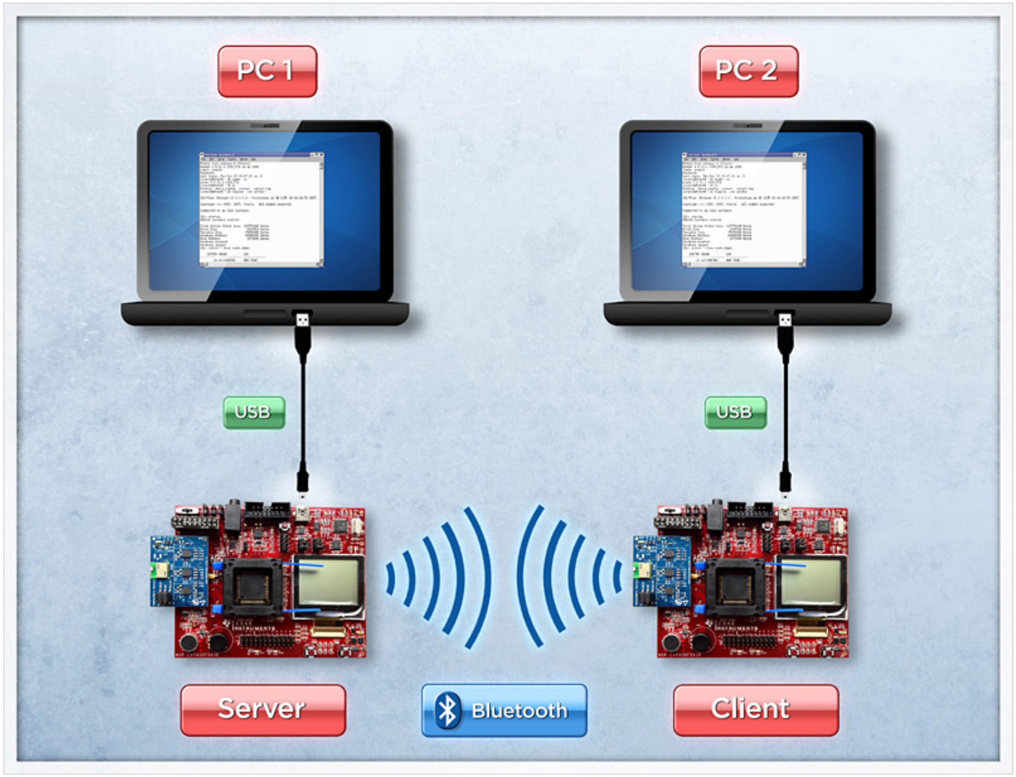
CC256x MSP430 TI Bluetooth Stack

Contents

Demo Overview

This demo allows users to evaluate TI's CC256x Bluetooth device by using the PAN1323EMK kit and the MSP-EXP430F5438 board. The CC256x+MSP430 Bluetooth sample applications are provided to enable a rich out-of-box experience to the user. These applications allow the user to use a console to send Bluetooth commands, setup a Bluetooth Device to accept connections, connect to a remote Bluetooth device, and communicate over Bluetooth.



Requirements

Hardware

- CC256xQFNEM (http://www.ti.com/tool/cc256xqfnem)

- MSP-EXP430F5438 Experimenter Board (http://www.ti.com/tool/msp-exp430f5438)

Note 1: For downloading the sample demos to the EVM, you require MSP430 USB Debugging Interface (http://www.ti.com/tool/msp-fet430uif). This needs to be purchased separately.

Note 2: Hardware depends on the application, specific demos will have detailed hardware configurations. But all the APPS should work with the hardware listed above.

Software

- CC256x+MSP430 Bluetooth Software - Contains the Sample Applications - Download (http://www.ti.com/tool/tibluetoothstack-sdk)
- Lite FET-Pro430 Tool - To flash the example code in the MSP430 chip - Download (http://www.elprotronic.com/download.html)

Hardware Setup

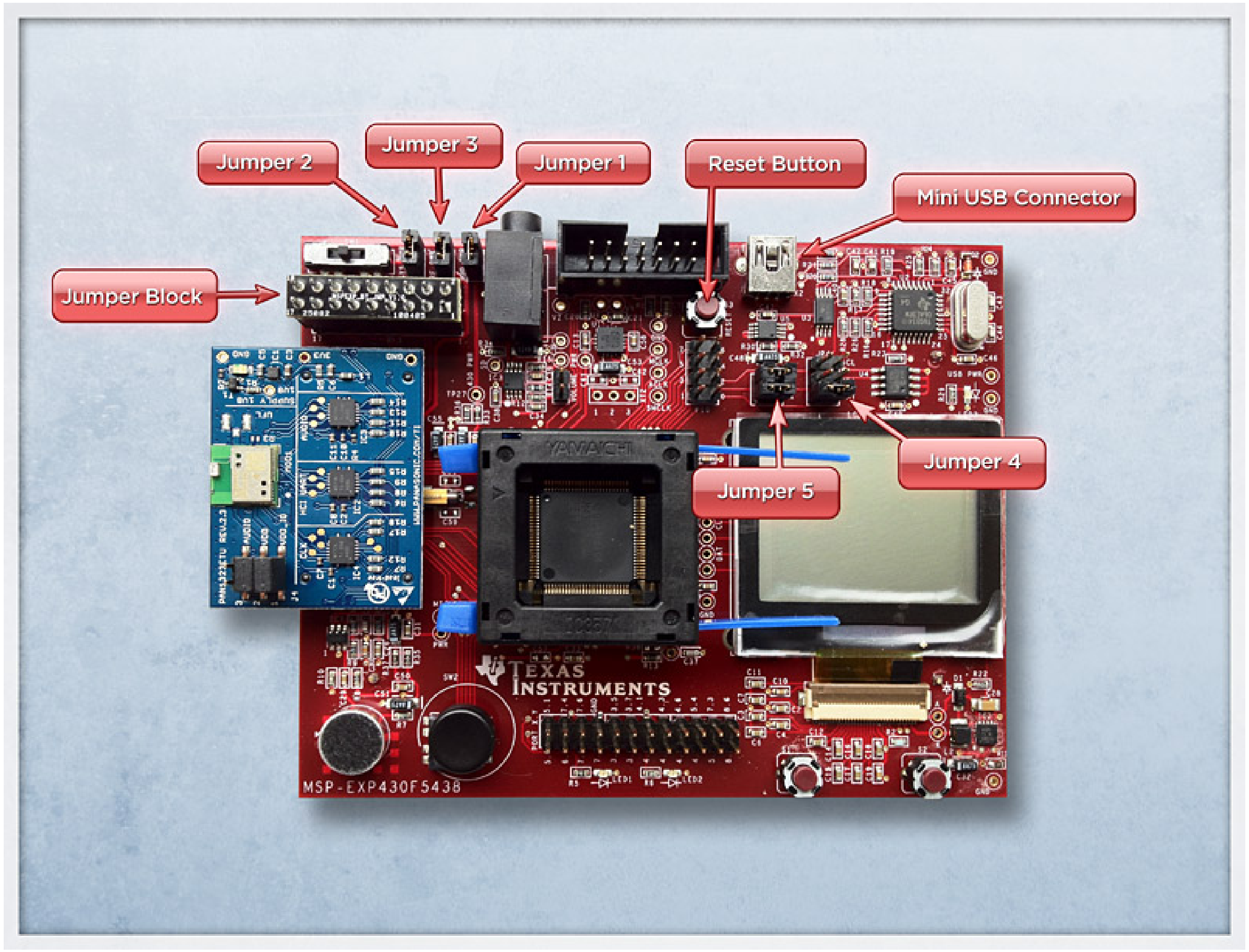
MSP430F5438 Experimenter's Board

a) Connect the CC256x board PAN1323ETU (http://www.panasonic.com/industrial/electronic-components/rf-modules/bluetooth/pan1323etu.aspx)onto the MSP-EXP430F5438 board (http://www.ti.com/tool/msp-exp430f5438) using the RF1/RF2 connectors (RF EVM connectors)

b) Place the included jumper block on RF3 connector.

c) Make sure that JP1, JP2, JP3, JP4 and JP5 are as shown in the picture below.

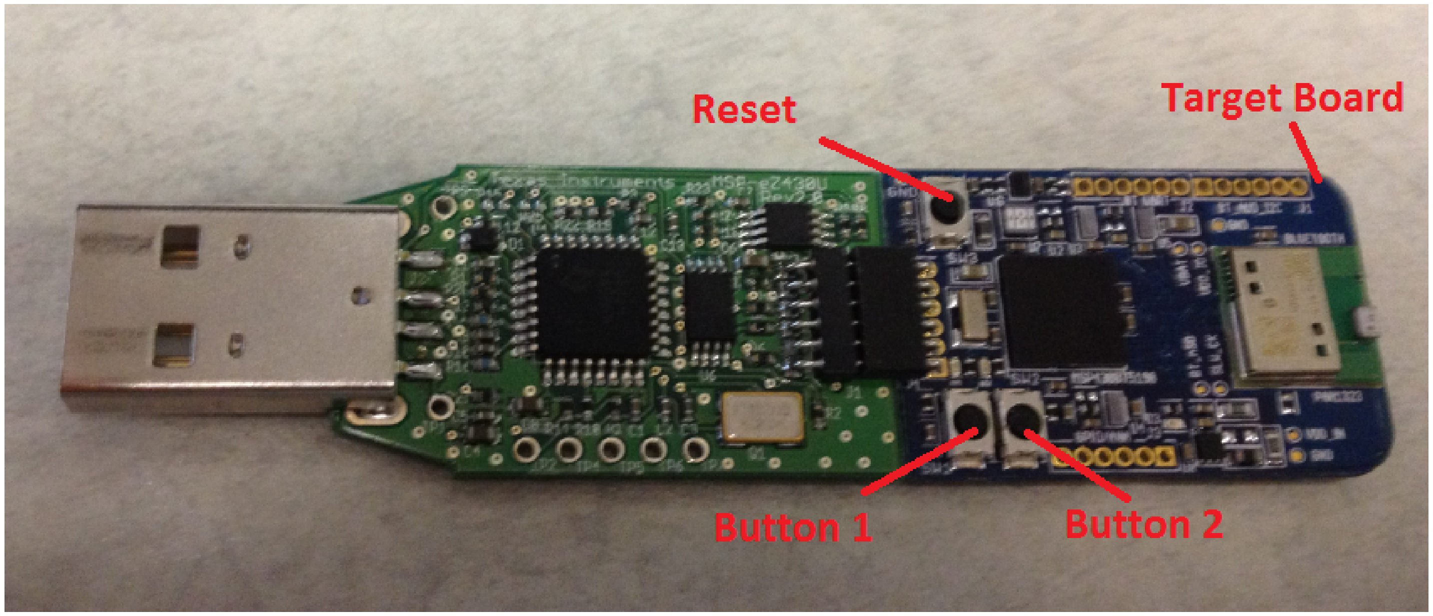
d) Make sure the other default jumper configuration remains on your MSP-EXP430F5438 board



EZ430-RF256x

a) On Windows-7, please run the windows applications in administrator mode.

b) Connect the RF256xT Target Board to the USB board. The picture below shows the connected boards.



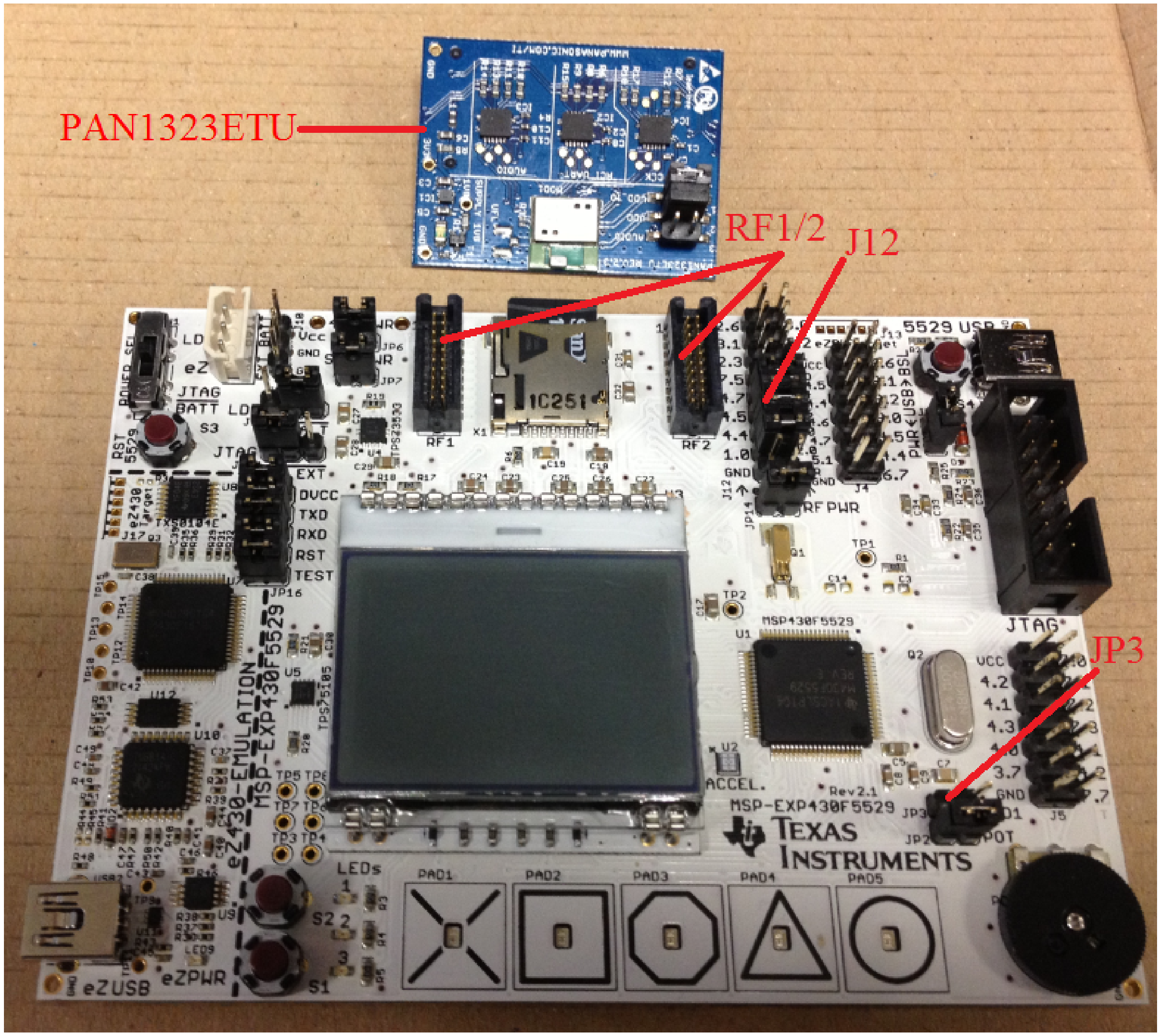
MSP430F5529 Experimenter’s Board

a) Connect the CC256x board PAN1323ETU (http://www.panasonic.com/industrial/electronic-components/rf-modules/bluetooth/pan1323etu.aspx)onto the MSP-EXP430F5529 board (http://www.ti.com/tool/msp-exp430f5529) using the RF1/RF2 connectors (RF EVM connectors)

b) Remove the JP3 Jumper.

c) In J12, make sure Pin 3 to Pin 4 (P1.0 to P2.0), Pin 5 to Pin 6 (P4.4 to P4.0), Pin 7 to Pin 8 (P4.5 to P4.6), Pin 9 to Pin 10 (P4.7 to P2.4) are connected as shown in the picture.

d) Make sure the other default jumper configuration remains on your MSP-EXP430F5529 board



CC256XB Information

To use the CC256XB module, the patch CC256XB.h has to be included. To make sure this patch is included, in BTPSVend.C add a line

```
#define __SUPPORT_CC256XB_PATCH__
```

This will make sure that the CC256XB patch is loaded and the CC256XB module can be used without issues.

Software Setup

Downloading and Installing the Software

The Bluetooth SDK can be downloaded from here (http://www.ti.com/tool/tibluetoothstack-sdk). You will be prompted for a TI login when you try to download the SDK. If you don't already have a TI login you will be asked to create one. After creating a TI login you will need to complete the export approval form. After completing the form a request will be submitted to TI for approval. After the request is approved you will see a confirmation on the next page and be provided with a link to download the software. Click the "Download" button to start the download.

After the download completes run the setup file (e.g. CC256x_MSP430_Bluetopia_SDK_vx_x_Setup.exe). The setup process will ask you to accept the TI Bluetooth Stack Clickwrap License Agreement. After accepting the license agreement the SDK will be installed to "C:\TI\Connectivity\CC256X BT\CC256x MSP430 Bluetopia SDK\". The SDK can be accessed using Start / Programs / Texas Instruments / CC256x MSP430 Bluetopia SDK v x.x.

You now have access to sample applications' source code as well as binaries to flash directly to your MSP430 board.

Flashing the Bluetooth Code

Customers can use MSP430 Flasher with MSP430DLLV3 after updating the firmware of FET430 USB Emulator. For instructions visit MSP430 FET430 setup

You can use the included batch file Flash_CC256x_MSP430_Bluetopia.bat to flash the software The flashing tool (MSP430Flasher.exe) and documentation (MSP430Flasher_Doc.pdf) are available under Tools folder at the install location.

Once MSP430Flasher tool is ready to flash the code, you can use the included batch file or the shortcut Flash_CC256x_MSP430_Bluetopia.bat to flash the BR/EDR SPP sample application to the board.

Platform Specific

EZ430

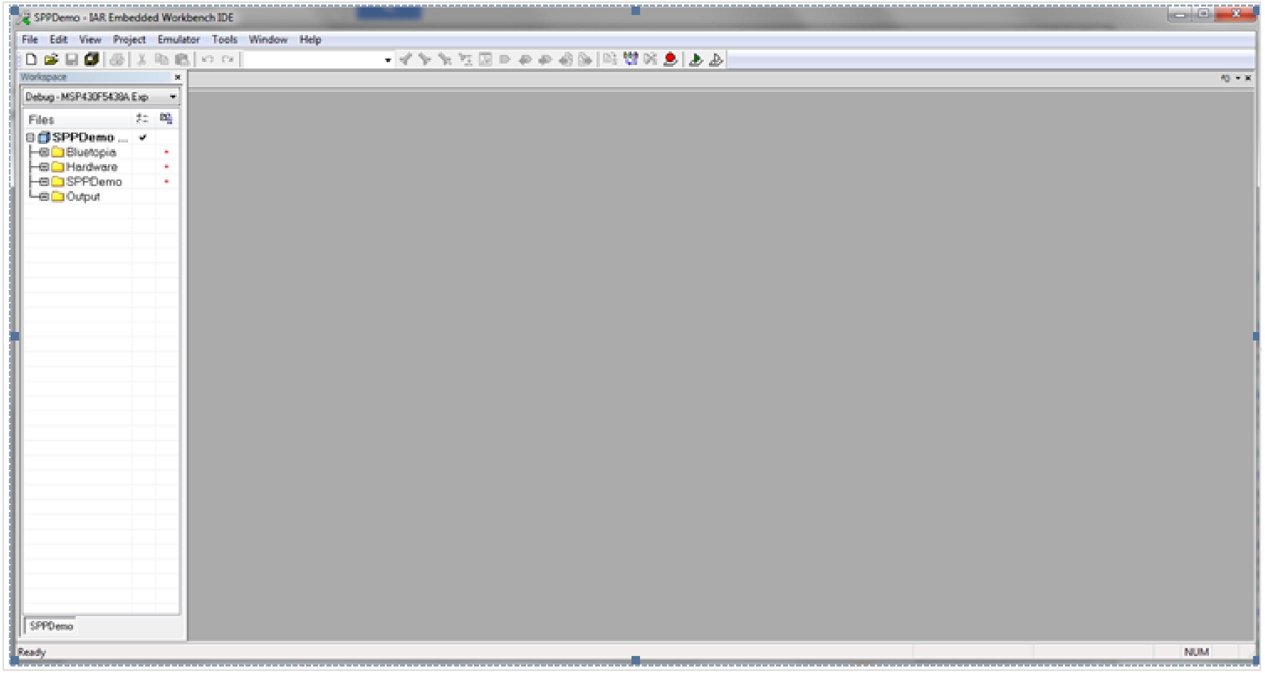
All the demo apps have support for EZ430. Here is the software configurations for the demos:

IAR

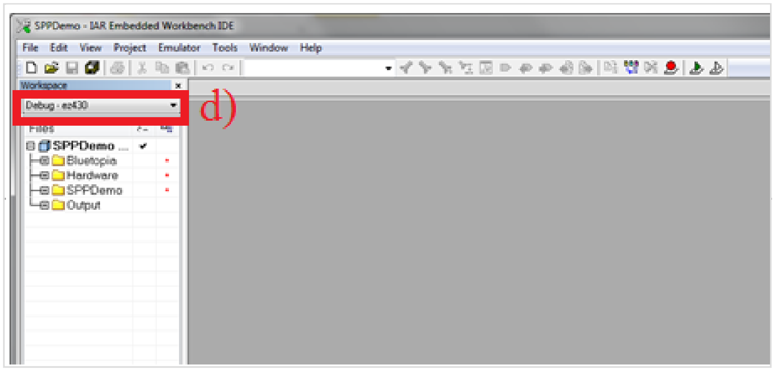
a) Go to TI's_Bluetooth_Stack_Version-Number_MSP430\MSP430_Experimentor\Samples. Here you will see a list of samples.

b) Select the demo that you want to load on the device. For our example we will use SPPDemo

c) Navigate to Projects->IAR and select SPPDemo.eww . The IDE should open.

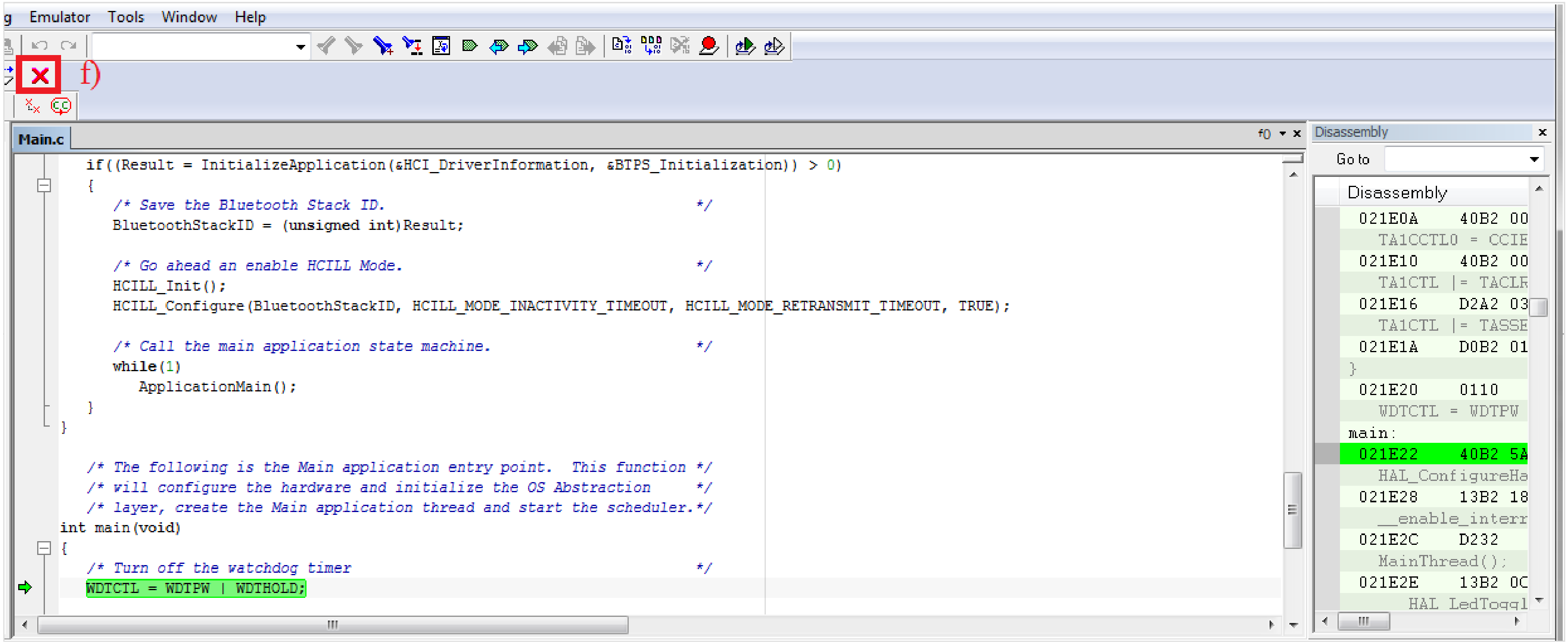


d) In the workspace tab, the configurations to load the device should be present. From the dropdown menu, select either debug or release configurations for EZE-430. In our case we select, Debug-EZE 430.


SPPDemo - IAR Embedded Workbench IDE
File Edit View Project Emulator Tools Window Help
Workspace
Debug - ez430
d)
Files
SPPDemo
Bluetopia
Hardware
SPPDemo
Output

e) Select download and debug(make sure that the EZE-430 device is conencted) by either clicking on the Play icon or selecting Project->Download and Debug.

f) Let the IDE debug and load the software on the device. This may take 5-10 minutes. Once it is done debugging, click on the X on the IDE to stop debugging.



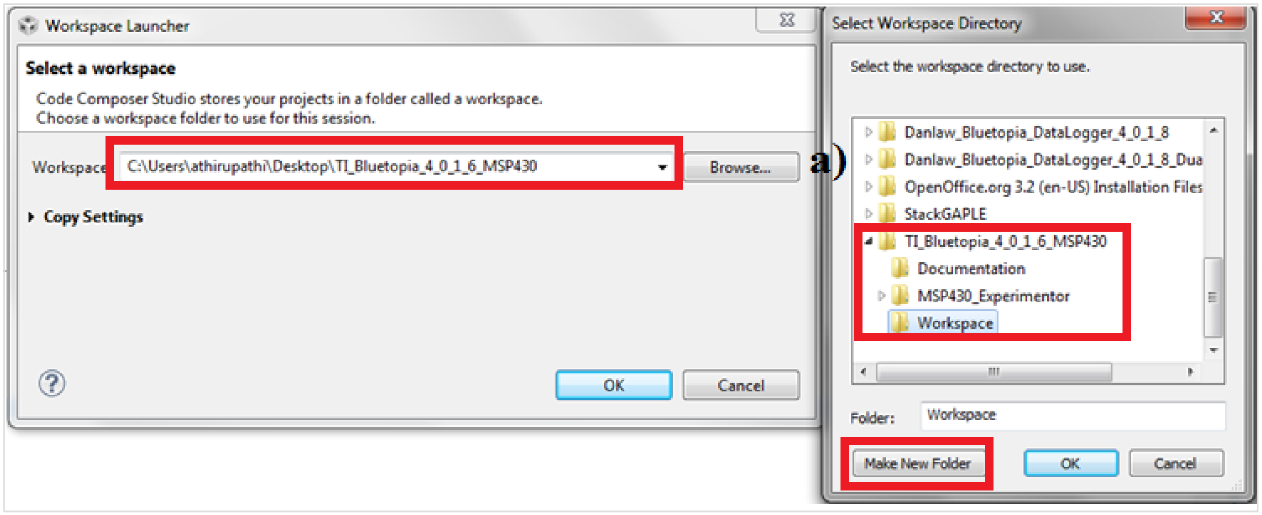
```
    if((Result = InitializeApplication(&HCI_DriverInformation, &BTPS_Initialization)) > 0)
    {
       /* Save the Bluetooth Stack ID.                                   */
       BluetoothStackID = (unsigned int)Result;

       /* Go ahead an enable HCILL Mode.                                 */
       HCILL_Init();
       HCILL_Configure(BluetoothStackID, HCILL_MODE_INACTIVITY_TIMEOUT, HCILL_MODE_RETRANSMIT_TIMEOUT, TRUE);

       /* Call the main application state machine.                       */
       while(1)
          ApplicationMain();
    }
}

   /* The following is the Main application entry point.  This function */
   /* will configure the hardware and initialize the OS Abstraction     */
   /* layer, create the Main application thread and start the scheduler.*/
int main(void)
{
   /* Turn off the watchdog timer                                        */
   WDTCTL = WDTPW | WDTHOLD;
```

```
Disassembly
 021E0A      40B2 00
   TA1CCTL0  = CCIE
 021E10      40B2 00
   TA1CTL  |= TACLR
 021E16      D2A2 03
   TA1CTL  |= TASSE
 021E1A      D0B2 01
}
 021E20      0110
   WDTCTL  = WDTPW
main:
 021E22      40B2 5A
   HAL_ConfigureHa
 021E28      13B2 18
   __enable_interr
 021E2C      D232
   MainThread();
 021E2E      13B2 0C
      HAL_LedToggl
```

g) The software is now ready to use. To begin using the profile disconnect and reconnect the EZE-430 after flashing.

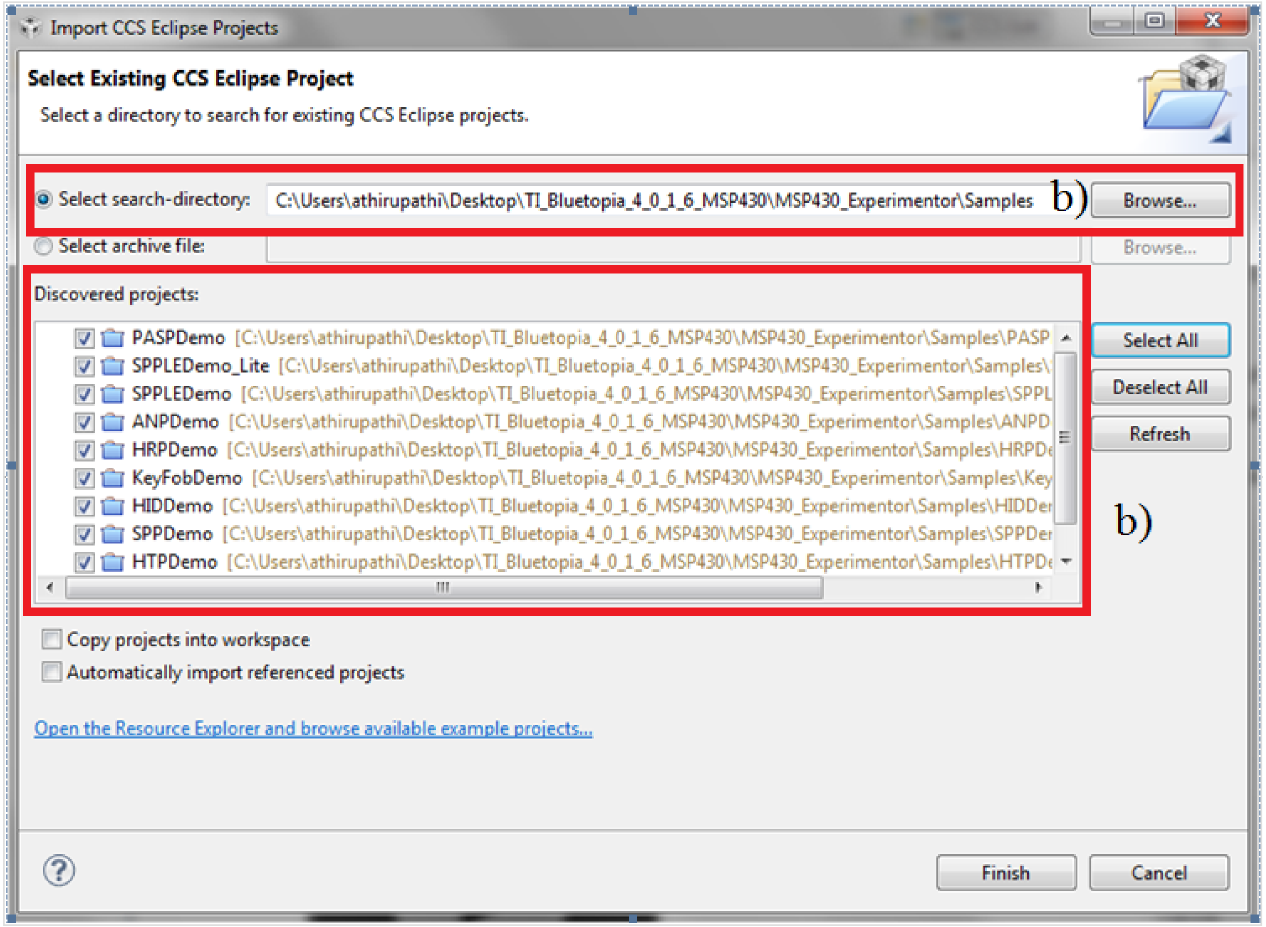
CCS

a) Open Code Composer Studio 5. Select File->Switch Workspace. Navigate to the folder where the TI's Bluetooth Stack Samples are located and make a new folder for the workspace.


Workspace Launcher
Select a workspace
Code Composer Studio stores your projects in a folder called a workspace.
Choose a workspace folder to use for this session.
Workspace C:\Users\athirupathi\Desktop\TI_Bluetopia_4_0_1_6_MSP430
Browse...
Copy Settings
OK
Cancel
a)
Select Workspace Directory
Select the workspace directory to use.
Danlaw_Bluetopia_DataLogger_4_0_1_8
Danlaw_Bluetopia_DataLogger_4_0_1_8_Dua
OpenOffice.org 3.2 (en-US) Installation Files
StackGAPLE
TI_Bluetopia_4_0_1_6_MSP430
Documentation
MSP430_Experimentor
Workspace
Folder: Workspace
Make New Folder
OK
Cancel

CCS will now exit and reload the software. Each time you load a project it is preferable to create a new workspace for it.

b) Select Project-> Import Existing CCS Eclipse project. In the select Auto search directory point to the TI's_Bluetooth_Stack_Version-Number_MSP430\MSP430_Experimentor\Samples folder. CCS will recognize all available profiles. Select all profiles and hit finish. (If you prefer you can selectively load the profiles you want).



Select search-directory: C:\Users\athirupathi\Desktop\TI_Bluetopia_4_0_1_6_MSP430\MSP430_Experimentor\Samples

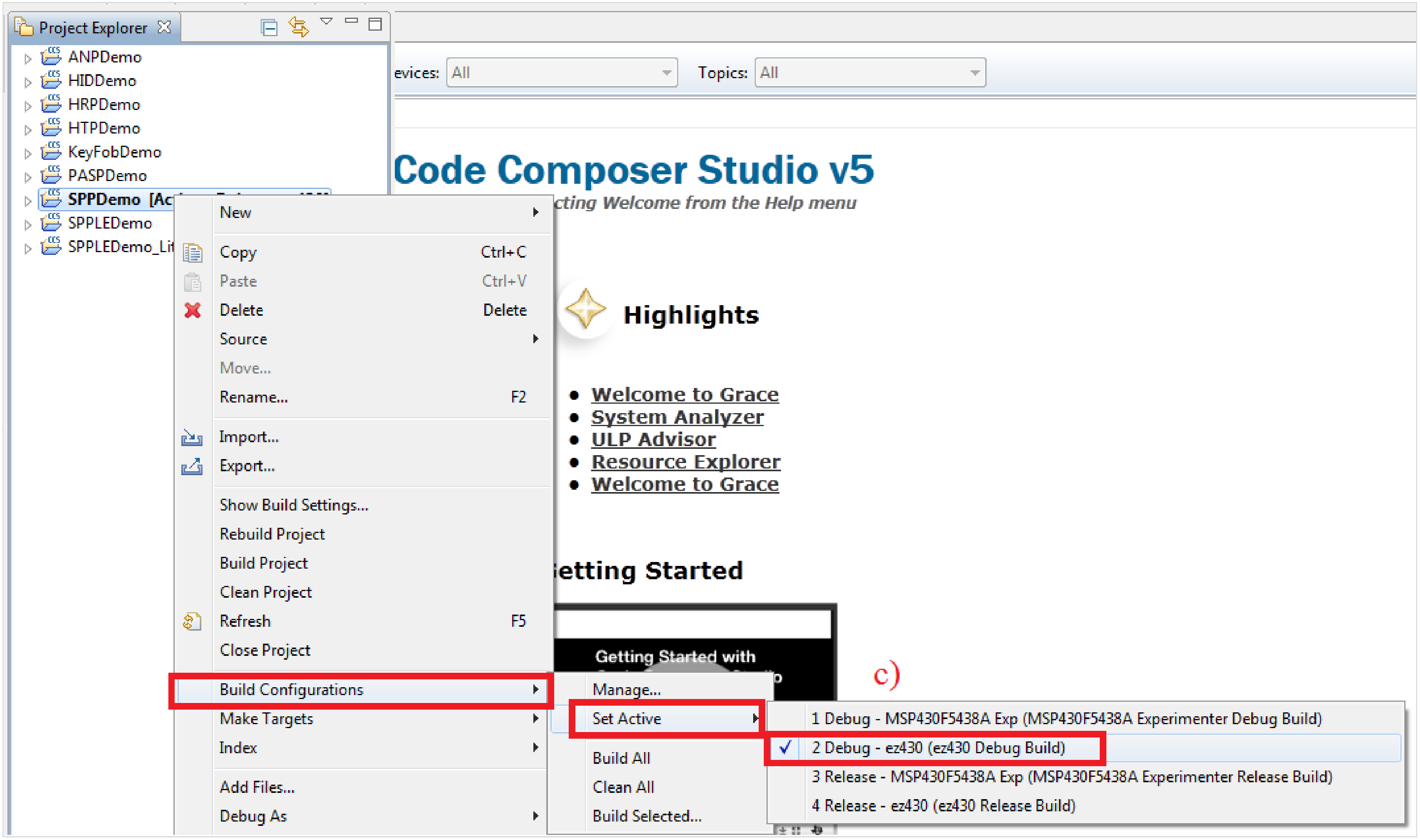
Discovered projects:

- [✓] PASPDemo [C:\Users\athirupathi\Desktop\TI_Bluetopia_4_0_1_6_MSP430\MSP430_Experimentor\Samples\PASP
- [✓] SPPLEDemo_Lite [C:\Users\athirupathi\Desktop\TI_Bluetopia_4_0_1_6_MSP430\MSP430_Experimentor\Samples\
- [✓] SPPLEDemo [C:\Users\athirupathi\Desktop\TI_Bluetopia_4_0_1_6_MSP430\MSP430_Experimentor\Samples\SPPL
- [✓] ANPDemo [C:\Users\athirupathi\Desktop\TI_Bluetopia_4_0_1_6_MSP430\MSP430_Experimentor\Samples\ANPD
- [✓] HRPDemo [C:\Users\athirupathi\Desktop\TI_Bluetopia_4_0_1_6_MSP430\MSP430_Experimentor\Samples\HRPD
- [✓] KeyFobDemo [C:\Users\athirupathi\Desktop\TI_Bluetopia_4_0_1_6_MSP430\MSP430_Experimentor\Samples\Key
- [✓] HIDDemo [C:\Users\athirupathi\Desktop\TI_Bluetopia_4_0_1_6_MSP430\MSP430_Experimentor\Samples\HIDDe
- [✓] SPPDemo [C:\Users\athirupathi\Desktop\TI_Bluetopia_4_0_1_6_MSP430\MSP430_Experimentor\Samples\SPPDe
- [✓] HTPDemo [C:\Users\athirupathi\Desktop\TI_Bluetopia_4_0_1_6_MSP430\MSP430_Experimentor\Samples\HTPD

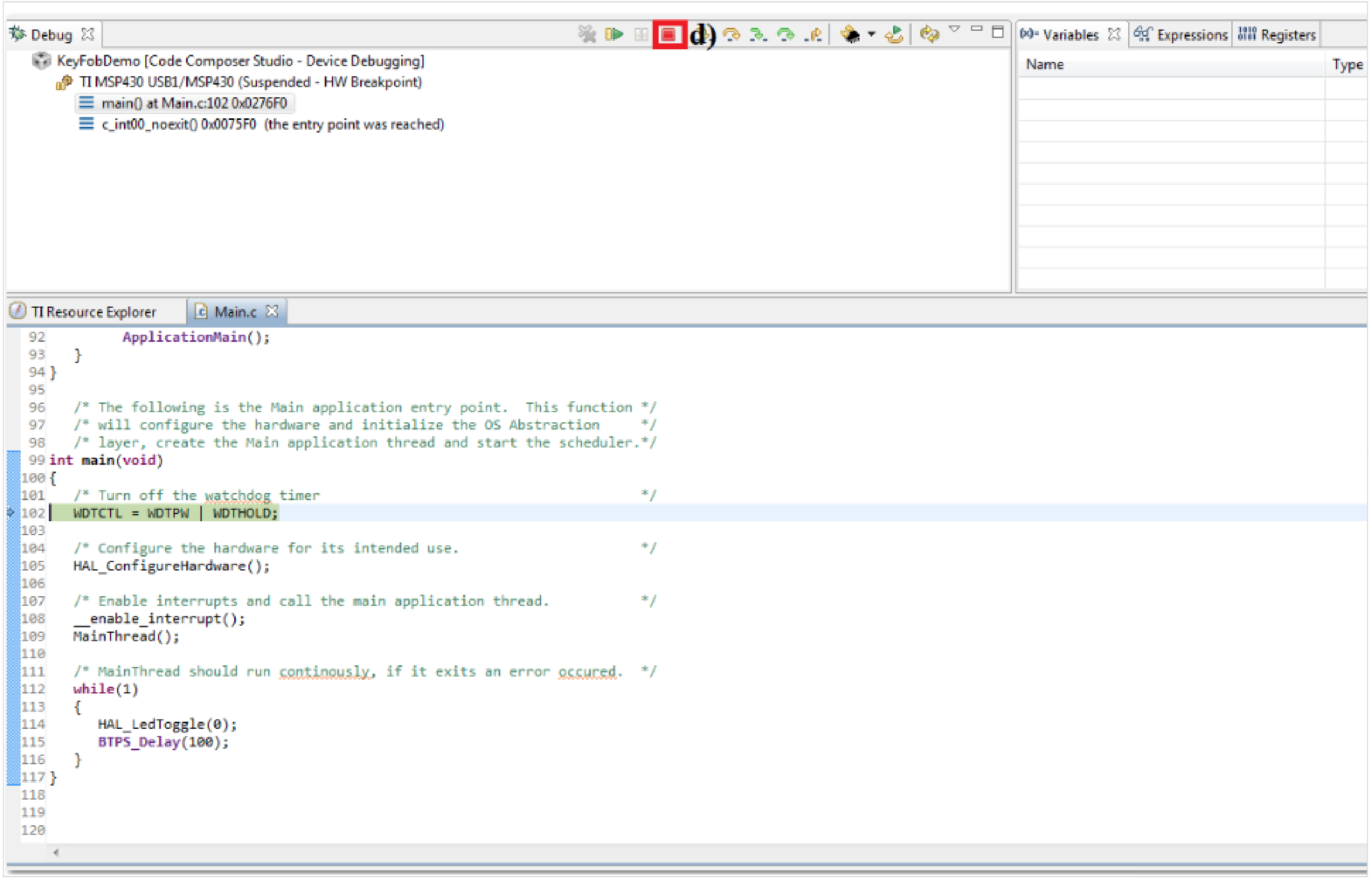
- [ ] Copy projects into workspace
- [ ] Automatically import referenced projects

Open the Resource Explorer and browse available example projects...

c) The projects should load on the left in Project Explorer. Click on the profile that you want to load. By default it should select the Debug, MSP 430 configuration for the profile. To change the configuration, right click on the profile-> Build Configurations -> Set Active. Select either the debug or release configuration for EZE-430.



d) To begin flashing the device, select from the toolbar -> Debug as -> Code composer session. Let the IDE debug and load the software on the device. This may take 5-10 minutes. Once it is done debugging, click on the stop button on the IDE to stop debugging.



```
KeyFobDemo [Code Composer Studio - Device Debugging]
  TI MSP430 USB1/MSP430 (Suspended - HW Breakpoint)
    main() at Main.c:102 0x0276F0
    c_int00_noexit() 0x0075F0  (the entry point was reached)
```

```
         ApplicationMain();
   }
}

   /* The following is the Main application entry point.  This function */
   /* will configure the hardware and initialize the OS Abstraction     */
   /* layer, create the Main application thread and start the scheduler.*/
int main(void)
{
   /* Turn off the watchdog timer                                        */
   WDTCTL = WDTPW | WDTHOLD;

   /* Configure the hardware for its intended use.                       */
   HAL_ConfigureHardware();

   /* Enable interrupts and call the main application thread.            */
   __enable_interrupt();
   MainThread();

   /* MainThread should run continously, if it exits an error occured.  */
   while(1)
   {
      HAL_LedToggle(0);
      BTPS_Delay(100);
   }
}
```

e) The software is now ready to use. To begin using the profile disconnect and reconnect the EZE-430 after flashing.

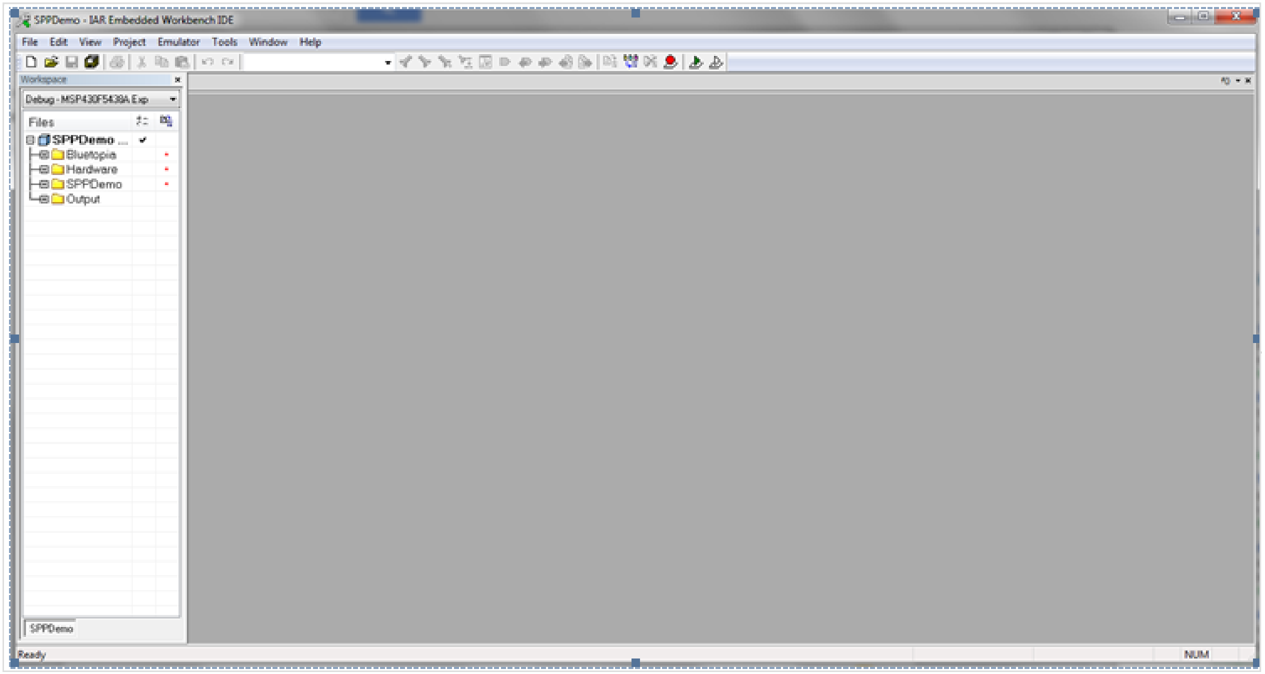
MSP430F5438/MSP430F5529

IAR

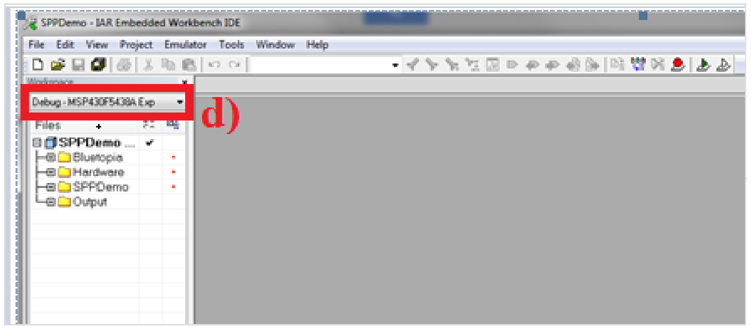
a) Go to TI's_Bluetooth_Stack_Version-Number_MSP430\MSP430_Experimentor\Samples. Here you will see a list of samples.

b) Select the demo that you want to load on the device. For our example we will use SPPDemo

c) Navigate to Projects->IAR and select SPPDemo.eww . The IDE should open.

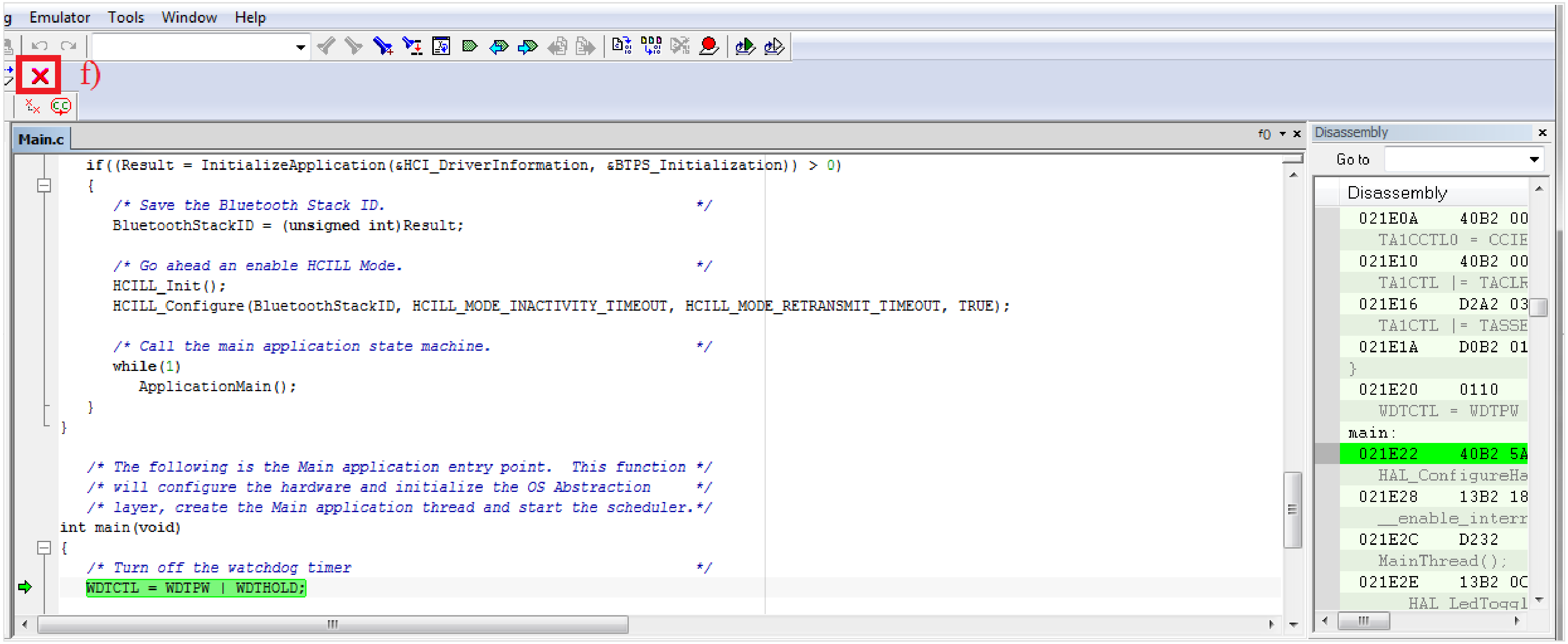


d) In the workspace tab, the configurations to load the device should be present. From the dropdown menu, select either debug or release configurations for EZE-430. In our case we select, Debug-MSP430F5438A.


SPPDemo - IAR Embedded Workbench IDE
File Edit View Project Emulator Tools Window Help
Workspace
Debug - MSP430F5438A Exp
d)
Files
SPPDemo ...
Bluetopia
Hardware
SPPDemo
Output

e) Select download and debug(make sure that the MSP430 device is conencted) by either clicking on the Play icon or selecting Project->Download and Debug.

f) Let the IDE debug and load the software on the device. This may take 5-10 minutes. Once it is done debugging, click on the X on the IDE to stop debugging.



```
    if((Result = InitializeApplication(&HCI_DriverInformation, &BTPS_Initialization)) > 0)
    {
       /* Save the Bluetooth Stack ID.                                    */
       BluetoothStackID = (unsigned int)Result;

       /* Go ahead an enable HCILL Mode.                                  */
       HCILL_Init();
       HCILL_Configure(BluetoothStackID, HCILL_MODE_INACTIVITY_TIMEOUT, HCILL_MODE_RETRANSMIT_TIMEOUT, TRUE);

       /* Call the main application state machine.                        */
       while(1)
          ApplicationMain();
    }
}

   /* The following is the Main application entry point.  This function */
   /* will configure the hardware and initialize the OS Abstraction     */
   /* layer, create the Main application thread and start the scheduler.*/
int main(void)
{
   /* Turn off the watchdog timer                                       */
   WDTCTL = WDTPW | WDTHOLD;
```

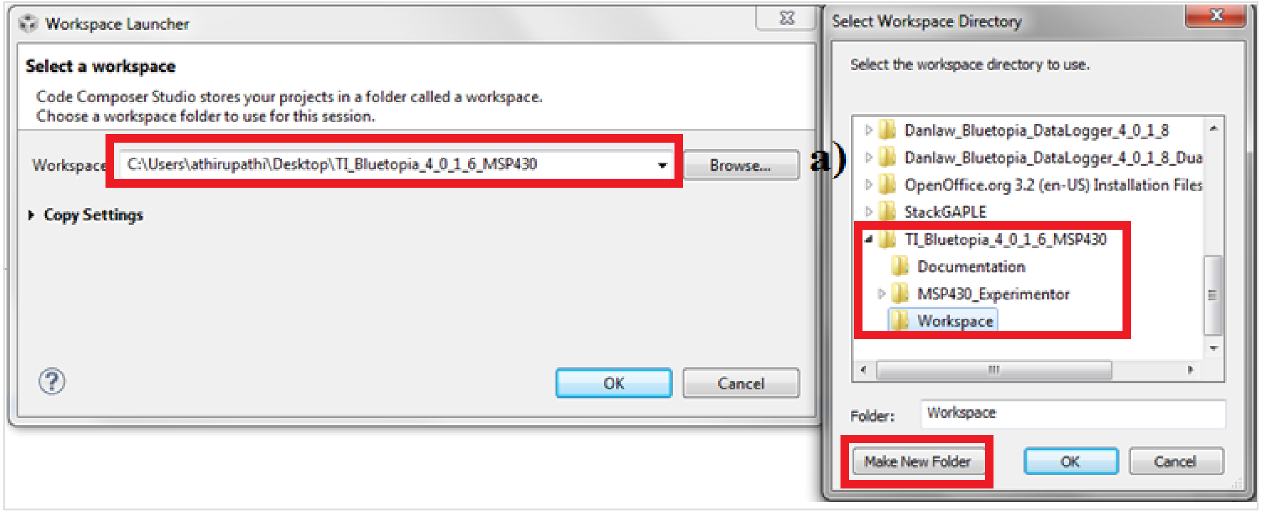
```
 021E0A      40B2 00
   TA1CCTL0 = CCIE
 021E10      40B2 00
   TA1CTL |= TACLR
 021E16      D2A2 03
   TA1CTL |= TASSE
 021E1A      D0B2 01
}
 021E20      0110
   WDTCTL = WDTPW
main:
 021E22      40B2 5A
   HAL_ConfigureHa
 021E28      13B2 18
   __enable_interr
 021E2C      D232
   MainThread();
 021E2E      13B2 0C
      HAL_LedToggl
```

g) The software is now ready to use. To begin using the profile disconnect and reconnect the MSP430 after flashing.

CCS

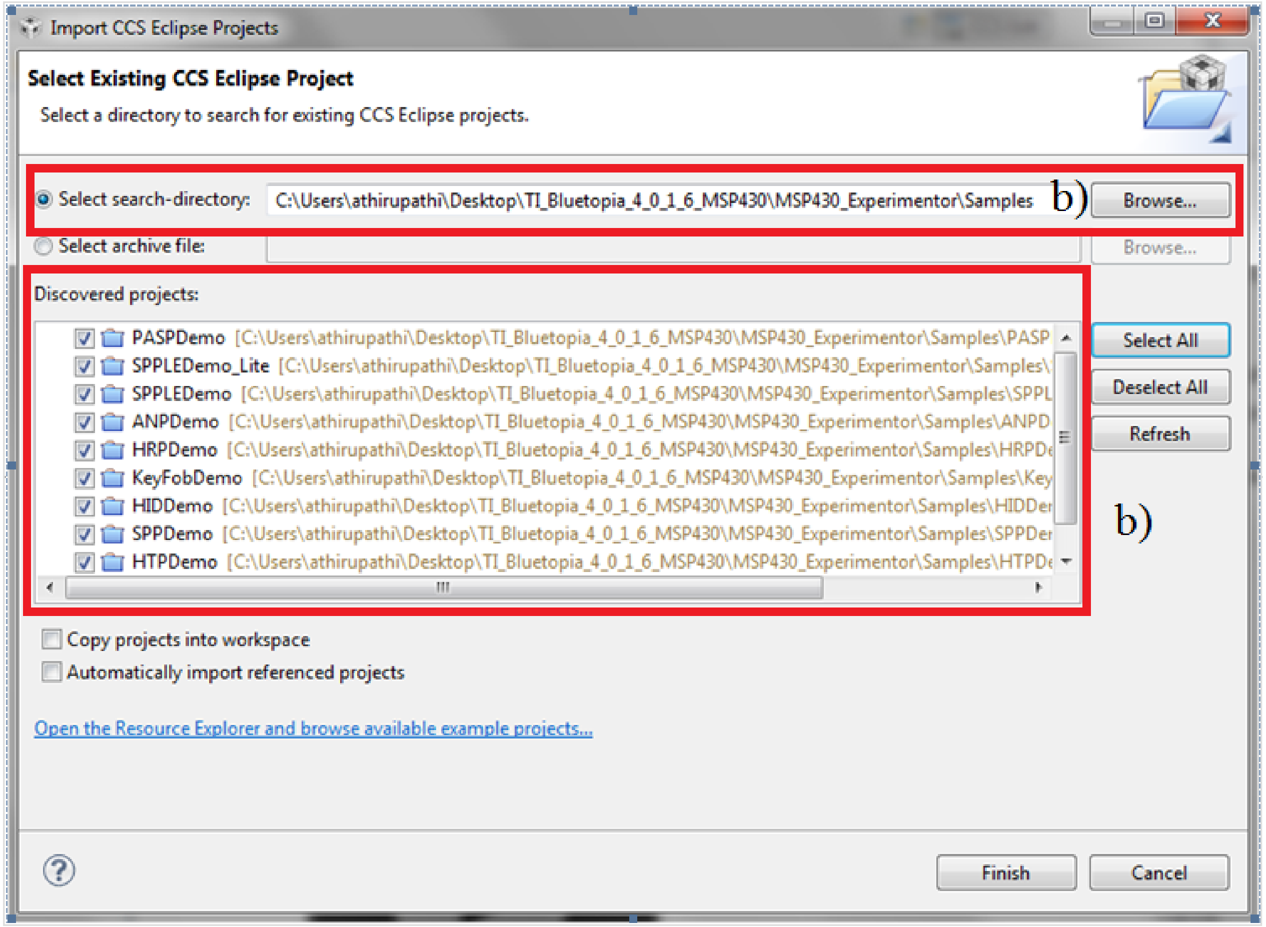
Note: The MSP430F5529 is not supported with CCS due to code size limitations.

a) Open Code Composer Studio 5. Select File->Switch Workspace. Navigate to the folder where the TI's Bluetooth stack Samples are located and make a new folder for the workspace.


Workspace Launcher
Select a workspace
Code Composer Studio stores your projects in a folder called a workspace.
Choose a workspace folder to use for this session.
Workspace C:\Users\athirupathi\Desktop\TI_Bluetopia_4_0_1_6_MSP430
Browse...
Copy Settings
OK
Cancel
a)
Select Workspace Directory
Select the workspace directory to use.
Danlaw_Bluetopia_DataLogger_4_0_1_8
Danlaw_Bluetopia_DataLogger_4_0_1_8_Dua
OpenOffice.org 3.2 (en-US) Installation Files
StackGAPLE
TI_Bluetopia_4_0_1_6_MSP430
Documentation
MSP430_Experimentor
Workspace
Folder: Workspace
Make New Folder
OK
Cancel

CCS will now exit and reload the software. Each time you load a project it is preferable to create a new workspace for it.

b) Select Project-> Import Existing CCS Eclipse project. In the select Auto search directory point to the TI's_Bluetooth_Stack_Version-Number_MSP430\MSP430_Experimentor\Samples folder. CCS will recognize all available profiles. Select all profiles and hit finish. (If you prefer you can selectively load the profiles you want).


Import CCS Eclipse Projects
Select Existing CCS Eclipse Project
Select a directory to search for existing CCS Eclipse projects.
Browse...
Browse...
b)
b)
Select All
Deselect All
Refresh
Finish
Cancel

Select search-directory: C:\Users\athirupathi\Desktop\TI_Bluetopia_4_0_1_6_MSP430\MSP430_Experimentor\Samples

Select archive file:

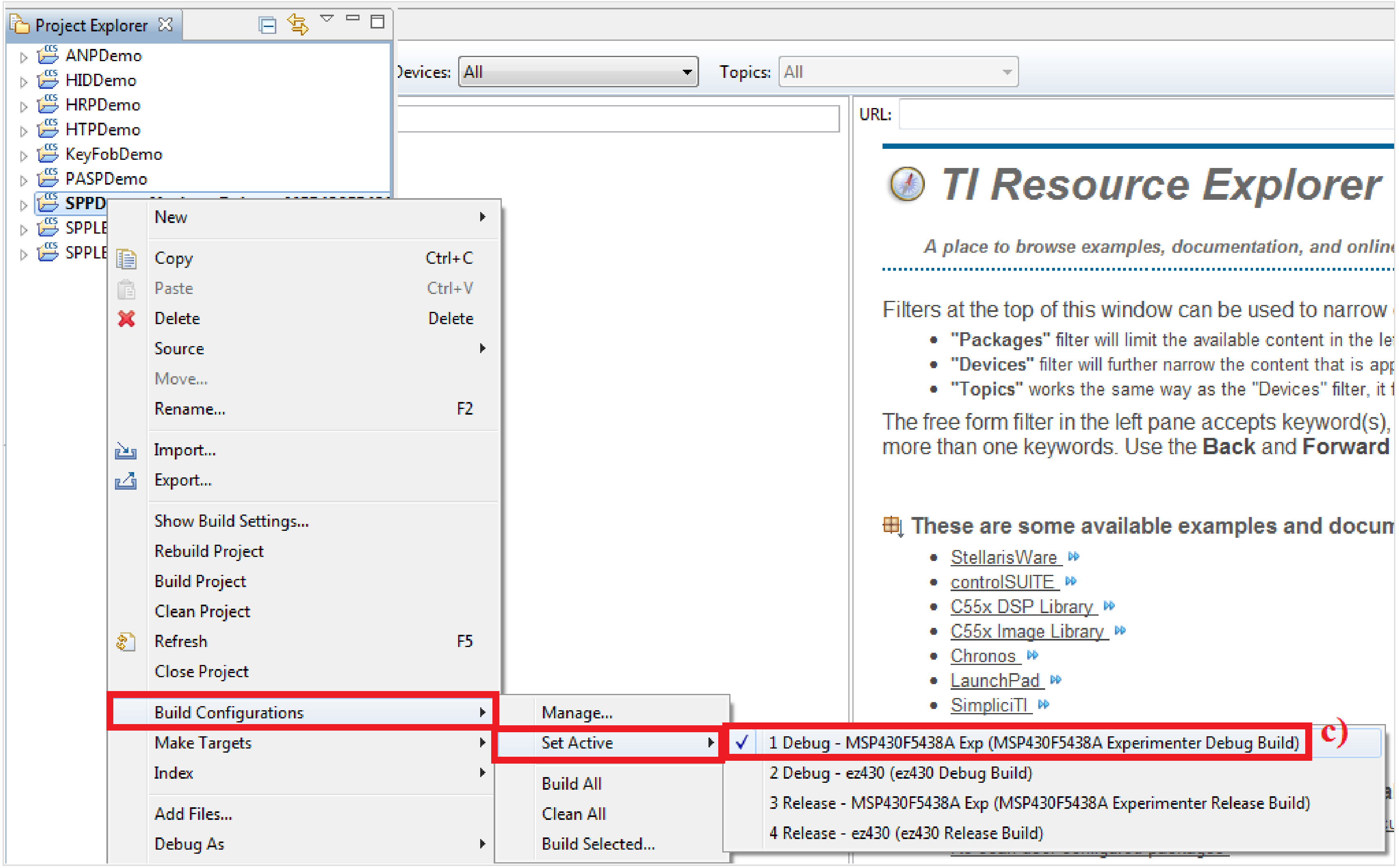
Discovered projects:

- [✓] PASPDemo [C:\Users\athirupathi\Desktop\TI_Bluetopia_4_0_1_6_MSP430\MSP430_Experimentor\Samples\PASP
- [✓] SPPLEDemo_Lite [C:\Users\athirupathi\Desktop\TI_Bluetopia_4_0_1_6_MSP430\MSP430_Experimentor\Samples\
- [✓] SPPLEDemo [C:\Users\athirupathi\Desktop\TI_Bluetopia_4_0_1_6_MSP430\MSP430_Experimentor\Samples\SPPL
- [✓] ANPDemo [C:\Users\athirupathi\Desktop\TI_Bluetopia_4_0_1_6_MSP430\MSP430_Experimentor\Samples\ANPD
- [✓] HRPDemo [C:\Users\athirupathi\Desktop\TI_Bluetopia_4_0_1_6_MSP430\MSP430_Experimentor\Samples\HRPD
- [✓] KeyFobDemo [C:\Users\athirupathi\Desktop\TI_Bluetopia_4_0_1_6_MSP430\MSP430_Experimentor\Samples\Key
- [✓] HIDDemo [C:\Users\athirupathi\Desktop\TI_Bluetopia_4_0_1_6_MSP430\MSP430_Experimentor\Samples\HIDDe
- [✓] SPPDemo [C:\Users\athirupathi\Desktop\TI_Bluetopia_4_0_1_6_MSP430\MSP430_Experimentor\Samples\SPPDe
- [✓] HTPDemo [C:\Users\athirupathi\Desktop\TI_Bluetopia_4_0_1_6_MSP430\MSP430_Experimentor\Samples\HTPD

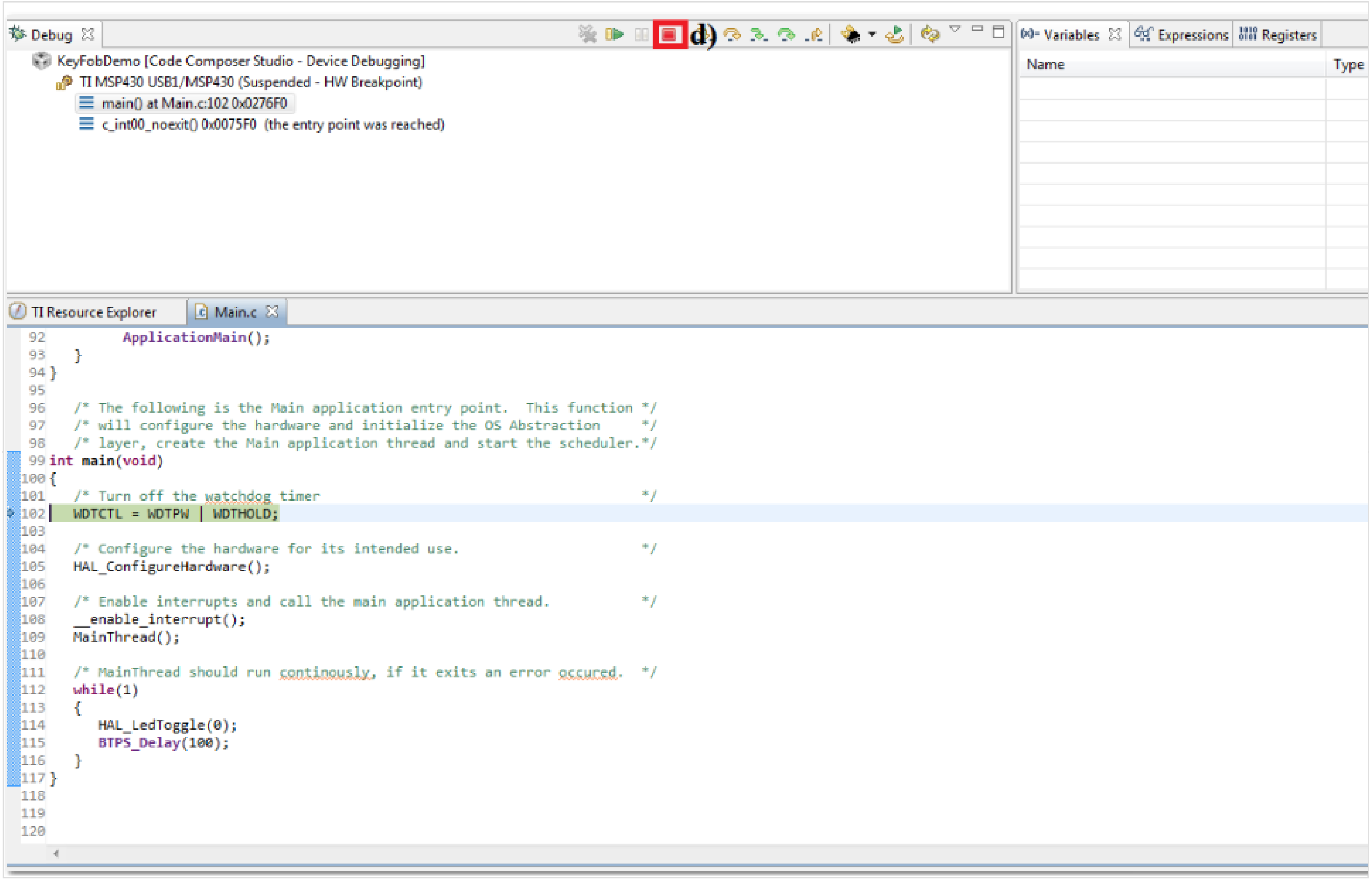
- [ ] Copy projects into workspace
- [ ] Automatically import referenced projects

Open the Resource Explorer and browse available example projects...

c) The projects should load on the left in Project Explorer. Click on the profile that you want to load. By default it should select the Debug, MSP 430 configuration for the profile. To change the configuration, right click on the profile-> Build Configurations -> Set Active. Select either the debug or release configuration for MSP430.


Project Explorer
ANPDemo
HIDDemo
HRPDemo
HTPDemo
KeyFobDemo
PASPDemo
SPPD
SPPLE
SPPLE
New
Copy Ctrl+C
Paste Ctrl+V
Delete Delete
Source
Move...
Rename... F2
Import...
Export...
Show Build Settings...
Rebuild Project
Build Project
Clean Project
Refresh F5
Close Project
Build Configurations
Make Targets
Index
Add Files...
Debug As
Manage...
Set Active
Build All
Clean All
Build Selected...
✓ 1 Debug - MSP430F5438A Exp (MSP430F5438A Experimenter Debug Build)
2 Debug - ez430 (ez430 Debug Build)
3 Release - MSP430F5438A Exp (MSP430F5438A Experimenter Release Build)
4 Release - ez430 (ez430 Release Build)
c)
Devices: All
Topics: All
URL:
TI Resource Explorer
A place to browse examples, documentation, and onlin
Filters at the top of this window can be used to narrow
"Packages" filter will limit the available content in the le
"Devices" filter will further narrow the content that is app
"Topics" works the same way as the "Devices" filter, it
The free form filter in the left pane accepts keyword(s),
more than one keywords. Use the Back and Forward
These are some available examples and docum
StellarisWare
controlSUITE
C55x DSP Library
C55x Image Library
Chronos
LaunchPad
SimpliciTI

d) To begin flashing the device, select from the toolbar -> Debug as -> Code composer session. Let the IDE debug and load the software on the device. This may take 5-10 minutes. Once it is done debugging, click on the stop button on the IDE to stop debugging.



```
KeyFobDemo [Code Composer Studio - Device Debugging]
    TI MSP430 USB1/MSP430 (Suspended - HW Breakpoint)
        main() at Main.c:102 0x0276F0
        c_int00_noexit() 0x0075F0  (the entry point was reached)
```

```
         ApplicationMain();
   }
}

   /* The following is the Main application entry point.  This function */
   /* will configure the hardware and initialize the OS Abstraction     */
   /* layer, create the Main application thread and start the scheduler.*/
int main(void)
{
   /* Turn off the watchdog timer                                       */
   WDTCTL = WDTPW | WDTHOLD;

   /* Configure the hardware for its intended use.                      */
   HAL_ConfigureHardware();

   /* Enable interrupts and call the main application thread.           */
   __enable_interrupt();
   MainThread();

   /* MainThread should run continously, if it exits an error occured.  */
   while(1)
   {
      HAL_LedToggle(0);
      BTPS_Delay(100);
   }
}
```

e) The software is now ready to use. To begin using the profile disconnect and reconnect the MSP430 after flashing.

Demos

Classic Bluetooth Demos

SPP Demo

- This application shows how to utilize the SPP module and also how handle the different callback events. The application can be used to interface with a remote SPP Client or Server.

- For a step by step demonstration on how to run this application visit the SPP Demo Wiki.

HID Demo

- The human interface device enables a host to connect and control a HID device.
- For a step by step demonstration on how to run this application visit the HID Demo Wiki.

A3DP Demo

- The Assisted Advanced Audio Distribution Profile shows how to send stereo audio over Bluetooth.
- For a step by step demonstration on how to run this application for the Sink Role visit the A3DP Demo Sink Wiki
- For a step by step demonstration on how to run this application for the Source Role visit the A3DP Demo Source Wiki

A3DP MultiRoom Demo

- The A3DP MultiRoom demo allows us to stream high quality audio to 2 different speakers at the same time.
- For a step by step demonstration on how to run the MultiRoom application visit the A3DP MultiRoom Demo Wiki.

HFP Demo

- The Hands Free Profile shows how to provide remote control and voice connections over Bluetooth to a mobile device. This profile supports the HandsFree Role.
- For a step by step demonstration on how to run this application visit the HFP Demo Wiki

Classic BT + BLE Demos

SPP + SPPLE Demo

- This application shows how to utilize Low Energy (LE) and the GATT profile. This sample emulates using SPP over LE using the GATT Profile. This sample acts as a LE Master and LE Slave.
- This application also shows how to utilize the SPP module and how to handle the different SPP callback events. This will act as either a SPP Server or SPP Client and exposes the same command set for SPP as the SPPDemo.
- For a step by step demonstration on how to run this application visit the SPP + SPPLE Demo Wiki.

KeyFob Demo

- The KeyFobDemo is used to demonstrate button press or accelerometer data over LE.
- For a step by step demonstration on how to run this application visit the KeyFob Demo Wiki.

SPPLE Demo Lite

- The SPPLE Lite demo is a server-only version of SPP LE where, in addition to supporting the SPP LE protocol, the classic Bluetooth profile SPP is also initialized as a server, and the device is ready to accept connections over classic Bluetooth (a.k.a. BR/EDR) or Bluetooth Low Energy (BLE).
- For a step by step demonstration on how to run this application visit the SPPLE Demo Lite Wiki.

Bluetooth Low Energy Demos

ANP Demo

- This application shows how to utilize Low Energy (LE) and the GATT profile. This sample implements the Alert Notification Profile (ANP) using the GATT profile. This sample acts as a LE Master and LE Slave.
- For a step by step demonstration on how to run this application visit the ANP Demo Wiki.

HRP Demo

- This application shows how to utilize Low Energy (LE) and the GATT profile. This sample implements the Heart Rate Monitoring Profile (HRP) using the GATT profile. This sample acts as a LE Master and LE Slave.
- For a step by step demonstration on how to run this application visit the HRP Demo Wiki.

HTP Demo

- This application shows how to utilize Low Energy (LE) and the GATT profile. This sample implements the Health Thermometer Profile (HTP) using the GATT profile. This sample acts as a LE Master and LE Slave.
- For a step by step demonstration on how to run this application visit the HTP Demo Wiki.

PASP Demo

- This application shows how to utilize Low Energy (LE) and the GATT profile. This sample implements the Phone Alert Status Profile (PASP) using the GATT profile. This sample acts as a LE Master and LE Slave.
- For a step by step demonstration on how to run this application visit the PASP Demo Wiki.

HOGP Demo

- This application shows how to utilize Low Energy (LE) and the GATT profile. This sample implements the Human Interface Device (HID) using the GATT profile.
- For a step by step demonstration on how to run this application visit the HOGP Demo Wiki.
- {HOGP Demo Legacy page}.

PXP Demo

- This application shows how to utilize Low Energy (LE) and the GATT profile. This sample implements the ProXimity Profile(PXP) using the GATT profile. This sample acts as a LE Master and Slave.
- For a step by step demonstration on how to run this application visit the PXP Demo Wiki.

FMP Demo

- This application shows how to utilize Low Energy (LE) and the GATT profile. This sample implements the Find Me Profile(FMP) using the GATT profile. This sample acts as a LE Master and Slave.
- For a step by step demonstration on how to run this application visit the FMP Demo Wiki.

See Also

- CC256x Main Wiki Page
- CC256x Tiva TI Bluetooth Stack
- TI Bluetooth Stack Download (http://www.ti.com/tool/tibluetoothstack-sdk)

{{ 1. switchcategory:MultiCore= ▪ For technical support on MultiCore devices, please post your questions in the C6000 MultiCore Forum ▪ For questions related to the BIOS MultiCore SDK (MCSDK), please use the BIOS Forum Please post only comments related to the article **CC256x MSP430 TI Bluetooth Stack** here.	Keystone= ▪ For technical support on MultiCore devices, please post your questions in the C6000 MultiCore Forum ▪ For questions related to the BIOS MultiCore SDK (MCSDK), please use the BIOS Forum Please post only comments related to the article **CC256x MSP430 TI Bluetooth Stack** here.	C2000=*For technical support on the C2000 please post your questions on The C2000 Forum. Please post only comments about the article **CC256x MSP430 TI Bluetooth Stack** here.*	DaVinci=*For technical support on DaVincoplease post your questions on The DaVinci Forum. Please post only comments about the article **CC256x MSP430 TI Bluetooth Stack** here.*	MSP430=*For technical support on MSP430 please post your questions on The MSP430 Forum. Please post only comments about the article **CC256x MSP430 TI Bluetooth Stack** here.*	OMAP35x=*For technical support on OMAP please post your questions on The OMAP Forum. Please post only comments about the article **CC256x MSP430 TI Bluetooth Stack** here.*	OMAPL1=*For technical support on OMAP please post your questions on The OMAP Forum. Please post only comments about the article **CC256x MSP430 TI Bluetooth Stack** here.*	MAVRK=*For technical support on MAVRK please post your questions on The MAVRK Toolbox Forum. Please post only comments about the article **CC256x MSP430 TI Bluetooth Stack** here.*	*For technical support please post your questions at http://e2e.ti.com. Please post only comments about the article **CC256x MSP430 TI Bluetooth Stack** here.* }}

Links

Amplifiers & Linear
Audio
Broadband RF/IF & Digital Radio
Clocks & Timers
Data Converters

DLP & MEMS
High-Reliability
Interface
Logic
Power Management

Processors

- ARM Processors
- Digital Signal Processors (DSP)
- Microcontrollers (MCU)
- OMAP Applications Processors

Switches & Multiplexers
Temperature Sensors & Control ICs
Wireless Connectivity

Retrieved from "https://processors.wiki.ti.com/index.php?title=CC256x_MSP430_TI_Bluetooth_Stack&oldid=222132"

This page was last edited on 21 October 2016, at 09:26.