МИНИСТЕРСТВО ОБРАЗОВАНИЕ И НАУКИ УКРАИНЫ

НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

«ХАРЬКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ ИНСТИТУТ»

Кафедра «Стратегическоеуправление»

Отчет к лабораторной работе №9

Выполнил:

студент группы КН-26

Величко А.Ю.

Проверила: Букреева К.С.

Харьков 2017

**Рекомендации к работе:** Что бы решить заданную задачу, выберите микроконтроллер PIC16F84. Он имеет перезаписанную пользователем PROMпрограммы и может работать с прерыванием.

Прерывание в этом типе микроконтроллера может возникать по многим причинам, и среди них переполнение счетчика/таймера(TMRO).

Используя особенности архитектуры микроконтроллера PIC16F34, мы можем предложить следующий алгоритм для решения задачи:

*Шаг 1.*Организовать процедуру ожидания прибытия первого импульса, например, на вход RA1 микроконтроллера.

*Шаг 2*. Запустите внутренний счетчик времени после обнаружения импульса.

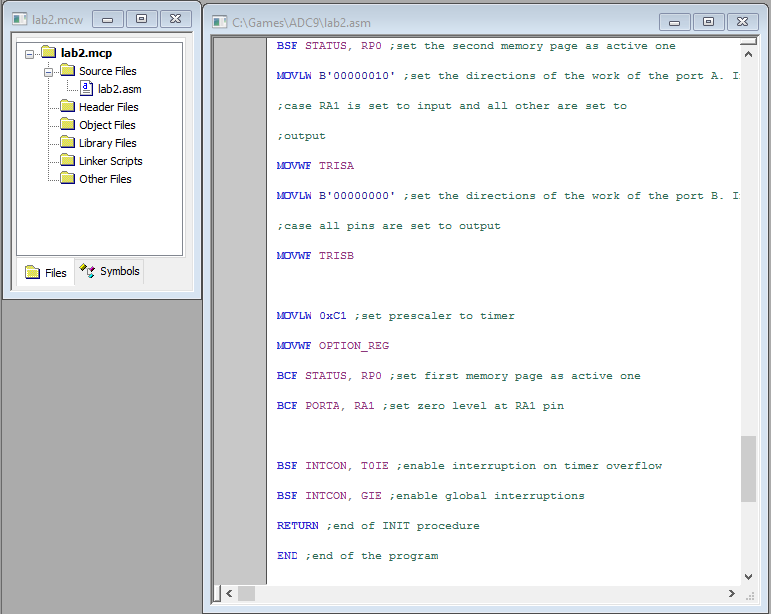
*Шаг 3*. Организовать процедуру ожидания прибытия второго импульса на том же вход RA1 микроконтроллера.

*Шаг 4.* Остановите внутренний счетчик / таймер после обнаружения второго импульса.

*Шаг 5.* Выведите результат из регистра счетчика / таймера в порт строб-сигнал, показывающий готовность данных на выходе микроконтроллера. Строб сигнал может выводиться на выход RA0.

**Ход работы:**

Создадим новый проект в MPLAB, который назовем lab2. В этом проекте скомпилируем заданный код.





В коде были обнаружены ошибки. Рассмотрим ошибки и исправим их.

**1**. Предупреждение [207] H: \ LECTURES \ LABWORKS \ LAB2 \ SRC \ LAB2.ASM 38:

(RB7) - эта строка описывает предупреждение. Предупреждение состоит из предопределенного числа- 207, путь к исходному файлу, номер строки в исходном файле, где было обнаружено предупреждение - 38,

RB7 EQU 07H

Это предупреждение появилось из-за пустых мест в начале программы.

Наши действия: удалим пустые места с начала строки.

**2**. Предупреждение [205] H: \ LECTURES \ LABWORKS \ LAB2 \ SRC \ LAB2.ASM 47.

(ORG). Давайте рассмотрим это предупреждение.ORG - это директива в MPASM, и директивы в соответствии с правилами MPASM не должны запускаться непосредственно с начала строки.

Наши действия: поместите 1-2 пустых места в начале строки перед ORG или просто нажмите TAB.

**3**. Предупреждение [203] H: \ LECTURES \ LABWORKS \ LAB2 \ SRC \ LAB2.ASM 48:

(GOTO). Это предупреждение похоже на предыдущее предупреждение. Исправьте его так же.

Ошибка [113] H: \ LECTURES \ LABWORKS \ LAB2 \ SRC \ LAB2.ASM 48.

(BEGIN). Давайте посмотрим на строку ошибок.

GOTO BEGIN

Эта строка кода перемещает поток программы на метку BEGIN. Итак, эта строка синтаксически правильный. Найдем ярлык BEGIN.Этот ярлык отсутствует в нашем коде.

Вместо этого у нас есть ярлык BEGIB. Так что в этом случае у нас есть тривиальная опечатка.

Наши действия: изменить BEGIB на BEGIN в строке 59

**4**. Предупреждение [205] H: \ LECTURES \ LABWORKS \ LAB2 \ SRC \ LAB2.ASM 49.

(ORG). Это предупреждение похоже на предупреждение номер 3. Исправьте его так же.

**5**. Предупреждение [203] H: \ LECTURES \ LABWORKS \ LAB2 \ SRC \ LAB2.ASM 51.

1. (INCF). Это предупреждение похоже на предупреждение номер 3. Исправьте его так же.

**6**. Ошибка [113] H: \ LECTURES \ LABWORKS \ LAB2 \ SRC \ LAB2.ASM 51.

Не определено ранее (TMR1). Проверьте определение регистров в разделе заголовка.

Определение регистра TMR1 отсутствует.

Наши действия: добавить определение TMR1 в раздел заголовкаTMR1 EQU 0CH

7. Предупреждение [203] H: \ LECTURES \ LABWORKS \ LAB2 \ SRC \ LAB2.ASM 55.

(CLRF). Это предупреждение похоже на предупреждение номер 3. Исправьте его так же.

8. Предупреждение [203] H: \ LECTURES \ LABWORKS \ LAB2 \ SRC \ LAB2.ASM 56.

(BCF).Это предупреждение похоже на предупреждение номер 3. Исправьте его так же.

**9**. Предупреждение [203] H: \ LECTURES \ LABWORKS \ LAB2 \ SRC \ LAB2.ASM 57.

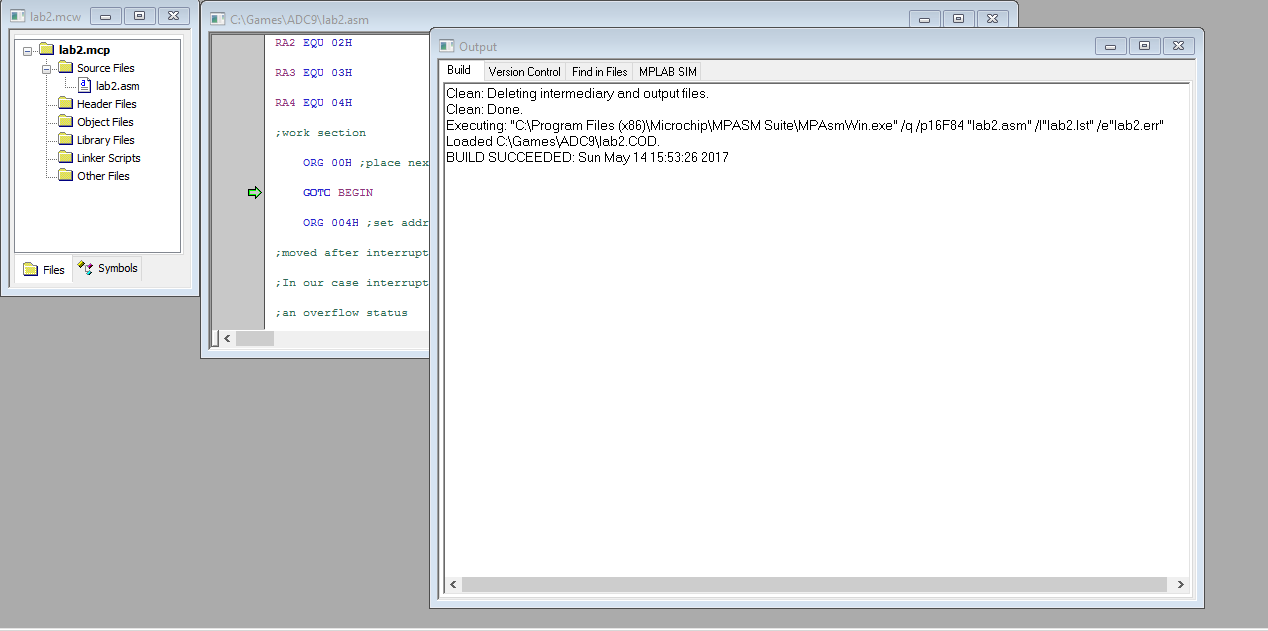
(RETFIE). Это предупреждение похоже на предупреждение номер 3. Исправьте его так же.

**10**. Ошибка [113] H: \ LECTURES \ LABWORKS \ LAB2 \ SRC \ LAB2.ASM 63.

(PA1). Проверьте определение регистров в разделе заголовка и определите, чтобиты порта А определяются как RA0, RA1, RA2, RA3, RA4. Но мы используем РА0, РА1, PA2, PA3, PA4 в программе.

Наши действия: исправить имена регистров

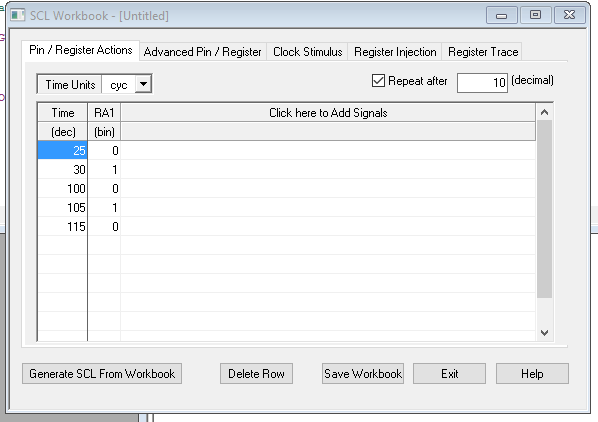
Следующий набор ошибок аналогичен предыдущим, поэтому мы их опускаем. Вы легко можете сами их исправить.



Теперь программа готова к записи в ППЗУ микроконтроллера. Но чтобы убедиться, что у него нет логических ошибок, имитируем отлаживаниепрограммы с помощью MPLAB SIM.

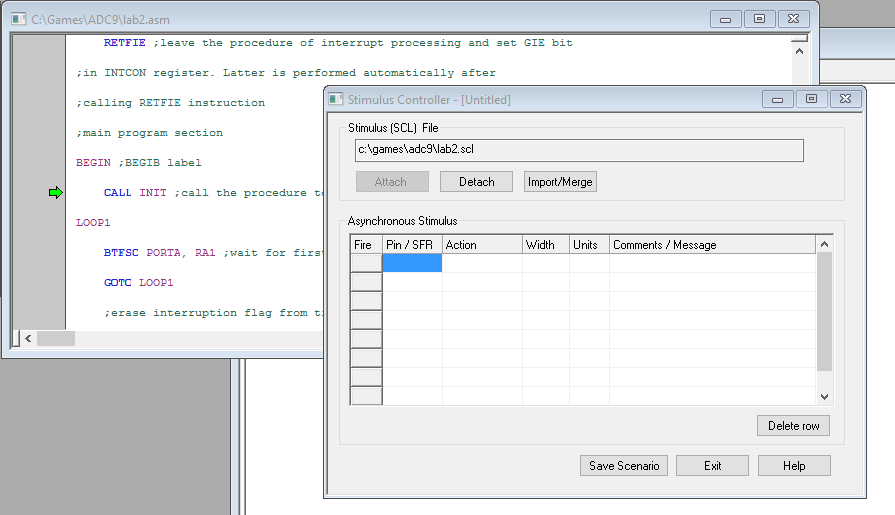
Выберите MPLAB SIM в качестве текущего отладчика (Debugger ->SelectTool -> 3MPLAB SIM). Затем выберите «Debugger->Reset->Processor» и попробуйте выполнить программу в пошаговом режиме (Debugger->StepInto).

Теперь нужно установить уровни сигналов на выходах сигналов микроконтроллеров. Сначала создадим stimulusworkbook, в которую мы поместим время и логику переноса. Перейдемв Debugger->SCL Generator->New Workbook. Выберемпервуювкладку “Pin /RegisterActions”. Введем данные, которые заданы в лабораторной работе.



Сохраним данную книгу. Затемнажмем “GenerateSCLFromWorkbook” чтобы создать файл стимула, например, lab2.scl и выйдем с программы.

Теперь приступим к симуляции. ПерейдемввкладкуDebugger->StimulusController->NewScenarioидобавимновыйсгенерированный файл(нажимаем “Attach” и выбираем файл lab2.scl).



Потом построим программу и нажмем F7 для пошаговоговыполнение программы.

**Вывод:** в ходе лабораторной работы я научился работать с таймером и прерыванием, принудительным моделированием штырей портов, комплексной откладкой программы.