях 79 и 85, на которых расположены команды, изменяющие состояние линий порта B).

Завершите все эксперименты. Проверьте и откомпилируйте программу проекта primer1. Скопируйте файл-архив с прошивкой для программирования микроконтроллера на внешний носитель. (Файл находится в каталоге проекта, имеет расширение hex). Запрограммируйте микроконтроллер, убедитесь что программа выполняется согласно заданию.

Создайте новый проект, наберите и отладьте программу, реализующую задачу полученного согласно варианта задания. Произведите программирование контроллера, убедитесь в полном соответствии работы программы полученному заданию.

3.11. Повторите п.3.1 – 3.10 для случая программы, составленной согласно варианту индивидуального задания в лабораторной работе «Программирование микроконтроллера серии PIC16».

4. Содержание отчета

4.1. Цель работы.

4.2. Сведения о среде проектирования MPLAB IDE.

4.3. Описание порядка работы со средой проектирование MPLAB.

4.4. Текст подпрограммы по п.3.11 и результаты ее анализа.

4.5. *.hex файл прошивки для программирования микроконтроллера.

5. Контрольные вопросы для проверки готовности студентов к выполнению лабораторной работы.

5.1. Какие виды языков программирования Вам известны, опишите их особенности?

5.2. Как рассчитываются временные характеристики программ?

5.3. Из каких основных частей состоит программа микроконтроллера составленная на языке ассемблера?

5.4. Опишите основные правила написания программ на языке ассемблера?

5.5. Как задается содержимое слова конфигурации?

5.6. Перечислите известные Вам директивы ассемблера?

6. Контрольные вопросы по итогам лабораторной работы.

6.1. Опишите состав среды проектирования MPLAB IDE?

6.2. Опишите порядок создания проекта?

6.3. Как задать точки остановки в программе?

6.4. Какие режимы симулятора Вам известны, опишите их?

6.5. Файл с каким расширением содержит прошивку микроконтроллера?

6.6. Опишите порядок поиска и исправления синтаксических ошибок в программе?

6.7. Как проверить в симуляторе временные характеристики программы?

6.8. Опишите возможности симулятора MPLAB IDE в части наблюдения за изменением содержимого регистров в процессе симулирования выполнения программы?

7. ЛИТЕРАТУРА.

7.1. Сайт компании MICROCHIP - www.microchip.ru

7.2. Корабельников Е.А. Руководство по конструированию устройств на микроконтроллерах. 2006. www.ikarab.narod.ru

7.3. Предьяко М.Справочник по PIC-микроконтроллерам.— М.: ДМК Пресс, 2002.

7.4 Ульрих В.А. Микроконтроллеры PIC16X7XX. – СПб.: Наука и техника, 2002.

7.5. Катцен Сид PIC-микроконтроллеры. Полное руководство. – М.: Додэка-XXI, 2010.

7.6. Литвин А.В. и др. Методические указания к лабораторной работе «Регистры специального назначения». Ростов-на-Дону: ДГТУ, 2007.

Редактор А.А. Литвинова

ЛР N 020639 от 26.04.96. В набор . .11. В печать . .11.
Офсет. Объем 1.00 усл.п.л., уч. - изд.л. Формат 60x80/16
Бумага тип N3. Заказ N . Тираж . Цена

Издательский центр ДГТУ
Адрес университета и полиграфического предприятия:
344010, Ростов-на-Дону, пл. Гагарина, 1