

## Debugging a Sample Project using Code Composer Studio on Board

When Code Composer Studio starts up, select *Project -> New CCS Project* from the menu as shown in Fig. 1.

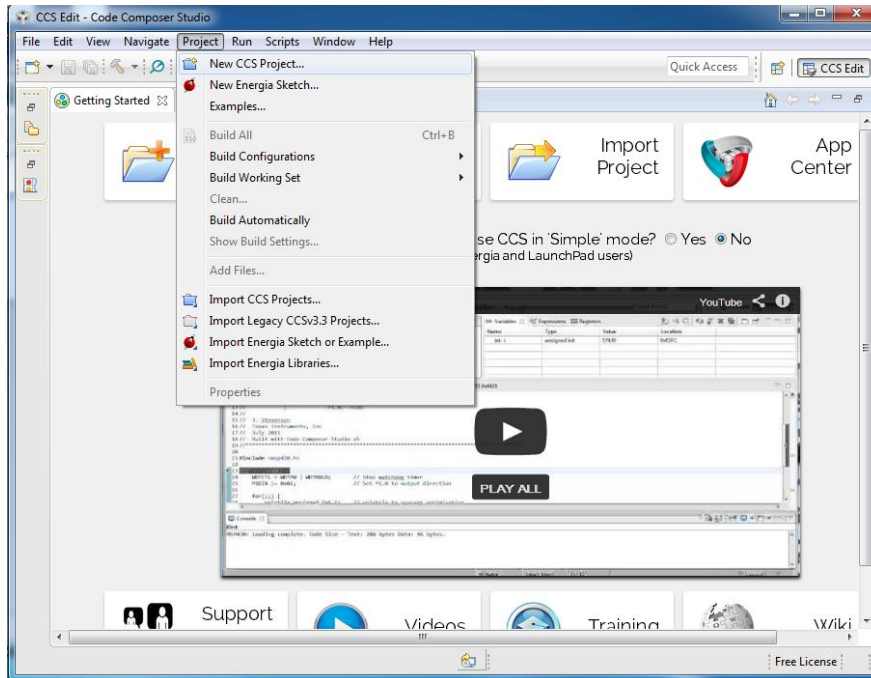


Figure 1: New Project creation menu

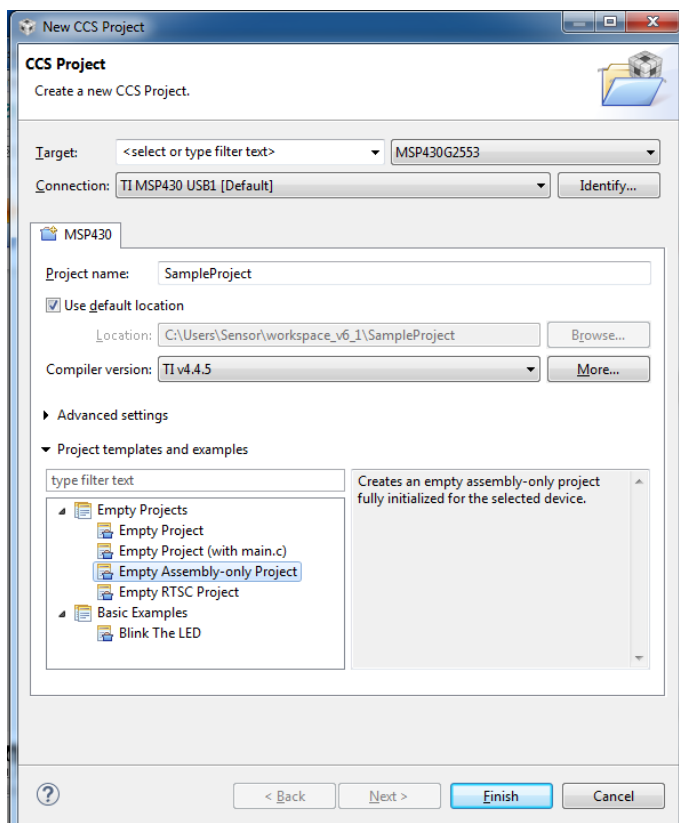


Figure 1: Project properties window

On the following window which is shown in Fig. 2, *Target* has to be selected as MSP430G2553. Give a proper name to the project and select the newest compiler version in *the compiler version* drop down list. Then from the *project templates and examples*, select the *Empty Assembly-only Project*.

Code Composer Studio will create **main.asm** source file for your assembly program. The structure of the main.asm file is shown below. During the experiments, you will place your assembly code to the section of the file which is commented as *;Main loop here* and the leave the rest of the file as is.

```

;-----
; MSP430 Assembler Code Template for use with TI Code Composer Studio
;
;-----
                .cdecls C,LIST,"msp430.h"          ; Include device header file
;-----
                .def      RESET                    ; Export program entry-point to
                                                ; make it known to linker.
;-----
                .text                               ; Assemble into program memory.
                .retain                             ; Override ELF conditional linking
                                                ; and retain current section.
                .retainrefs                         ; And retain any sections that have
                                                ; references to current section.
;-----
RESET          mov.w    #__STACK_END,SP           ; Initialize stackpointer
StopWDT        mov.w    #WDTPW|WDTHOLD,&WDTCTL    ; Stop watchdog timer

;-----
; Main loop here
;-----

;-----
; Stack Pointer definition
;-----
                .global  __STACK_END
                .sect    .stack

;-----
; Interrupt Vectors
;-----
                .sect    ".reset"                  ; MSP430 RESET Vector
                .short   RESET

```

Write the following assembly code to the place left as main loop on your main.asm file.

```

SetupP1        bis.b    #001h,&P1DIR              ; P1.0  output
;
Mainloop       xor.b    #001h,&P1OUT              ; Toggle P1.0
Wait           mov.w    #050000,R15               ; Delay to R15
L1             dec.w    R15                        ; Decrement R15
              jnz       L1                        ; Delay over?
              jmp       Mainloop                  ; Again
;

```

After placing your code, the resulting asm file should look like this:

```
;-----  
; MSP430 Assembler Code Template for use with TI Code Composer Studio  
;-----  
        .cdecls C,LIST,"msp430.h"          ; Include device header file  
  
;-----  
        .def      RESET                    ; Export program entry-point to  
                                           ; make it known to linker.  
;-----  
        .text                               ; Assemble into program memory.  
        .retain                               ; Override ELF conditional linking  
                                           ; and retain current section.  
        .retainrefs                         ; And retain any sections that have  
                                           ; references to current section.  
  
;-----  
RESET    mov.w    #__STACK_END,SP          ; Initialize stackpointer  
StopWDT  mov.w    #WDTPW|WDTHOLD,&WDTCTL    ; Stop watchdog timer  
  
;-----  
  
SetupP1   bis.b    #001h,&P1DIR             ; P1.0  output  
                                           ;  
Mainloop  xor.b    #001h,&P1OUT             ; Toggle P1.0  
Wait      mov.w    #050000,R15             ; Delay to R15  
L1        dec.w    R15                     ; Decrement R15  
          jnz      L1                     ; Delay over?  
          jmp      Mainloop                ; Again  
  
;-----  
  
;-----  
; Stack Pointer definition  
;-----  
        .global  __STACK_END  
        .sect    .stack  
  
;-----  
; Interrupt Vectors  
;-----  
        .sect    ".reset"                  ; MSP430 RESET Vector  
        .short   RESET
```

### Disabling the Compiler Optimizations

During the build process, compiler may see portions of your code unnecessary and remove them or try to optimize your code by taking other actions. In order to see the results of your code without any compiler modifications, you need to disable the compiler optimizing your code. You can do this by right clicking on your project and selecting the *Properties*. In the Properties Window, from the *Build* tab, select *MSP430 Compiler* and then *Optimization* section. In optimization section, select *optimization level* as **off** and click *ok*.

## Building

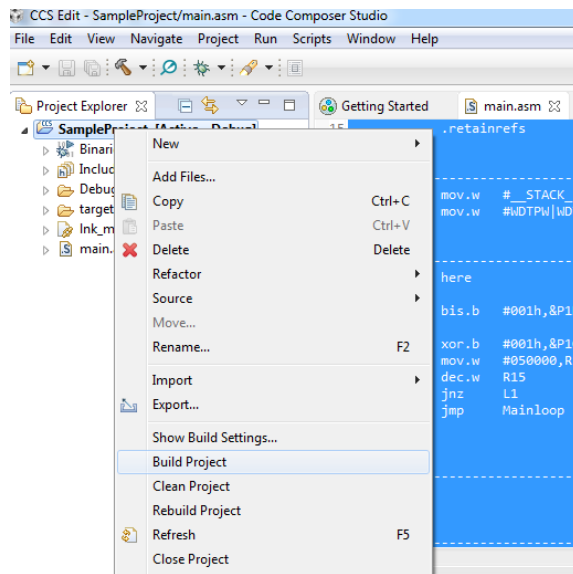


Figure 3: Building the project

Before loading your program to the board, you firstly need to build your program by right clicking on the project and selecting *Build Project* from the menu as shown in Fig. 3.

## Debugging and Loading

If build process does not complain about any errors, then your program is ready to be loaded to the board. If your board is not connected to the PC via USB connection, you should connect it now. In order to load your program and debug it, press F11 or select *Run -> Debug* from the menu. Code Composer Studio will load your program to the board and you will see the Debugging view right now as shown in the Fig. 4.

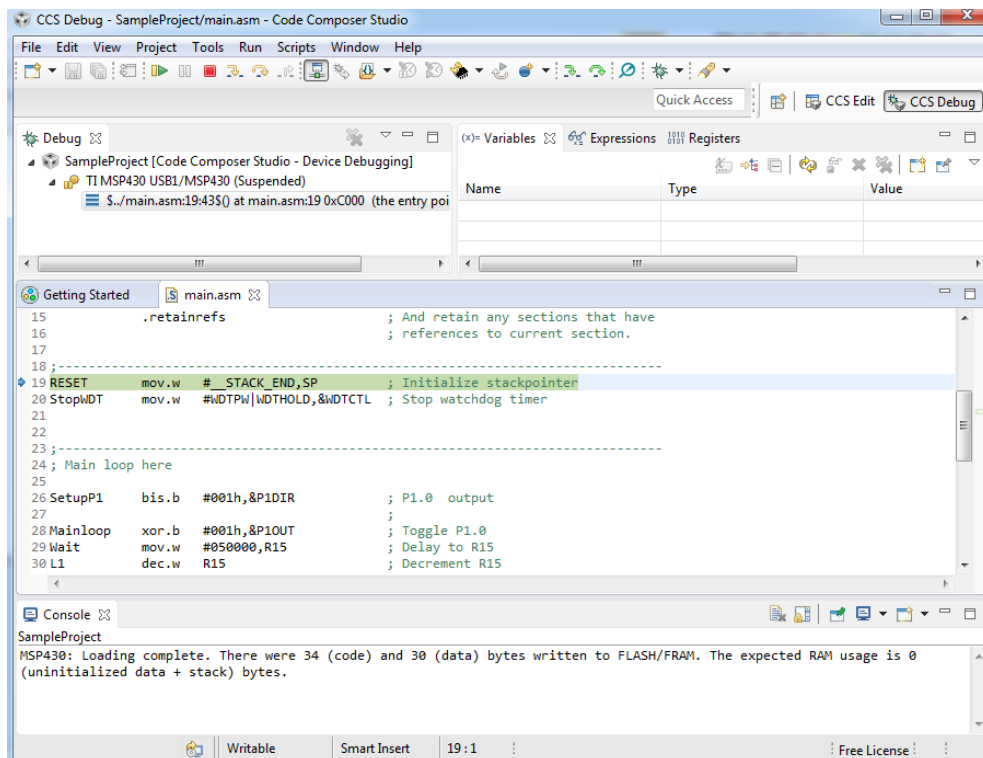


Figure 4: Debugging view

When the program is loaded to the board, then we have chance run the assembly code step by step by either pressing F5 on the keyboard or using the debugging menu (i.e., step into, step over) as shown in Fig. 5.

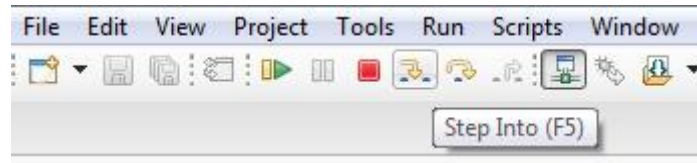


Figure 5: Debugging menu

You can use the step into and step over to execute the current assembly code line by line. The effect of these commands and your assembly code can be seen in the registers of the MSP430. You can view the contents of the all registers by Registers tab of the Debugging View which is shown in Fig. 6.

(x)= Variables Expressions 1010 0101 Registers		
Name	Value	Description
Core Registers		Core Registers
1010 0101 PC	0xC00E	Core
1010 0101 SP	0x0400	Core
1010 0101 SR	0x0000	Core
1010 0101 R3	0x0000	Core
1010 0101 R4	0x9959	Core
1010 0101 R5	0xFFB6	Core
1010 0101 R6	0x1EF7	Core
1010 0101 R7	0xDB7F	Core
1010 0101 R8	0x36A7	Core
1010 0101 R9	0x75EF	Core
1010 0101 R10	0xFF93	Core
1010 0101 R11	0xE9EC	Core
1010 0101 R12	0x0000	Core
1010 0101 R13	0xFD90	Core
1010 0101 R14	0x0000	Core
1010 0101 R15	0x4EF6	Core
Special_Function		
ADC10		
System_Clock		
Comparator_A		
Flash		
Port_1_2		
Port_3_4		

Figure 6: Registers Tab

When you click resume on debugging menu, your program starts running on the board. The debugging view can be terminated by simply clicking on Terminate on debugging menu or pressing Ctrl+F2.