

Министерство образования Республики Беларусь
Учреждение образования
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Кафедра радиотехнических систем

А.И. Бурак, В.Н. Левкович

**ИНТЕГРИРОВАННАЯ СРЕДА MPLab IDE РАЗРАБОТКИ ПРОГРАММ
ДЛЯ МИКРОКОНТРОЛЛЕРОВ PICmicro ФИРМЫ MICROCHIP**

Методическое пособие
к лабораторным работам по курсу
«Цифровые и микропроцессорные устройства»
для студентов специальностей
39 01 01 «Радиотехника» и 39 01 02 «Радиоэлектронные системы»
всех форм обучения

Минск 2002

УДК 681.325.5-181.48(075.8)

ББК 32.973.26-04 я 73

Б91

Б91 Интегрированная среда Mplab IDE разработки программ для микроконтроллеров PICmicro фирмы Microchip: Метод. пособие к лабораторным работам по дисциплине «Цифровые и микропроцессорные устройства» для студентов спец. 39 01 01 «Радиотехника» и 39 01 02 «Радиоэлектронные системы» всех форм обучения./ А.И. Бурак, В.Н. Левкович -Мн: БГУИР, 2002.- с.:31 ил. 10.

ISBN

Бурак А.И.

В методическом пособии рассмотрены основные функции и порядок работы в интегрированной среде Mplab IDE, предназначенной для написания и отладки программ для однокристальных микроконтроллеров PICmicro. Материал пособия рассчитан на начинающих пользователей.

УДК 681.325.5-181.48(075.8)

ББК 32.973.26-04 я 73

ISBN

©Бурак А.И., В.Н. Левкович, 2003

© БГУИР, 2003

СОДЕРЖАНИЕ

Введение

1. Работа с MPLAB IDE

- 1.1. Настройка среды проектирования
- 1.2. Создание нового простого проекта
- 1.3. Создание нового исходного файла
- 1.4. Ввод исходного текста программы
- 1.5. Компиляция исходного файла
- 1.6. Запуск программы
- 1.7. Открытие дополнительных окон
- 1.8. Использование окон с переменными
- 1.9. Точки останова

2. Среда симулятора MPLAB SIM0

- 2.1. Функции отладчика MPLAB IDE
- 2.2. Среда симулятора MPLAB SIM
- 2.3. Характеристики симулятора MPLAB SIM
- 2.4. Точки останова и трассировки
- 2.5. Точки останова по условию
- 2.6. Функции стимула

Приложение 1. Описание кнопок графического меню MPLAB IDE

Приложение 2. Назначение полей линейки состояния

Приложение 3. Назначение файлов среды MPLAB IDE

ВВЕДЕНИЕ

MPLAB IDE– бесплатная интегрированная среда разработки для микроконтроллеров PICmicro фирмы Microchip Technology Incorporated. MPLAB IDE позволяет писать, отлаживать и оптимизировать текст программы.

MPLAB IDE включает в себя редактор текста, симулятор и менеджер проектов, поддерживает работу эмуляторов (MPLAB-ICE, PICMASTER) и программаторов (PICSTART plus, PRO MATE) фирмы Microchip и других отладочных средств фирмы Microchip и третьих производителей.

Легко настраиваемые инструментальные средства, тематическая помощь, «выпадающие» меню и назначение «горячих» клавиш в MPLAB IDE позволяют:

- получить код программы.
- наблюдать выполнение программы с помощью симулятора, определять время выполнения программы.
- просматривать текущее значение переменных и специальных регистров.
- работать с программаторами PICSTAR и PRO MATE II.
- использовать систему помощи по MPLAB IDE.

В данном пособии будут рассмотрены следующие модули MPLAB IDE:

- менеджер проекта MPLAB– используется для создания и работы с файлами, относящимися к проекту.
- редактор MPLAB– предназначен для написания и редактирования исходного текста программы, шаблонов и файлов сценария линкера.
- MPLAB-SIM симулятор– программный симулятор моделирует выполнение программы в микроконтроллере с учетом состояния портов ввода/вывода.
- MPASM ассемблер– компилирует исходный текст программы.

1. Работа с MPLAB IDE

1.1. Настройка среды проектирования

Рассмотрим работу в MPLAB IDE на конкретном примере.

Рабочий стол среды (рис.1) состоит из:

- 1) Главное текстовое меню.
- 2) Графическое меню.
- 3) Рабочая область, в которой размещаются открытые окна с файлами, диалогами или другой информацией.
- 4) Линейка состояния, отображающая текущую настройку системы.

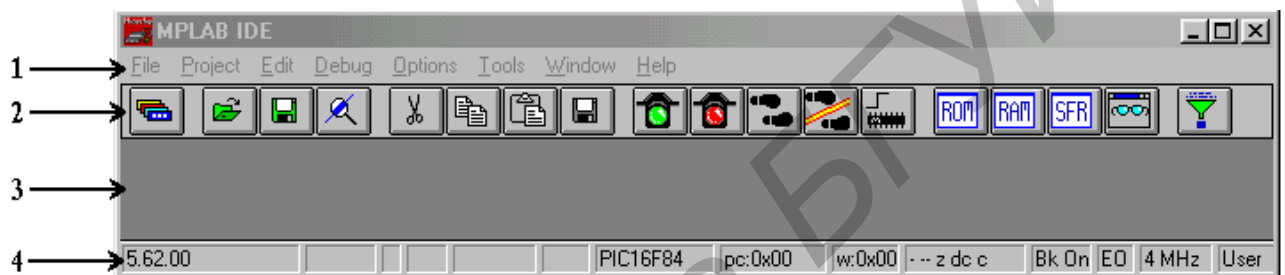


Рис. 1. Рабочий стол среды MPLAB IDE

Выберите пункт Options > Development Mode, нажмите кнопку Tools для выбора инструментального средства и типа микроконтроллера, используемого в проекте.

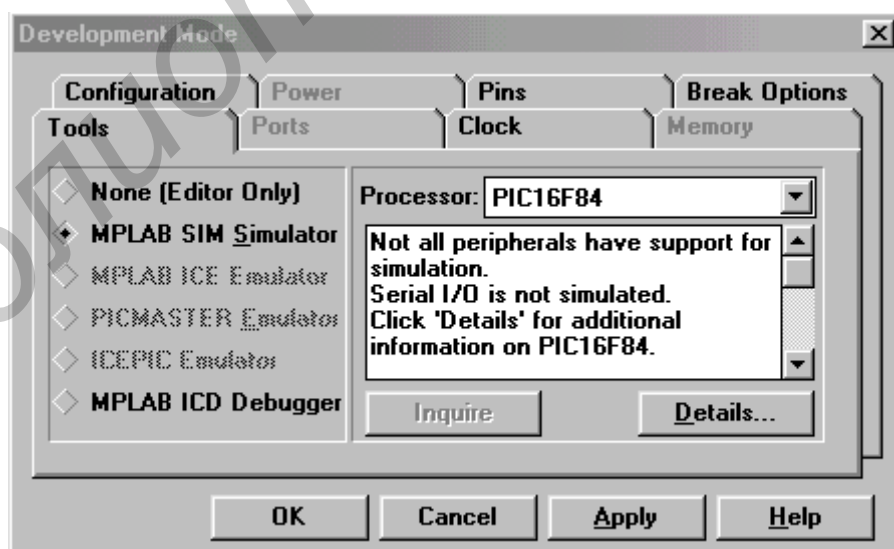


Рис.2. Меню Development Mode

Выберите симулятор MPLAB SIM и микроконтроллер PIC16F84.

Симулятор инициализирован, в линейке состояния на рабочем столе появится микроконтроллер «PIC16F84» и режим «SIM». Теперь среда

проектирования находится в режиме симулятора для микроконтроллера PIC16F84.

1.2. Создание нового простого проекта

Для работы симулятора MPLAB SIM нужен код программы (файл с расширением .HEX), который получается компиляцией исходного текста программы. В данном примере файл называется tutor84.hex, позже он может быть загружен непосредственно в микроконтроллер с помощью программатора.

Выберите пункт меню File > New и укажите, где вы хотите сохранить проект. В данном случае создается файл tutor84.pjt в каталоге “c:\Program Files \MPLAB”.

Тип файла .PJT будет назначен автоматически. Файлы с таким расширением являются файлами проекта в среде MPLAB IDE.

Подтверждение имени файла проекта и места его размещения приведет к переходу к диалоговому окну установки свойств компиляции. Пока можно использовать настройки по умолчанию. Нажмите кнопку ОК для сохранения параметров компиляции и возврату к диалоговому окну настройки проекта.

1.3. Создание нового исходного файла

Поставьте курсор в любое место пустого, неназванного файла, который был создан автоматически при создании нового проекта. Выберите пункт меню File > Save As... , укажите файл tutor84.asm, нужную директорию и нажмите кнопку ОК.

На рабочем столе MPLAB IDE будет открыт пустой файл с новым именем. Имя исходного файла должно быть такое же, как и имя проекта, в данном случае tutor84. Если изменяется имя исходного файла, также должно быть изменено имя проекта.

1.4 Ввод исходного текста программы

Сохраните приведенный ниже текст в файле tutor84.asm. При необходимости текст комментария вводится после символа “.”.

```
List p=16f84
Include <p16f84.inc>

C1    equ      0x0C ;
      org      0x00 ;

reset goto      start ;
      org      0x04 ;

start movlw     0x09 ;
      movwf    C1  ;

loop  incfz c1,F ;
      goto    loop ;
      goto    bug ;

end
```

1.5. Компиляция исходного файла

Компиляция исходного файла может быть выполнена несколькими способами. Описанный здесь метод использует пункт меню Project > Build All. После выбора указанного пункта меню исходный текст программы сохраняется и запускается программа MPASM. Как только компилирование будет завершено, на экране появится окно результатов (Рис.3).

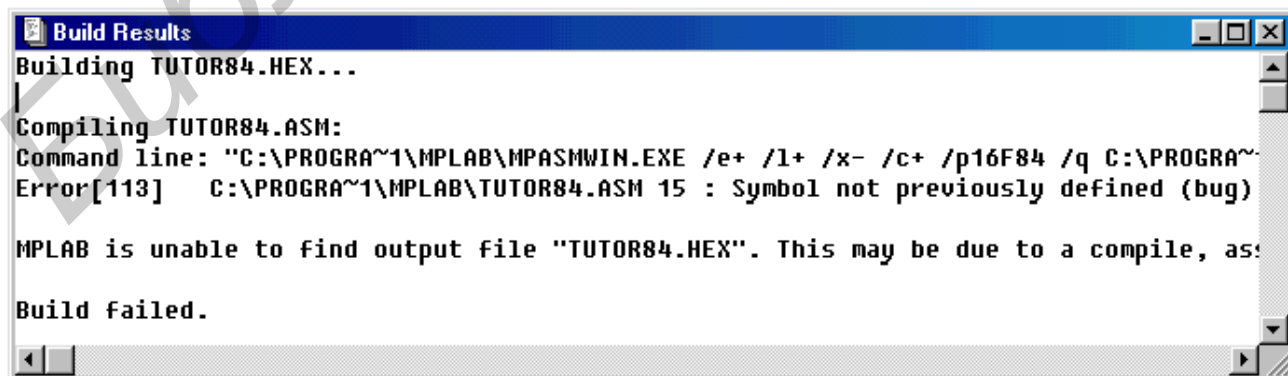


Рис. 3. Окно результатов компиляции

В последней строке текста программы преднамеренно была сделана ошибка (подраздел 1.4). При выполнении компилирования MPASM выдаст

ошибку о несуществующей метке. Двойной щелчок “мышью” на сообщении об ошибке перенесет курсор на строку в исходном тексте, где была сделана ошибка.

Исправьте последнюю строку, замените слово “bug” на “start”.

Вновь дайте команду “выполнить компиляцию” – Project > Build All.

После исправления всех ошибок на экране появится окно результатов с сообщением об успешной компиляции "Build completed successfully". Теперь можно использовать симулятор для проверки работы программы.

Примечание. При старте компиляции открытые исходные файлы сохраняются на диске.

1.6. Запуск программы

Выберите пункт меню Debug > Run > Reset для инициализации системы. Счетчик команд будет установлен в нуль, что является вектором сброса для микроконтроллера PIC16F84. В линейке состояния PC будет равен 0x00.

Затем выберите пункт меню Debug > Run > Step: темная полоса указывает на инструкцию в исходном тексте программы, которая будет выполнена следующей. Значение счетчика программы будет равно «PC:0x04».

Многие пункты меню продублированы комбинациями клавиш на клавиатуре компьютера и кнопками графического меню MPLAB IDE.

Пункт меню Debug > Run > Step - дублирующая клавиша <F7>. Нажмите кнопку <F7> несколько раз, наблюдая за значением счетчика команд PC и темной полосой, указывающей следующую инструкцию.

Выбором пункта меню Debug > Run > Run или нажатием на клавишу <F9> выполняется запуск программы с текущего места. Темная полоса, указывающая текущую инструкцию, изменит свой цвет, ни один параметр на линейке состояния не будет изменять своего значения до приостановки программы.

Остановите программу Debug > Run > Halt <F5>, указатель текущей инструкции примет первоначальный цвет, информация в линейке состояния обновится.

1.7. Открытие дополнительных окон

В MPLAB IDE существует большое количество способов контролировать ход выполнения программы. Например, программа, которая используется в примере, увеличивает значение счетчика, но как можно убедиться в том, что это действительно происходит? Для просмотра текущего значения регистра воспользуйтесь пунктом меню Window > File Registers. На экране появится небольшое окно со всеми регистрами ОЗУ PIC16F84.

При каждом шаге программы (нажатие клавиши <F7>) значение регистров в окне будет обновляться. В нашем случае изменяется значение счетчика в регистре с адресом 0x0C. Изменение значения регистра отображается в окне памяти данных, выделяя его другим цветом. Однако в сложных программах одновременно могут изменяться несколько регистров, что усложняет контроль хода выполнения программы. Проблема может быть решена использованием окон с переменными.

1.8. Использование окон с переменными

1.8.1. Создание окна

Чтобы создать окно, выберите пункт меню Window > Watch Window > New Watch Window.

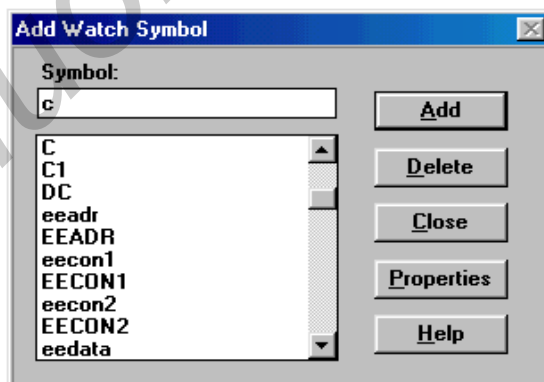


Рис. 4. Диалоговое окно добавления переменных

На экране появится диалоговое окно добавления переменных, показанное на рис.4.

Введите в поле 'Symbol:' новую переменную 'c1' и нажмите кнопку «Add» (добавить). Переменная добавится в список. Нажмите кнопку Close, и на экране появится окно с переменной C1 (рис. 5):

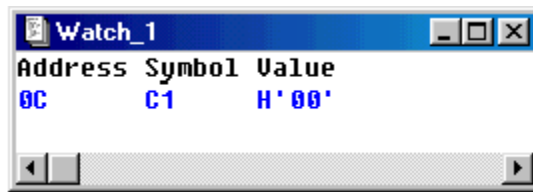


Рис.5. Окно просмотра переменных

1.8.2. Сохранение окна

Сохранение параметров окна на диске выполняется командой `Window > Watch Window > Save Active Watch` или выбором команды `Save Watch` в системном меню окна (для вызова системного меню окна нажмите на иконку в левом верхнем углу окна). На экране появится диалог сохранения, показанный на рисунке. Введите имя и нажмите кнопку ОК. Состояние окна, его расположение на экране сохраняется вместе с проектом. При открытии проекта окно также будет восстановлено.

1.8.3. Редактирование окна

Редактирование окна можно выполнять только после его создания (открытия). Для редактирования окна вы можете использовать подменю `Window > Watch Window` или системное меню окна. С его помощью можно добавлять и удалять переменные, управлять настройками отображения данных.

1.9. Точки останова

Нажатием кнопки `<F5>` (`Debug > Run > Halt`) остановите выполнение программы. Щелкните левой кнопкой “мыши” на строке с текстом `'movlw 0x09'`, затем правой кнопкой “мыши” на той же строке, появится диалоговое окно, показанное на рис 6.

Выберите пункт `Break Point(s)`, указательная линия изменит свой цвет, показывая точку останова.

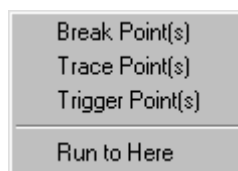


Рис.6. Контекстное меню точек остановки

Нажмите кнопку <F6> (Debug > Run > Reset) для сброса, а затем <F9> для начала выполнения программы.

Выполнение программы будет прервано в точке остановки, значение переменной 'c1' после сброса будет равно 0x00 (вы можете увидеть значение регистра в окне переменных или в окне памяти данных). Выполнение одного шага программы изменит значение переменной 'c1' на 0x09. Нажмите кнопку <F9> несколько раз, вы заметите изменение цвета линейки состояния во время выполнения программы.

Примечание. Если выполнение программы не прерывается в точке остановки, то проверьте глобальное разрешение остановки программы Global Break Enable в пункте меню Options >Development Mode- Break Options.

2. ОТЛАДЧИК И СИМУЛЯТОР MPLAB IDE

2.1. Функции отладчика MPLAB IDE

После ввода и компиляции проекта в MPLAB IDE можно посмотреть, как выполняется программа. Одним из методов может быть программирование микроконтроллера и установка его в устройство, чтобы проверить работу программы. Как правило, с первого раза новая программа работать будет неправильно и потребуются отладка текста программы.

Отладку программы можно выполнить с помощью симулятора MPLAB SIM или эмулятора MPLAB-ICE для управления реальным устройством.

В любом случае будет необходимо прерывание выполнения программы для проверки значения регистров общего и специального назначения и состояния процессора.

MPLAB SIM моделирует выполнение программы любого типа PICmicro с учетом состояния портов ввода/вывода на скорости, которая зависит от быстродействия персонального компьютера. Симулятор поддерживает следующие функции:

- эмуляция памяти программ;
- прерывание выполнения программы в точках остановки;
- работа по шагам;
- контроль регистров общего и специального назначения.

Все указанные функции используют данные из проекта MPLAB IDE. Символьные метки могут использоваться для указания точек остановки и трассировки, а также изменения значений регистров памяти данных.

В режиме симулятора инструкции выполняются с той скоростью, которую позволяют вычислительные мощности персонального компьютера. Скорость выполнения программы в симуляторе значительно медленнее, чем при реальной работе, поскольку симулятор должен следить за состоянием портов ввода/вывода, значением регистров памяти данных общего и служебного назначения, учитывать точки остановки и трассировки.

В симуляторе предусмотрена автоматическая работа по шагам. Программа выполняется непрерывно, но значительно медленнее, с обновлением всех значений в открытых окнах после исполнения каждой инструкции.

2.2 Среда симулятора MPLAB SIM

Дискретный симулятор MPLAB SIM встроен в среду разработки MPLAB IDE и предназначен для следующих семейств микроконтроллеров PICmicro:

- PIC12CXXX
- PIC14000
- PIC16C5X
- PIC16CXX
- PIC16FXXXX
- PIC17CXXX

- PIC18CXXX
- PIC18FXXX

2.3 Характеристики симулятора MPLAB SIM

MPLAB SIM выполняет контроль и управление состоянием портов ввода/вывода на границах инструкций с периодичностью T_{CY}. T_{CY} – время выполнения одной инструкции равно 4 T_{osc}, где T_{osc} период тактового генератора микроконтроллера. Поэтому некоторые физические события не могут быть смоделированы точно, это

- асинхронные сигналы (относительно тактового генератора микроконтроллера);
- сигналы, которые имеют период меньше T_{CY}.

Все внешние события синхронизируются с границами циклов команд микроконтроллера, поэтому событие длительностью меньше одного машинного цикла смоделировано быть не может. Следующие ограничения распространяются на симуляцию портов ввода/вывода и периферийных модулей PICmicro из-за синхронизации событий с границами цикла инструкции:

- тактовые импульсы таймеров не могут быть меньше времени T_{CY}, хотя предделитель таймера способен обрабатывать импульсы короче времени выполнения одной инструкции;
- ШИМ-импульсы, меньше T_{CY}, не поддерживаются;
- не поддерживается 8-разрядное сравнение;
- асинхронные сигналы (относительно тактового генератора микроконтроллера) не моделируются;
- форму сигнала генератора на выводах RCO/RC1 описывать нельзя;
- MPLAB SIM не моделирует работу последовательных интерфейсов.

2.4. Точки остановки и трассировки

Функции отладки, влияющие на выполнение инструкций программы, основаны на следующих элементах:

- точки остановки;
- точки трассировки;
- счетчик проходов.

Максимальное число именованных адресов точек каждого типа – 16. Точки трассировки и остановки полностью независимы друг от друга и могут быть установлены в любой части программы.

Настройка точек остановки производится в меню `Debug > Break Settings`, точек трассировки - `Debug > Trace Settings`.

2.4.1. Точки остановки

Точка остановки – условие, при котором микроконтроллер прекращает выполнение программы.

MPLAB IDE обеспечивает следующие условия остановки:

- совпадение адреса точки остановки с программным счетчиком PC;
- заполнение буфера трассировки;
- выполнение указанного числа проходов точки остановки;
- переполнение стека;
- переполнение счетчика сторожевого таймера WDT;
- остановка пользователем.

Остановка при совпадении адреса точки остановки со счетчиком команд PC прерывает ход выполнения программы на инструкции, адрес которой соответствует адресу точки остановки. Например, если точка остановки назначена на адрес 5Ah, то остановка будет выполнена на адресе 5Ah до выполнения инструкции.

Остановка при полном буфере трассировки прерывает ход выполнения программы при заполнении выделенного 3-Кбайтного буфера трассировки.

Остановка после выполнения указанного числа проходов MPLAB IDE

позволяет указывать число проходов точки остановки. Счетчик проходов используется для того, чтобы остановить выполнение программы после определенного числа проходов точки остановки. Например, если разрешен счет числа проходов, то программа будет приостановлена в точке остановки, когда значение счетчика обнулится.

Остановка при переполнении стека прерывает ход выполнения программы при переполнении аппаратного стека микроконтроллера.

Остановка при срабатывании сторожевого таймера WDT прерывает ход выполнения программы при переполнении сторожевого таймера.

Остановка пользователем. Пользователь прерывает выполнение программы следующими действиями:

- выбор пункта меню Debug > Run > Halt,
- нажатие кнопки <F5> на клавиатуре;
- нажатие кнопки остановки на графическом меню MPLAB IDE.

2.4.2. Точки трассировки

Дополнительным средством контроля за ходом выполнения программы являются точки трассировки.

Симулятор MPLAB SIM поддерживает 3-Кбайтный буфер трассировки, в котором сохраняются данные о состоянии микроконтроллера. Допускается запись в буфер при переполнении, удаляя более старые значения (если не выбран параметр остановки программы при переполнении буфера). MPLAB IDE непрерывно фиксирует состояние микроконтроллера в указанных точках трассировки. Структура информации, записываемой в буфер:

- 16-разрядный адрес точки трассировки;
- 16-разрядный код и данные инструкции;
- время и измененные регистры.

Остановка записи в буфер трассировки (Halt Trace) позволяет просмотреть состояние микроконтроллера в точках трассировки без нарушения хода выполнения программы. Для продолжения записи в буфер трассировки

нажмите кнопку Halt Trace.

Просмотр состояния микроконтроллера в ходе выполнения программы в симуляторе MPLAB SIM возможен при помощи окна буфера трассировки. Каждая запись буфера трассировки содержит адрес в памяти программ, выполняемый код, время записи и измененные регистры. При записи времени в буфер трассировки используются часы MPLAB IDE. Сброс времени записи осуществляется при сбросе системных часов MPLAB IDE.

2.4.3. Настройка числа проходов для точек останова

MPLAB IDE позволяет назначить один 16-разрядный счетчик прохода для любого адреса памяти программ.

Когда выполнение программы приостановлено, можно изменить число проходов точки останова/трассировки в диалоговых окнах настройки точек останова/трассировки соответственно. Сначала указывается желаемый адрес (или диапазон адресов), а затем записывается 16-разрядное значение счетчика проходов. Когда счетчик проходов будет декрементирован до нуля, произойдет остановка программы (запись в буфер трассировки).

Если указан счетчик проходов, то остановка программы произойдет при обнаружении другого условия останова или при обнулении счетчика проходов.

Пример последовательности настройки точек останова показан на рис.7. Следует иметь в виду, что настраиваемые точки останова независимы друг от друга. Создадим две точки останова со следующими условиями: при выполнении любых инструкций в пределах диапазона адресов от 100 до 200 и после выполнения 10000 проходов адреса 500.

1. Настроить диапазон адресов 100-200 для точки останова:

- введите имя Halt_Range в пункте Title;
- введите начала диапазона адресов 100 в строке Start и конец диапазона 200 в строке End;

- нажмите кнопку Add для сохранения настроек.
2. Настроить вторую точку останова с адресом 500:
- введите имя 10K_Passes в пункте Title;
 - введите начала диапазона адресов 500 в строке Start и конец диапазона 500 в строке End;
 - нажмите кнопку Add для сохранения настроек.
3. Настроить остановку после выполнения 10000 проходов адреса 500:
- выберите точку останова 10K_Passes;
 - отметьте пункт Address is Qualifier,
 - введите значение 10000 в пункте Pass Count;
 - нажмите кнопку See для сохранения настроек.

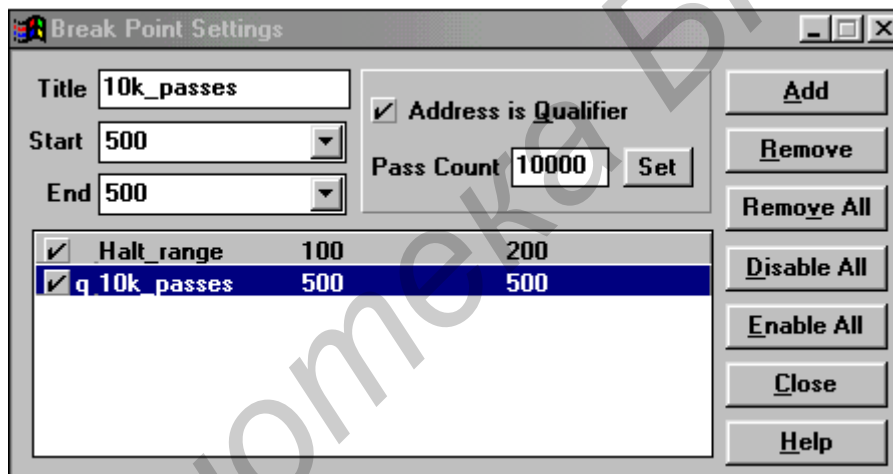


Рис. 7. Меню настроек точек останова

2.4.4. Настройка числа проходов для точки трассировки

Если указан счетчик проходов, то пока его значение не станет равно нулю, данные в буфер трассировки записываться не будут. Когда значение счетчика проходов будет равно нулю, данные записываются в буфер трассировки при проходе через точку трассировки.

Счетчик проходов декрементируется каждый раз при выполнении команды с указанным адресом. Целесообразно использовать эту особенность, чтобы определить число раз исполнения команды.

2.5. Точки остановки по условию

Если настроена точка остановки по условию, MPLAB IDE прервет выполнение программы, когда значение внутреннего регистра микроконтроллера будет удовлетворять указанному условию.

2.5.1. Условия

MPLAB IDE остановит ход выполнения программы в одном из следующих случаев:

- остановка пользователем– MPLAB IDE продолжает выполнять программу, пока не будет нажата кнопка Halt;
- выбор опции Halt в диалоговом окне точки остановки по условию;
- остановка после выполнения указанного числа циклов микроконтроллера;
- удовлетворение условиям остановки.

2.5.2. Данные трассировки

Данные трассировки позволяют контролировать значения регистров в диалоговом окне точек остановки по условию.

2.5.3. Единственный цикл

MPLAB IDE выполняет отдельные циклы микроконтроллера, пока не встретится условие остановки.

2.5.4. Многократные циклы

В режиме многократных циклов инструкции выполняются в режиме реального времени (для эмулятора), проверяются условия выбранных точек остановки и условной остановки. Симулятор останавливается только при выполнении указанного условия. Точки остановки и условия остановки по значению регистров проверяются только в точках, определенных в диалоговом окне настройки параметров точек остановки.

2.6. Функции стимула

Стимул подготавливает сигналы для симулятора MPLAB SIM. Программисту предоставляется возможность моделировать состояние портов ввода/вывода или записывать значения непосредственно в регистры.

Существует четыре вида стимулов:

- асинхронный стимул– непосредственное управление состоянием портов ввода/вывода;
- файл состояния порта ввода/вывода– текстовый файл, описывающий состояние порта ввода/вывода;
- файл стимула регистра– текстовый файл, содержащий 8-разрядное значение регистра;
- стимул тактового сигнала– регулярный, программируемый, периодический источник тактового сигнала.

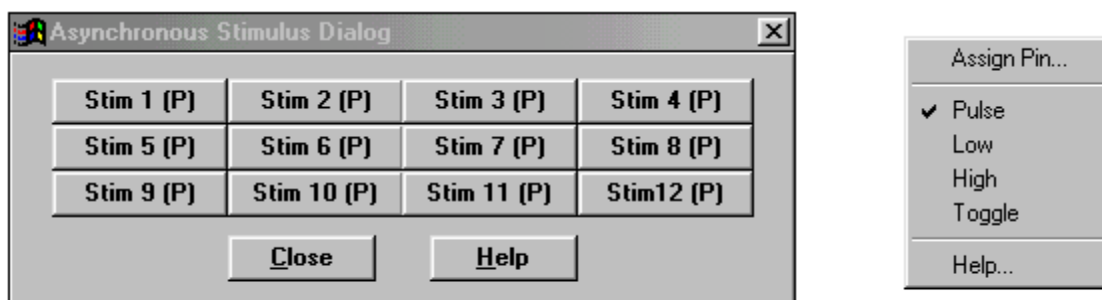
2.6.1. Асинхронный стимул

Используется для моделирования логического состояния порта ввода/вывода, настроенного на вход (установка значений +5В или 0В). Вы можете нажатием кнопки в диалоговом окне асинхронного стимула указывать входной сигнал на портах ввода/вывода.

Для примера рассмотрим управление состоянием сигнала на входе RB0 PORTB микроконтроллера PIC16F84.

Выберите пункт меню Debug > Simulator Stimulus > Asynchronous Stimulus, на экране появится диалоговое окно, показанное на рис.8,а.

Поместите курсор на кнопку «Stim 1 (P)» и нажмите правую кнопку «мыши», на экране появится сокращенное меню (рис. 8,б).



а

б

Рис.8. Меню настройки асинхронного воздействия

Выберите пункт Toggle.

Снова поместите курсор на кнопку «Stim 1 (T)» (символ «P» изменился на «T» – Toggle) и нажмите правую кнопку «мыши», на экране появится сокращенное меню выбора портов ввода/вывода PIC16F84.

Двойным щелчком левой кнопкой «мыши» выберите порт RBO. Обратите внимание, что значение «Stim 1 (T)» заменено на «RBO (T)».

Выберите пункт меню Debug > Run > Animate для запуска быстрого выполнения программы по шагам. Нажмите кнопку «RBO (T)» в диалоговом окне асинхронного стимула. Изменение состояния PORTB можно увидеть в окне регистров специального назначения.

2.6.2. Файлы стимула порта ввода/вывода

Текстовый файл стимула портов ввода/вывода состоит из нескольких колонок. Первая колонка определяет номер цикла при выполнении программы, в котором будут изменяться значения портов ввода/вывода.

Порядок создания файла стимула:

1. Создайте новый текстовый файл (File > New File), в котором будет вводиться текст файла стимула.
2. Впишите первое слово в первой строке “CYCLE”.
3. Первый столбец определяет номер цикла (в соответствии с секундомером MPLAB IDE), в котором будет изменяться состояние порта ввода/вывода.

4. Справа от слова CYCLE укажите имена выводов микроконтроллера, которые должны соответствовать типу моделируемого микроконтроллера.
5. Чтобы увидеть список поддерживаемых портов ввода/вывода, воспользуйтесь пунктом меню Debug > Simulator Stimulus > Asynchronous Stimulus и нажмите правой кнопкой «мыши» на одной из кнопок окна.
6. Заполните таблицу, указав номер цикла и состояние вывода. После знака «;» или «!» можно вписать комментарии.
7. Сохраните файл стимула - File > Save As... Теперь файл готов к использованию в проекте.

Порядок использования файла стимула:

1. Подключите файл стимула к проекту - Debug > Simulator Stimulus > >Pin Stimulus > Enable.
2. Откройте дополнительные окна. Например, окно секундомера Window > Stopwatch и окно специальных регистров Window > Special Function Registers. В окне секундомера будет показано прошедшее время при выполнении каждой команды. Если секундомер сбрасывается, файл стимула портов ввода/вывода тоже сбрасывается в начальное состояние.
3. Сбросьте микроконтроллер и наблюдайте за состоянием порта ввода/вывода, выполняя программу по шагам.

2.6.3. *Файл стимула регистра*

Текстовый файл стимула регистра состоит из одной колонки значений, которые будут переданы регистру, когда адрес в счетчика команд РС равен значению, указанному в диалоговом окне стимула регистра. Это может быть полезно, например, при моделировании работы АЦП.

Порядок создания файла стимула регистра:

1. Создайте новый текстовый файл - File > New File. Вы будете вводить текст файла стимула в этом окне.
2. В безымянном окне введите список значений в одну колонку, которые Вы хотите назначить регистру.
3. Сохраните созданный файл - File > Save As... Выберите директорию, в которой хотите сохранить файл, введите имя файла с расширением .reg.

Порядок использования файла стимула регистра:

1. Выберите пункт меню Debug > Simulator Stimulus > Register Stimulus > Enable для открытия диалогового окна настройки стимула регистра.
2. Укажите адрес в памяти программ, в котором значения из файла должны быть подставлены в регистр.
3. Укажите адрес регистра в памяти данных, в который будут подставляться значения из файла.
4. Нажмите кнопку Browse и выберите файл стимула регистра.
5. Откройте окно памяти данных - Window > File Registers.
6. Сбросьте микроконтроллер и выполните шаги программы. Каждый раз, когда значение адреса счетчика команд РС совпадет с указанным адресом в диалоговом окне, в регистр будет занесено значение из файла стимула. Значения из файла стимула будут подставляться циклически, пока выполняется программа в симуляторе MPLAB-SIM.

Пример работы с файлом стимула регистра. Этот пример подразумевает, что создан проект, описанный в главе 2.

1. Создайте новый файл - File > New File и введите построчно следующие числа: 10, 2E, 38, 41, 50, 7A, 99, AO, FD.
2. Сохраните файл File > Save As... с именем tutor84.reg. Этот файл будет использоваться для последовательного ввода значений в регистр.
3. Выберите пункт меню Debug > Simulator Stimulus > Register Stimulus > Enable для открытия диалогового окна стимула регистра (рис. 9).

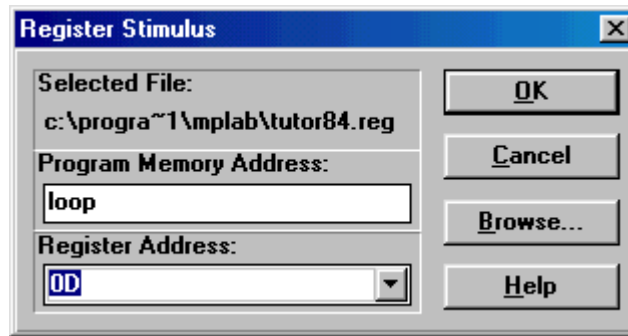


Рис. 9. Меню стимула регистра

- Укажите в качестве адреса памяти программ (Program Memory Address) метку Loop, адрес регистра памяти данных (Register Address) 0x0D и нажмите кнопку Browse для указания файла стимула tutor84.reg.
4. Откройте окно с регистрами Window > File Registers для наблюдения эффекта стимула регистра.
 5. Сбросьте микроконтроллер и выполните шаги программы. Каждый раз, когда значения адреса счетчика команд PC совпадет с указанным адресом Loop, в регистр 0x0D будет занесено значение из файла стимула tutor84.reg. Значения 0x10, 0x2E будут внесены в регистр, выбранный в диалоговом окне стимула регистра, при проходе через метку Loop. После того как подставлено последнее значение (0xFD из файла tutor84.reg), при следующей подстановке будет использоваться первое значение (0x10).

2.6.4 Стимул тактового сигнала

Стимул тактового сигнала формирует на входе порта ввода/вывода регулярную форму сигнала с периодичностью, кратной тактовым циклам микроконтроллера.

Выберите пункт меню Debug > Simulator Stimulus > Clock Stimulus для открытия диалогового окна стимула тактового сигнала (рис. 10).

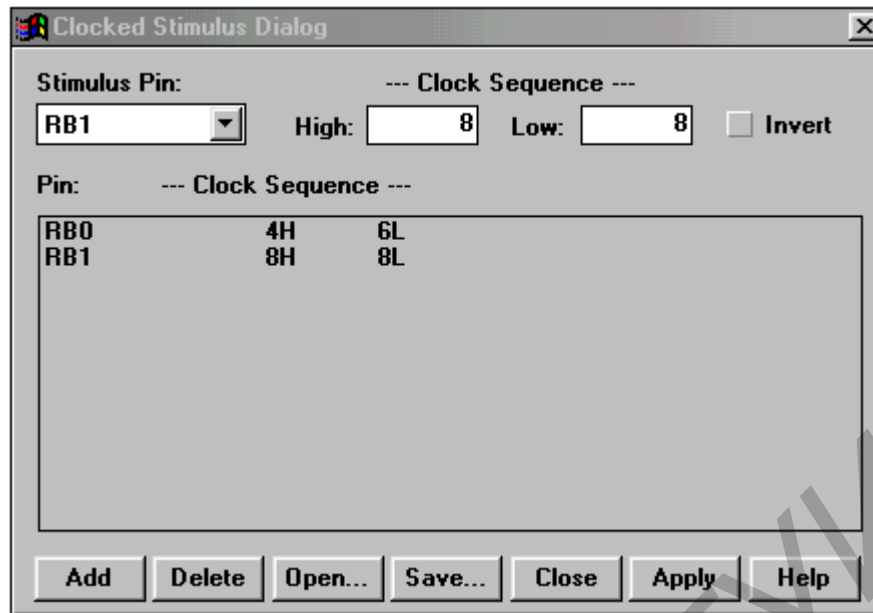


Рис. 10. Меню стимула тактового сигнала

Настройки, показанные на рисунке следующие:

RB0– 4 цикла- высокий уровень, 6 циклов- низкий;

RB1– 8 циклов- высокий уровень, 8 циклов- низкий;

Для добавления стимула тактового сигнала укажите имя порта ввода/вывода, длительность высокого и низкого уровня сигнала и нажмите кнопку Add.

Описание кнопок графического меню MPLAB IDE

Меню редактора MPLAB IDE



Расположение и назначение кнопок по умолчанию.

1. Выбор графического меню.
2. Создать новый файл.
3. Открыть файл.
4. Сохранить файл.
5. Вырезать в буфер.
6. Копировать в буфер.
7. Вставить из буфера.
8. Печатать.
9. Найти.
10. Повторить поиск.
11. Заменить.
12. Повторить замену.
13. Отменить действие.
14. Увеличить отступ.
15. Уменьшить отступ.
16. Перейти на строку.
17. Показать нумерацию строк.
18. Контекстная помощь.

Меню отладчика



Расположение и назначение кнопок по умолчанию.

1. Выбор графического меню.
2. Запустить выполнение программы с текущего места.
3. Остановить выполнение программы.
4. Выполнить текущую инструкцию программы.
5. Выполнить текущую инструкцию программы (инструкции CALL выполняются за один шаг с полным выполнением подпрограммы).
6. Сброс микроконтроллера.
7. Выбрать значение счетчика команд PC.
8. Добавить инструкцию.
9. Новое окно с переменными.
10. Окно изменений.
11. Точка остановки.
12. Точка трассировки.
13. Триггер.
14. Сбросить все точки остановки.
15. Условная остановка.
16. Прекратить запись в буфер трассировки.
17. Системный сброс.
18. Отличия версии.

Меню проекта



Расположение и назначение кнопок по умолчанию.

1. Выбор графического меню.
2. Создать новый проект.
3. Открыть проект.
4. Закрыть проект.
5. Сохранить проект.
6. Редактировать параметры проекта.
7. Выполнить проект.
8. Компиляция всех исходных файлов.
9. Компиляция текущего исходного файла.
10. Подключение модулей к проекту.
11. Помощь.

Библиотека БГУИР

Меню пользователя



Расположение и назначение кнопок по умолчанию.

1. Выбор графического меню.
2. Открыть проект.
3. Закрывать проект.
4. Найти.
5. Вырезать в буфер.
6. Копировать в буфер.
7. Вставить из буфера.
8. Сохранить файл.
9. Запустить выполнение программы с текущего места.
10. Остановить выполнение программы.
11. Выполнить текущую инструкцию программы.
12. Выполнить текущую инструкцию программы (инструкции CALL выполняются за один шаг с полным выполнением подпрограммы).
13. Сброс микроконтроллера.
14. Окно памяти программ.
15. Окно памяти данных.
16. Окно регистров специального назначения.
17. Новое окно с переменными.
18. Выполнить проект.

Назначение полей линейки состояния

Формат поля	Пример отображения	Описание	Результат двойного нажатия левой кнопкой «мыши»
Строка №, Колонка № в активном окне или версия MPLAB IDE	Ln 1 Col 1	Номер строки и колонки размещения курсора в активном окне редактора. Номер версии MPLAB IDE	Открытие диалога перехода на определенную строку Не активно
Количество строк в файле	18	Отображает количество строк в активном окне редактора MPLAB IDE	Не активно
Модификация файла		Символ «#» появляется, если редактированный файл не был сохранен	Не активно
Защита от редактирования	WR	Включение/отключение защиты от редактирования: WR - редактирование разрешено RD - только чтение	Включает/отключает защиту от редактирования
Выравнивание текста	No Warp	Отображаются параметры выравнивания текста No Warp - выравнивание нет WR 72 - выравнивание текста, максимальное число символов в линии 72	Включает/выключает выравнивание текста
Параметр ввода текста	INS	Отображается параметр ввода текста INS - вставка OVR - замещение	Переключает параметр ввода текста вставка/замещение
Тип микроконтроллера	PIC16F84	Отображается тип микроконтроллера используемого в проекте	Не активно
Счетчик команд	PC:0x00	Текущее значение счетчика команд PC	Открывает диалог выбора значения счетчика команд
Регистр W	W:0x00	Значение регистра W	Не активно
Биты статуса	Z dc c	Состояние битов ALU Верхний регистр - бит установлен Нижний регистр - бит сброшен	Не активно
Флаг разрешения прерываний	BkOn	Глобальное разрешение прерываний	Разрешет/запрещает прерывания
Среда проектирования	Sim	EO - только редактор Sim - симулятор MPLAB SIM Si - симулятор SIMICE ICE - эмулятор MPLAB-ICE Em - эмулятор PICMASTER	Открывает окно настройки проекта
Частота	4MHz	Тактовая частота микроконтроллера	Открывает окно

			настройки проекта
Режим	User	Режим среды проектирования MPLAB IDE	Не активно

Приложение 3

Назначение файлов среды MPLAB IDE

*.ASM	исходный файл на ассемблере
*.C	исходный файл на C
*.CFG	файл конфигурации
*.CSV	файл трассировки, только для MPLAB-ICE 2000
*.COD	объектный код
*.DAT	файл данных симулятора
*.ERR	список ошибок при компиляции
*.H	дополнительный файл C
*.HEX	код для микроконтроллера PICmicro
*.HLP	файл помощи
*.MNC	настройка MPASM
*.MNI	настройка MPLAB IDE
*.KEY	назначение клавиш в MPLAB IDE
*.LKR	файл сценария для MPLINK
*.LST	файл листинга программы
*.PJT	файл проекта
*.REG	файл стимула регистра
*.STI	файл стимула вывода
*.TB	файл настройки точек останова
*.TBR	настройки графического меню
*.TPL	файл шаблонов
*.TRC	файл буфера трассировки
*.WAT	файл окна с переменными

Св. план 2003, поз. 56
Учебное издание

Бурак Андрей Иосифович,
Левкович Василий Николаевич

**АРХИТЕКТУРА И ОСНОВЫ ПРОГРАММИРОВАНИЯ
ОДНОКРИСТАЛЬНЫХ МИКРОКОНТРОЛЛЕРОВ PIC16F84**

Методическое пособие
к лабораторным работам
по дисциплине
«Цифровые и микропроцессорные устройства»
для студентов специальностей
390101 «Радиотехника» и 390102 «Радиоэлектронные системы»
всех форм обучения

Редактор Е.Н. Батурчик

Подписано в печать
Бумага
Усл. печ. л.
Заказ

Формат 60x84 1/16.
Печать Гарнитура
Уч.-изд. л. 2.0. Тираж 100 экз.

Издатель и полиграфическое оформление:
Учреждение образования
«Белорусский государственный университет информатики
и радиоэлектроники»
Лицензия ЛП №156 от 05.02.2001.
Лицензия ЛВ №509 от 03.08.2001.
220013, Минск, П.Бровки,6