



WORKING WITH SYNCHRONOUS IMPULSE IN THE MPLAB SIMULATOR

Abstract: Working with synchronous stimulus in MPLAB SIM: The paper presents the advantages of using synchronous stimuli. The process of defining stimuli using the build-in dialogs is shown and explained at every stage of the dialog. The paper covers all types of synchronous stimuli showing an example of how to define each type. Synchronous stimuli allow easier debugging the code with simulated external signals.

Author information:

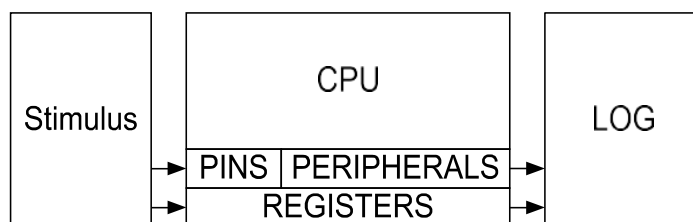
Daniel Denev
 PhD Student
 Faculty of Technical Sciences
 at Konstantin Preslavsky – University of Shumen
 ✉ slimshady33@abv.bg
 🌐 Bulgaria

Keywords:

Synchronous stimulus, Stimuli, MPLAB.

1. ВЪВЕДЕНИЕ

Базовата развойна среда MPLAB за контролерите на Microchip разполага със симулатор, чрез който приложната програма може да се изпълнява и настройва. По време на изпълнението на програмата в средата на симулатора могат да се симулират различни външни за контролера въздействия. Това се изпълнява чрез стимули, които могат да бъдат смяна на ниво или импулс върху входно/изходните (I/O) изводи на порт, смяна на стойностите в регистър със специални функции (SFR) или друга памет. На фиг. 1 е показана блоковата схема на основните модули на симулатора MPLAB SIM



Фиг. 1. Блокова схема на основните модули на симулатора MPLAB SIM

Стимулите могат да се зададат в определен цикъл на инструкцията или в определен момент в процеса на симулация. Също така стимулите могат да бъдат активирани и когато дадено условие е изпълнено.

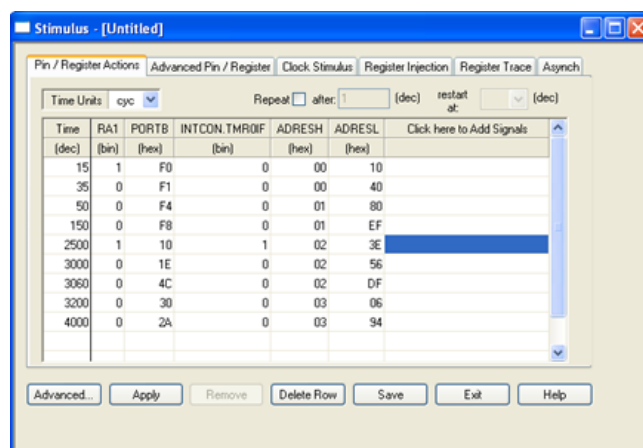
В развойната среда MPLAB има два вида стимули:

- синхронни – предварително зададена серия от промени;
- асинхронни – еднократна промяна, задействана от пусков бутон на програмата.
- За да се зададе кога и как външните стимули да се активират, се използват два начина – стимулен прозорец и файл със специфичен формат (SCL). Тези два метода могат да се използват за задаване както на синхронни, така и на асинхронни стимули.
- Чрез стимулите могат да се задават различни събития:

- ръчни тригери – това са промени в нивата, предизвикани от кликване на бутон с мишката. Те позволяват да се симулира действие като затваряне на
- ключ или подаване на импулс на извод;
- циклични сигнали – чрез тях се генерира поредица от импулси за определен период от време или докато се изпълнява програмата;
- редове от стойности, които могат да бъдат подавани на изводи, регистри или битове в регистри. Списък с редове от стойности може да бъде въведен, както в диалогов прозорец, така и от файл.

2. ЗАДАВАНЕ НА СИНХРОННИ СТИМУЛИ В MPLAB SIM

Синхронните стимули се задават в т. нар. SCL Workbook (фиг. 2)



Фиг. 2. Диалогов прозорец на SCL Workbook

SCL означава Stimulus Control Language, но с графичният интерфейс на MPLAB всички стимули могат да бъдат въведени лесно в графичните прозорци. След като събитията са описани в Workbook, те се компилират в SCL файл, който се зарежда в симулатора.

Диалоговият прозорец съдържа шест страници – една за асинхронни и пет за синхронни стимули.

Първата страница на синхронните стимули съдържа списък от времена и сигнали, които да бъдат подадени на изводите и регистрите.

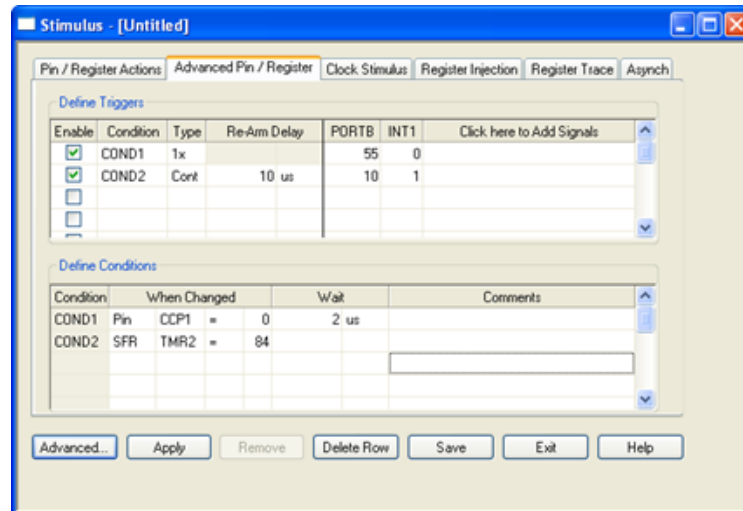
В едно и също време могат да се променят различни изводи и регистри на микроконтролер PIC18FXX2. Примерът от фиг. 2 показва поредица от събития, които се прилагат към:

- извод RA1;
- регистър PORTB;
- флага за заявка за прекъсване от Timer0;
- буфера на аналого-цифровия преобразувател.
- Стойностите в колоната, обозначена с PORTB, са 8 битови и се прилагат съответно към 8-те извода на порта.

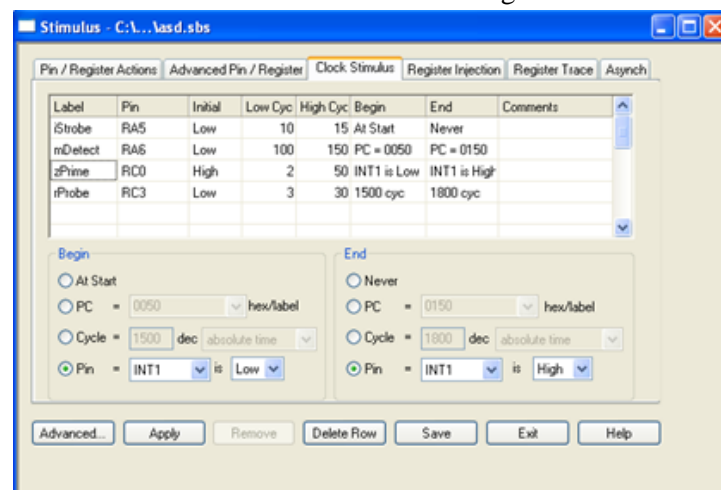
Следващата колона определя за всяко събитие дали да се активира прекъсване от Timer0. Колони ADRESH и ADRESL позволяват в буфера да бъде заредена 10 битова стойност.

В страницата Advanced Pin / Register (фиг. 3) се задават условията за активиране на стимулите. В този случай дадено условие трябва да бъде вярно, за да се активира стимул. Няколко събития могат да бъдат зададени на последователни редове. Те могат да бъдат „1x”, което означава, че ще се изпълняват само веднъж или “Continuous” - всеки път, когато условието е изпълнено.

В примера стойността 55H се зарежда в регистър PORTB 2 μ s след като на извод CCP1 се установи 0. Следващото събитие се изпълнява винаги, когато стойността TMR2 стигне 84H. Когато това се изпълни, в PORTB се зарежда 10H и на вход INT1 се установява единица. В колоната Re-Arm Delay е зададено 10 μ s изчакване. Това не позволява на събитието да се активира по-рано от 10 μ s от предходното изпълнение. В страница Clock Stimulus (фиг. 4) могат да се инициализират непрекъснати поредици от импулси.



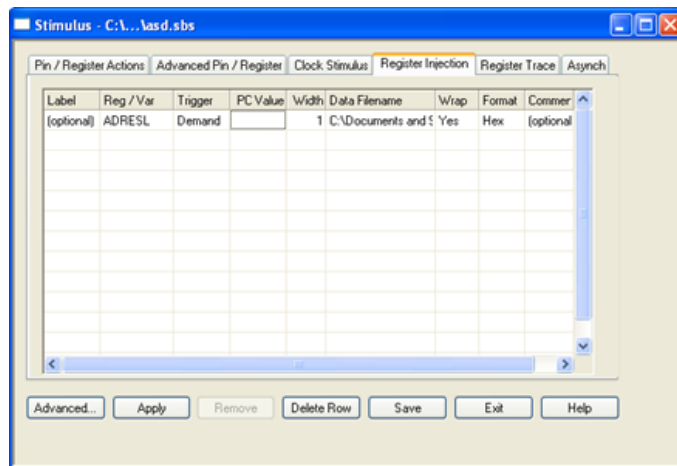
Фиг. 3. Advanced Pin / Register



Фиг. 4. Clock Stimulus

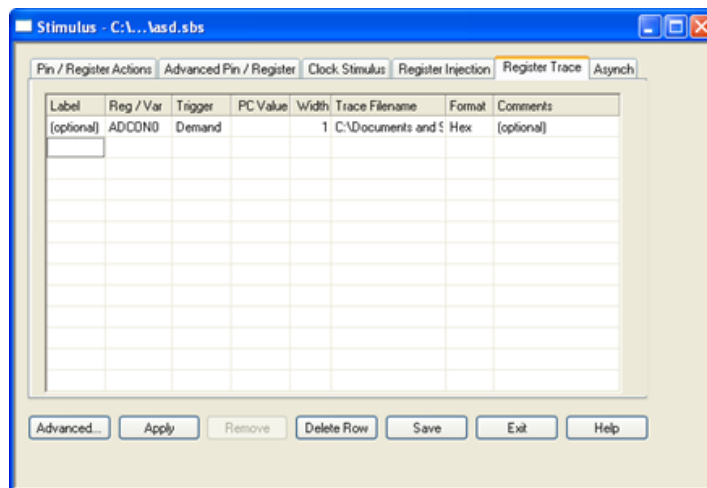
- На фиг. 4 са показани четири поредици с различни условия за активиране и деактивиране:
- iStrobe задава на RA5 периодичен сигнал с 10 машинни цикъла ниско ниво и 15 – високо. Този стимул е активен през цялото време на изпълнение на програмата.
 - mDetect задава на RA6 периодичен сигнал със 100 машинни цикъла ниско ниво и 150 – високо. Събитието няма да се изпълни, докато броячът не достигне стойност „50“, и ще се повтаря докато достигне стойност „150“.
 - zPrime задава начално високо ниво на RC0, последвано от 2 машинни цикъла ниско ниво и 50 – високо. Това продължава докато нивото на вход INT1 е ниско.
 - rProbe задава на RC3 периодичен сигнал с 3 машинни цикъла ниско ниво и 30 – високо. Активира се след 1500 цикъла от стартирането на програмата и е активен за 300 машинни цикъла.

В страницата Register Injection се задава зареждане на даден регистър със стойности, предварително записани в текстов файл. Във файла всяка стойност се въвежда на отделен ред.



Фиг. 5. Register Injection

В полето „register” се задава името на регистъра, чиито стойности ще се зареждат от избрания файл. В полето “trigger” се избира кога да се изпълни зареждането на регистъра – при заявка или когато РС достигне зададената стойност.



Фиг. 6. Register Trace

В полето “Data filename” се указва името на файла. В „Wrap” се избира циклично или еднократно зареждане. Форматът на данните от файла се задава в полето „Format”.

Страница “Register Trace” дава възможност прочетените стойностите от даден регистър да се запишат в текстов файл.

След като всички стимули са дефинирани, се генерира SCL файл, който може да се използва освен в MPLAB и в други развойни среди. На фиг. 7 е показан примерен SCL файл.

```

testbench for "pic18f452" is
begin
  process is
  begin
    wait for 15 ic;
    report "Stimulus actions after 15 ic";
    RA1 <= '1';
    wait for 20 ic;
    report "Stimulus actions after 35 ic";
    RA1 <= '0';
    wait for 15 ic;
    report "Stimulus actions after 50 ic";
    RA1 <= '0';
  end process;

  COND1: process is
  begin
    wait until CCP1 == '0';
    wait for 2 us;
    PORTB.RB <= B"01010101";
    INT1 <= '0';
    wait;
  end process COND1;

  istrobe: process is
  begin
    loop
      RA5 <= '0';
      wait for 10 ic;
      RA5 <= '1';
      wait for 15 ic;
    end loop;
    RA5 <= '1';
  end process istrobe;
end testbench;

```

Фиг. 7. Примерен SCL файл

3. ЗАКЛЮЧЕНИЕ

Развойната среда MPLAB на Microchip има значително подобрени възможности за задаване на синхронни стимули в сравнение с предходните версии.

Генерирането на единични импулси, импулсни поредици със зададена честота и коефициент на запълване и условията, при които те да бъдат изпълнявани, дава възможност за симулиране на външни за микроконтролерите събития, което намалява времето за настройка на приложенията.

References:

1. An introduction to MPLAB Integrated Development Environment, Microchip Web Seminars, 2004.
2. MPLAB ® IDE . 10, User's guide, Microchip Technology Inc. , 2007.
3. PIC18FXX2 Data Sheet. High Performance Enhanced Flash Microcontrollers with 10- bit A/D, Microchip Technology Inc. , 2001.