



**User Manual/Developer's
Guide/Reference Guide**

**WISE-1021
Non-OS
Software Development Kit**

ADVANTECH

Enabling an Intelligent Planet

Change Log:

Date	Version	Description / Major change
2016/04/07	V0.1	Will Chen, create draft document.

Table of Content

1. INTRODUCTION	3
1.1 Glossary.....	3
2. WISE-1021 NON-OS SOFTWARE DEVELOPMENT KIT (SDK)	4
2.1. Source Tree	4
2.2. Software Packages.....	4
3. SETUP DEVELOPMENT ENVIRONMENT	5
3.1. Preparing for Hardware	5
3.2. Setting for Connectivity	8
3.2.1. Command Line Interface (CLI)	8
3.2.2. Information and Parameters Setting.....	9
3.3. Code Composer Studio (CCS)	11
3.3.1. Running CCS installer.....	11
3.3.2. Import Project.....	13
3.3.3. Compile, Programming and Debug	16
4. EXAMPLE	18
4.1. Import Examples	18
4.2. Send Data.....	18
4.3. Exchange Data with Sensor Format	23
4.3.1. How to Handle Data between Mote and Gateway.....	23
4.3.2. Running Example	24
5. APPENDIX.....	26
5.1. References	26
5.1.1. TI.....	26
5.1.2. Linear Dust	26

1. Introduction

This document is description about WISE-1021 Non-OS software development kit (SDK) on WISE-1021. It includes source tree of SDK, development environment, flash programming...etc. There are some examples provided by WISE-1021 Non-OS SDK the user can refer to it for how to do about connection to WISE-PaaS, communication between mote and gateway...etc., and also can refer to it for developing their application.

1.1 Glossary

- Gateway
The gateway like WISE-3310 has connectivity with SmartMesh IP manager responsible for communication with WISE-1021.
- Mote
The mote is WISE-1021 that has connectivity with SmartMesh IP mote responsible for communication with gateway.

2. WISE-1021 Non-OS Software Development Kit (SDK)

2.1. Source Tree

The user can find source tree as below after file “WISE-1021_NonOS_SDK-VX.X.X.zip” is extracted.

Directory	Description
doc	Documents about WISE-1021 Non-OS SDK.
inc	Header files including examples, sensor driver and etc.
lib	Library.
project	Project files for IDE.
src	Source files including examples, sensor driver and etc.
packages	Software packages.

2.2. Software Packages

The WISE-1021 Non-OS SDK is based on MSP430F5419A to development. The user needs to install some TI packages before development. The user can find these packages in folder “packages” of source tree of WISE-1021 Non-OS SDK. The list of packages as following:

File Name	Description
ccs_setup_win32.exe	TI Code Composer Studio (CCS) Integrated Development Environment (IDE).
MSP430Drivers-1_00_00_01-windows-installer.exe	Stand-alone Driver Installer for USB Low-Level Drivers.

3. Setup Development Environment

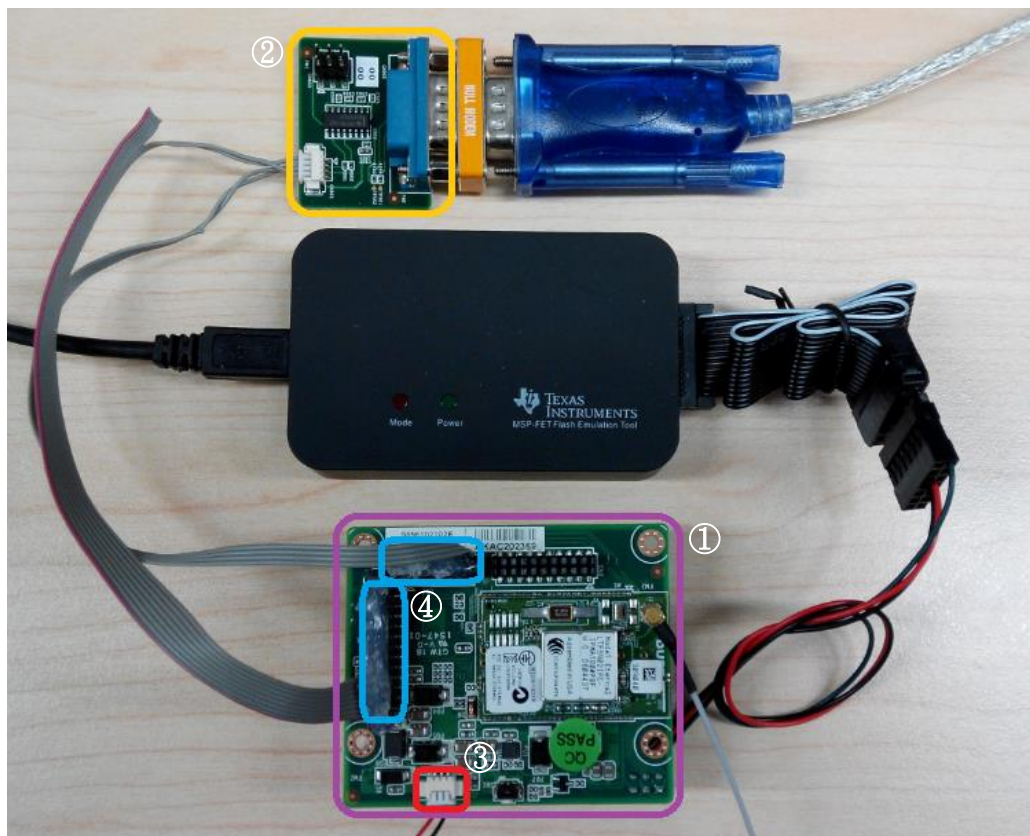
3.1. Preparing for Hardware

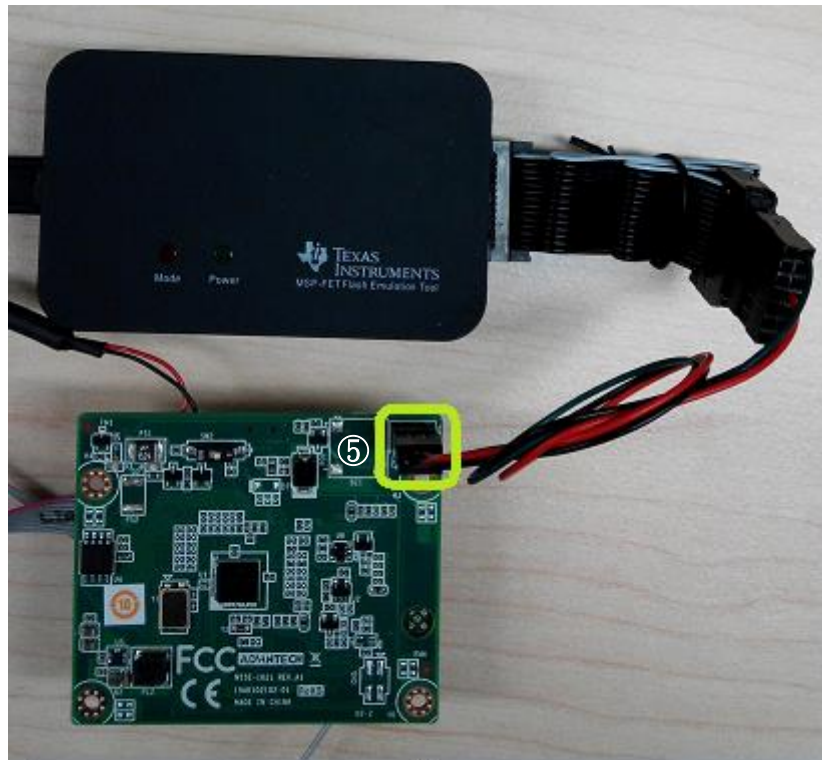
The user needs to prepare for hardware as following:

- WISE-1021 and WISE-3310.
- MSP-FET Flash Emulation Tool.
- One PC running the Microsoft® Windows® 7 or XP operating systems and the other running LINUX operating systems like Ubuntu 14.04.1.

Please refer to the following steps for setup a WISE series of boards before using WISE-1021 Non-OS SDK.

1. Please prepare boards and gateway as below.



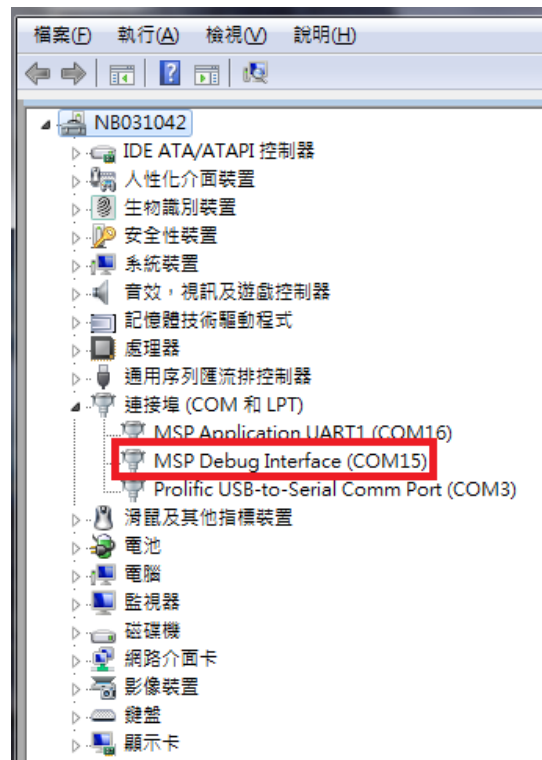


- ① WISE-1021
- ② ROM-ED20
- ③ DC-IN
- ④ CLI cable
- ⑤ JTAG cable
- ⑥ WISE-3310

2. Connect the MSP-FET Flash Emulation Tool to PC.

The device will be visible in the Device Manager as below after MSPDS-USB Low-Level driver installed. The driver installer

“MSP430Drivers-1_00_00_01-windows-installer.exe” should be found at folder “packages” of source tree of WISE-1021 Non-OS SDK.



3.2. Setting for Connectivity

WISE-1021 is designed in connectivity for wireless IoT communication with Linear Dust SmartMesh IP wireless sensor networks. The Linear Dust networks support a self-forming and self-healing mesh network solution compliant to 6LoWPAN Internet Protocol (IP) and IEEE802.15.4e standard. Before using WISE-1021 Non-OS SDK, the user needs to check or set some parameters through the command line interface (CLI).

3.2.1. Command Line Interface (CLI)

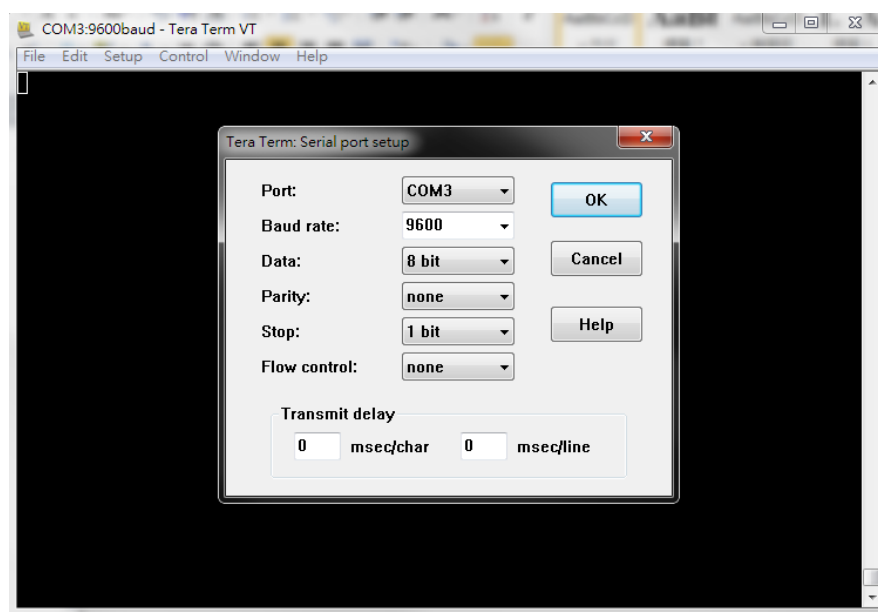
The CLI is interface to interact with connectivity by connecting a serial terminal program. It's used to getting information, checking/setting parameters or troubleshooting...etc. The user can refer to [SmartMesh IP Mote CLI Guide](#) for getting more detailed information about CLI. Please refer to the following steps for setup CLI.

1. Check device connected to PC, choose any terminal you like (e.g., the below snapshot is setting of Tera Term) and setup default serial port settings are as follows:

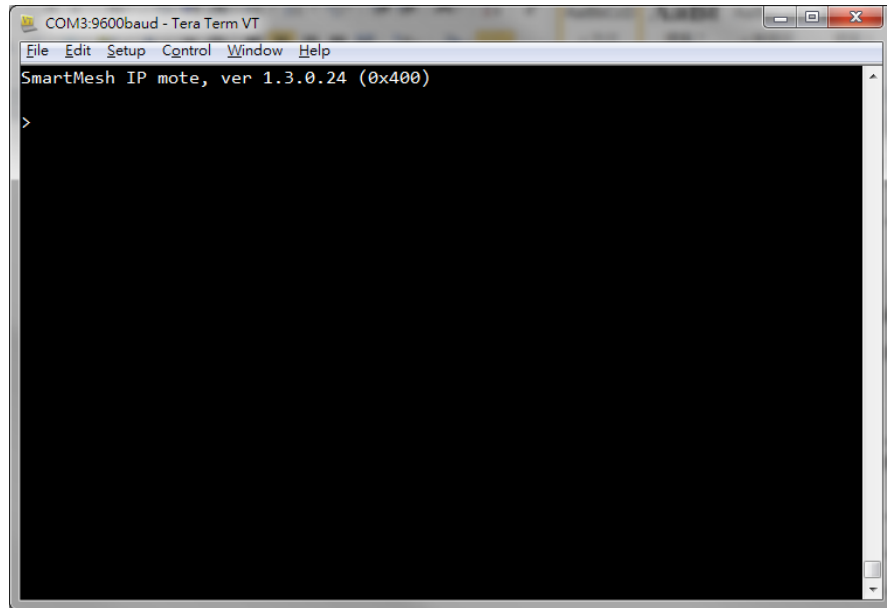
- Bits per second: 9600
- Data bits: 8
- Parity: None
- Stop bits: 1
- Flow control: None

Note1: please refer to section "[Preparing for Hardware](#)" for checking cable connected with CLI.

Note2: please check which serial port you connected to device and set it for terminal.



2. User can see snapshot as below after power on.



3.2.2. Information and Parameters Setting

The user needs to double check connectivity setting the same as below table with SmartMesh IP networks before using WISE-1021 Non-OS SDK:

Parameter	Expected Setting
Mode	Slave mode
Network ID	2001 (The same as manager on WISE-3310)
Join Key	4A4F494E414456414E54454348494F54 (The same as manager on WISE-3310)
Auto Join	On

The following will show the user how we could check/set these parameters through CLI.

- Get information

```
> minfo

Net stack  v1.2.1.17
state:     Search
mac:       00:17:0d:00:00:30:88:ff
moteid:    0
netid:     2001
```

```
bISwVer: 15
ldrSwVer: 1.0.3.12
UTC time: 0:0
reset st: 400
```

- Switch Mode

```
> set mode slave

> get mode // get information about which mode is set with WISE-1021.
slave

> reset // this function will take effect after reset.
```

- Change Network ID

```
> mset netid 100
> mget netid
netid=100

> reset // this function will take effect after reset.
```

Note: the network ID is short unsigned integer (2 bytes)

- Write Join Key

```
> mset jkey 4A4F494E414456414E54454348494F54

> reset // this function will take effect after reset.
```

Note: the join key is a 16 bytes hexadecimal password.

- Switch Auto Join

```
> mset autojoin 1

> get autojoin
autojoin = 1

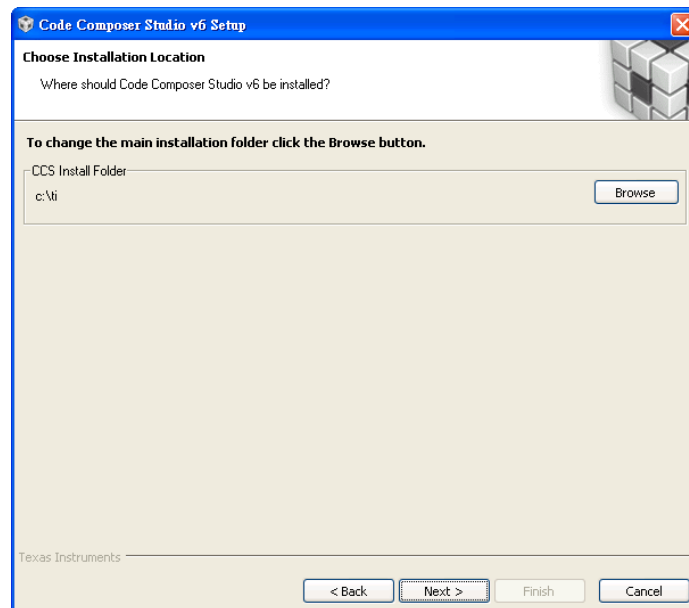
> reset // this function will take effect after reset.
```

3.3. Code Composer Studio (CCS)

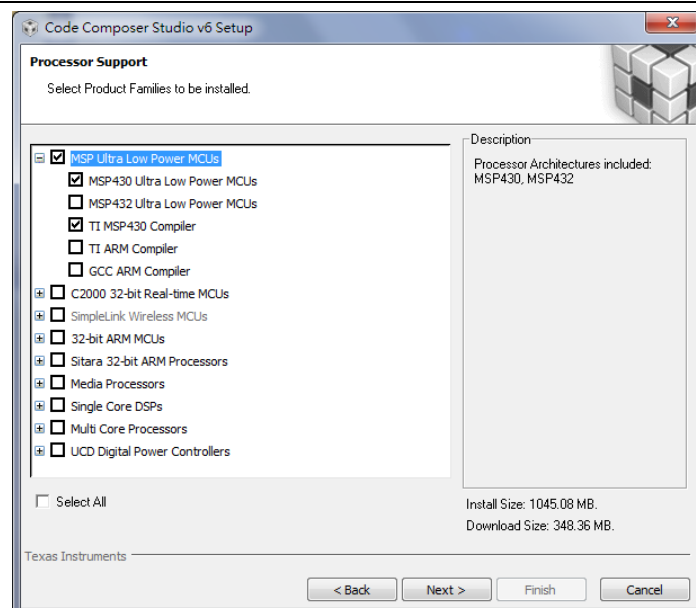
The user can refer to the following steps for setup CCS with WISE-1021 Non-OS SDK and get more information from section "[TI](#)".

3.3.1. Running CCS installer

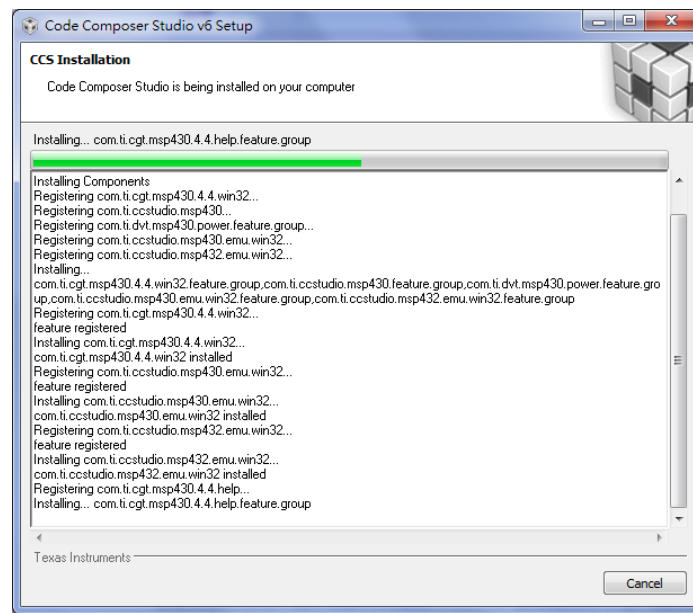
3. Find installer “ccs_setup_win32.exe” in source tree and run it to install CCS and set installation directory to default as below.



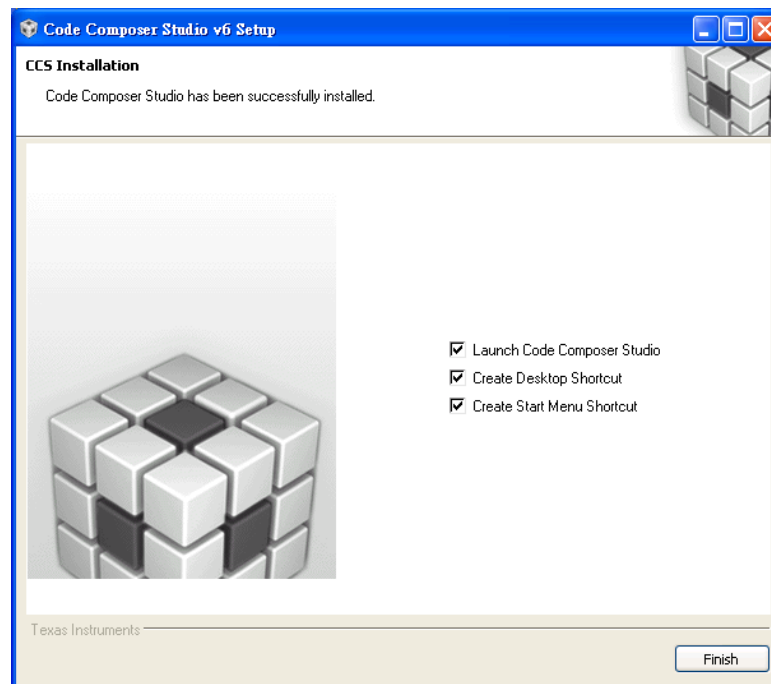
4. Select options about MSP430 as below for “Processor Support”.



5. Installation process.

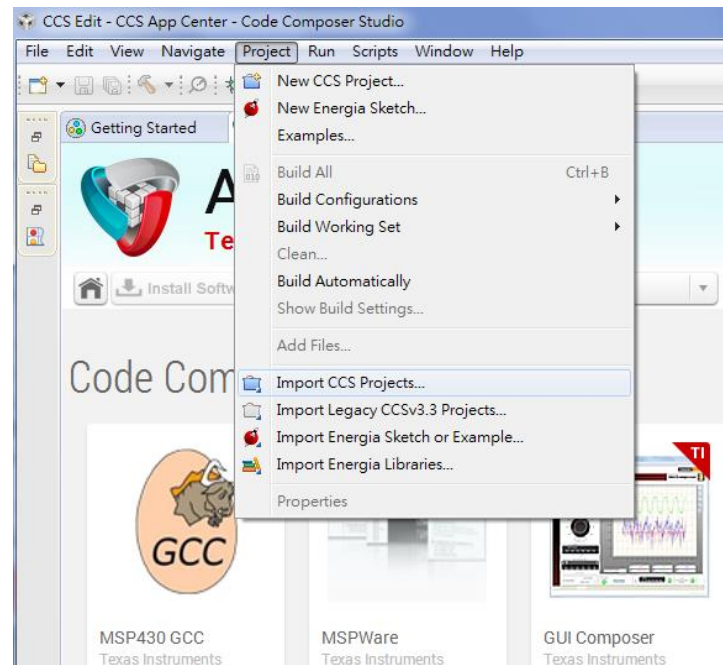


6. Finish installation.

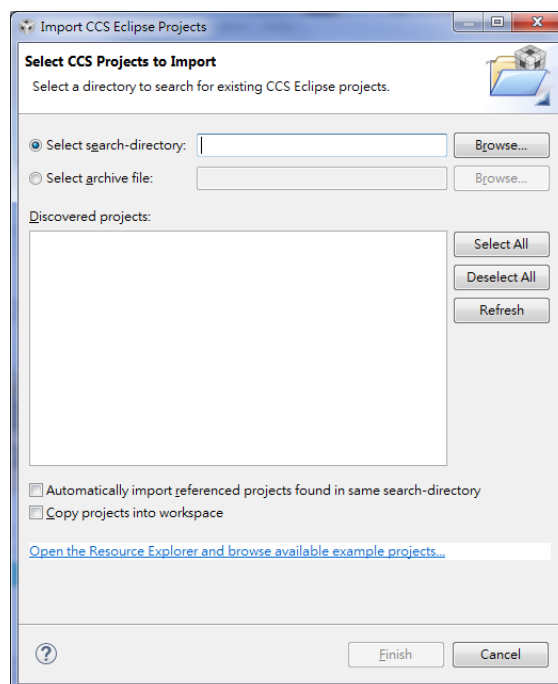


3.3.2. Import Project

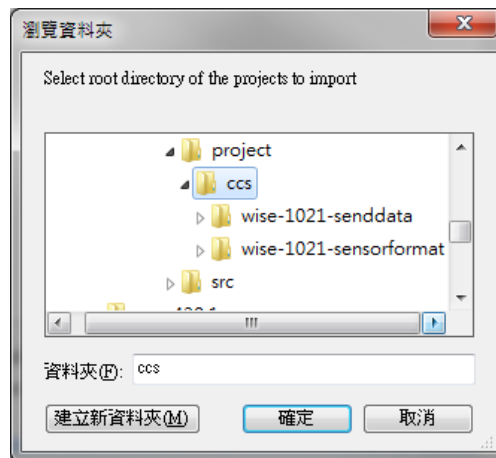
1. Launch CCS.
2. Choose Project -> Import CCS Projects from the menu.



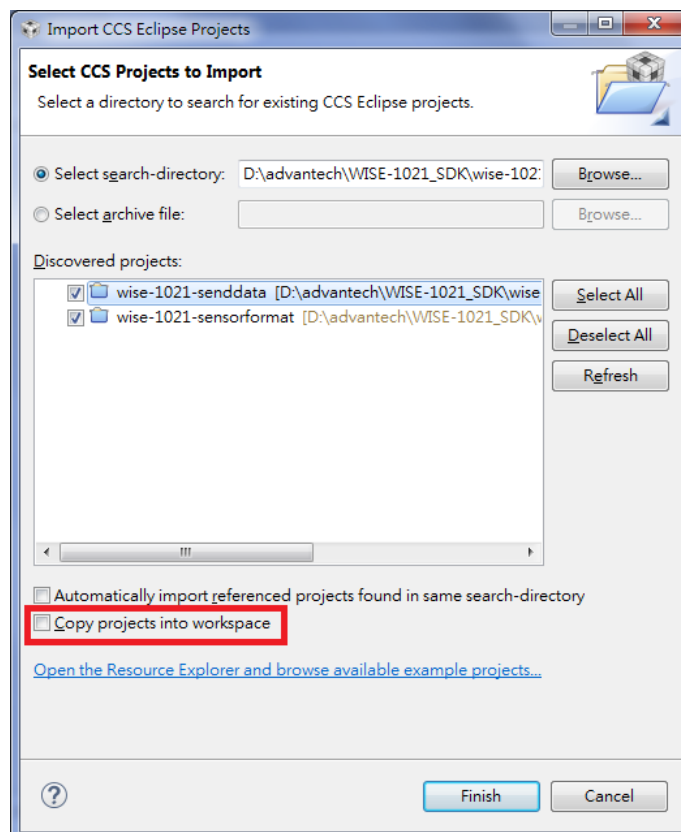
3. Select the Browse button in the Import CCS Eclipse Projects dialog.



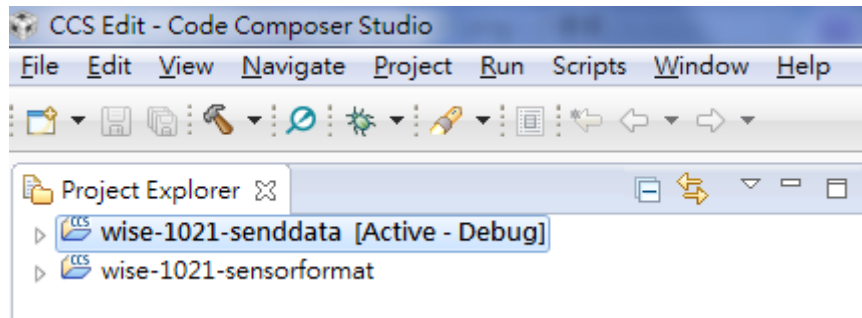
4. Select the CCS project directory of WISE-1021 Non-OS SDK where you extracted before to root directory. For e.g. D:\WISE-1021_NonOS_SDK_vX.X.X\project\ccs



5. Import projects as below and press “Finish” button.
Don’t check the “Copy projects into workspace” box.

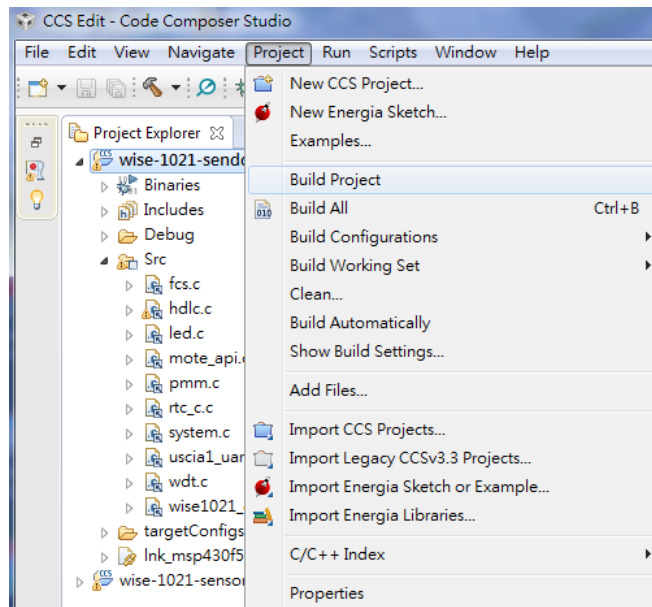


6. You can see imported projects as below in “Project Explorer”.

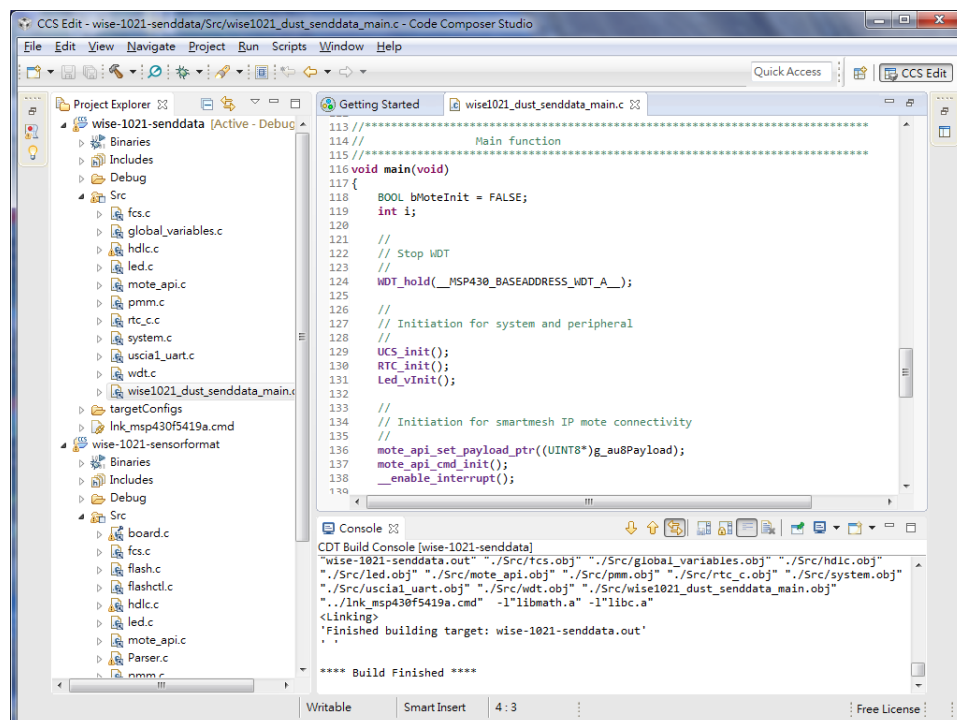


3.3.3. Compile, Programming and Debug

1. Select the “wise-1021-senddata” or “wise-1021-sensorformat” project in Project Explorer, and select Project -> Build Project to build it.

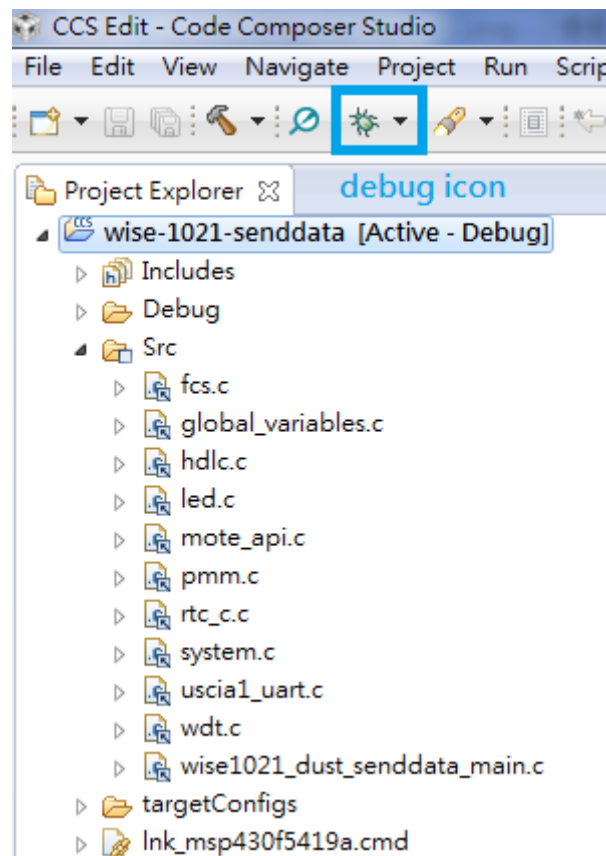


2. Finish compilation as below.



3. Check device connected from WISE-1021, MSP-FET Flash Emulation Tool to PC.
Please refer to section "[Preparing for Hardware](#)" for more information about hardware if user needed.

4. Launch application and programming.
Select the "wise-1021-senddata" or "wise-1021-sensorformat" project in Project Explorer, then click the debug icon as below to download code and programming to the device and begin debugging. Press F8 to begin execution.



4. Example

There are two examples in WISE-1021 Non-OS SDK and the user can import it from folder “project” of source tree of WISE-1021 Non-OS SDK. One is “Send Data” the user can refer to it for how to use connectivity to send data from mote to gateway. The other is “Exchange Data with Sensor Format” the user can refer to it for how to add sensor in mote and send data accessed from sensor to WISE-PaaS through mote and gateway.

4.1. Import Examples

Please refer to section “[Import Project](#)” for how to import all examples.

4.2. Send Data

This example is demonstrated how we could send data from mote to gateway. The user needs to run two programs synchronously that one is on WISE-1021 and the other is on gateway. In example it communicates with gateway using API in “mote_api.c” that can found located at folder “src\wsn\sm_ipmt” of source tree of WISE-1021 Non-OS SDK. The “mote_api.c” has been implemented based on [SmartMesh IP Mote API Guide](#) and the user can refer to it for getting detailed if needed.

Please refer to the following to run example between mote and gateway:

● Running example on Gateway

1. Get IoT Gateway SDK and Setup on PC.

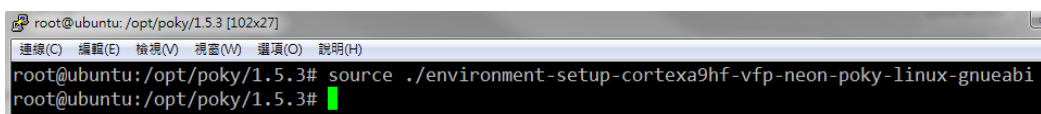
The IoT Gateway SDK is not provided as default and the user needs contact with Advantech to get it.

2. Compile example.

- ① Set environment variable for compilation.

PC # cd /opt/poky/1.5.3/

PC # source ./environment-setup-cortexa9hf-vfp-neon-poky-linux-gnueabi



```
root@ubuntu: /opt/poky/1.5.3 [102x27]
root@ubuntu: /opt/poky/1.5.3# source ./environment-setup-cortexa9hf-vfp-neon-poky-linux-gnueabi
root@ubuntu: /opt/poky/1.5.3#
```

② The example “wise1021_senddata_gw.tar.bz2” is located in folder “src\sample” of source tree of WISE-1021 Non-OS SDK. Please extract wise1021_senddata_gw.tar.bz2 to PC.

PC # tar jxvf wise1021_senddata_gw.tar.bz2

```
root@ubuntu: /tmp [102x27]
root@ubuntu:/tmp# tar jxvf wise1021_senddata_gw.tar.bz2
wise1021_senddata_gw/
wise1021_senddata_gw/DN_LICENSE.txt
wise1021_senddata_gw/ipmg/
wise1021_senddata_gw/ipmg/rs232.c
wise1021_senddata_gw/ipmg/wise1021_receive_sample.c
wise1021_senddata_gw/ipmg/IpMgWrapper.c
```

③ Compile and output binary.

PC # cd wise1021_senddata_gw/

PC # make clean

PC # make

```
root@ubuntu: /home/will/wsn_send_test_for_1021/wise1021_senddata_gw [102x27]
root@ubuntu:/home/will/wsn_send_test_for_1021/wise1021_senddata_gw# make
set -e; for d in sm_clib ipmg; do make -C ${d}; done
make[1]: Entering directory `/home/will/wsn_send_test_for_1021/wise1021_senddata_gw/sm_clib'
arm-poky-linux-gnueabi-gcc -march=armv7-a -mthumb-interwork -mfloat-abi=hard -mfpu=neon -mtune=cortex-a9 --sysroot=/opt/poky/1.5.3/sysroots/cortexa9hf-vfp-neon-poky-linux-gnueabi -O2 -pipe -g -feliminate-unused-debug-types -c dn_hdlc.c -o dn_hdlc.o
arm-poky-linux-gnueabi-gcc -march=armv7-a -mthumb-interwork -mfloat-abi=hard -mfpu=neon -mtune=cortex-a9 --sysroot=/opt/poky/1.5.3/sysroots/cortexa9hf-vfp-neon-poky-linux-gnueabi -O2 -pipe -g -feliminate-unused-debug-types -c dn_ipmg.c -o dn_ipmg.o
```

PC # ls ipmg/

```
root@ubuntu:/tmp/wise1021_senddata_gw# ls ipmg/
IpMgWrapper.c IpMgWrapper.o rs232.c rs232.o wise1021_receive_sample.c
IpMgWrapper.h Makefile rs232.h wise1021_receive_sample wise1021_receive_sample.o
root@ubuntu:/tmp/wise1021_senddata_gw#
```

3. Upload binary to WISE-3310.

The following commands are for the user reference. These commands are demonstrated how the PC's file system can be mounted through NFS on WISE-3310 and then the user can copy binary to WISE-3310 using command “cp”.

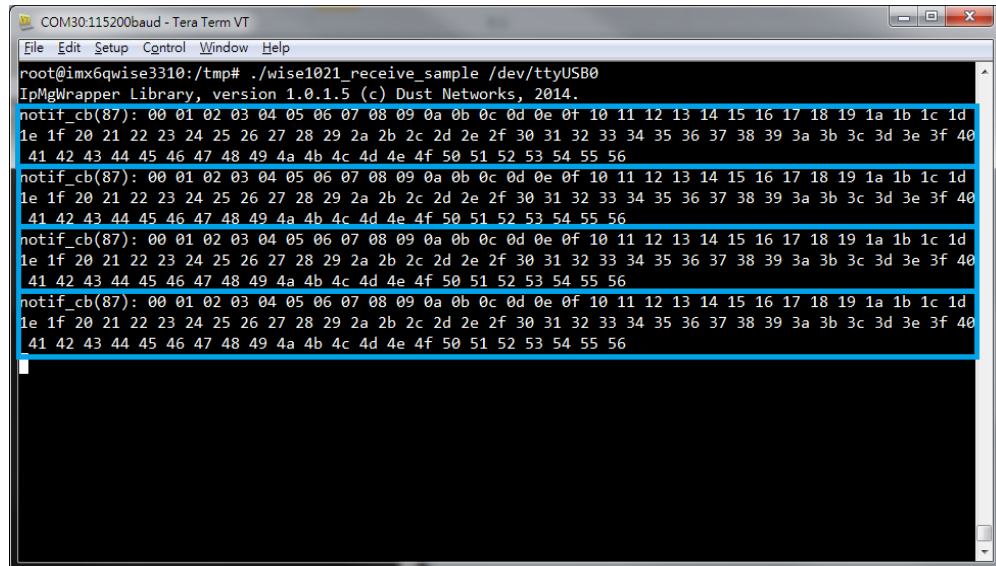
WISE-3310 # mount -t nfs -o nolock 172.22.12.223:/tmp /mnt/

WISE-3310 # cp /mnt/wise1021_senddata_gw/ipmg/wise1021_receive_sample /tmp

4. Begin execution.

The user can see snapshot as below while getting data from mote.

WISE-3310 # /tmp/ipmg/wise1021_receive_sample /dev/ttyUSB0



```
COM30:115200baud - Tera Term VT
File Edit Setup Control Window Help
root@imx6qwise3310:/tmp# ./wise1021_receive_sample /dev/ttyUSB0
IpMgWrapper Library, version 1.0.1.5 (c) Dust Networks, 2014.
notif_cb(87): 00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d
1e 1f 20 21 22 23 24 25 26 27 28 29 2a 2b 2c 2d 2e 2f 30 31 32 33 34 35 36 37 38 39 3a 3b 3c 3d 3e 3f 40
41 42 43 44 45 46 47 48 49 4a 4b 4c 4d 4e 4f 50 51 52 53 54 55 56
notif_cb(87): 00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d
1e 1f 20 21 22 23 24 25 26 27 28 29 2a 2b 2c 2d 2e 2f 30 31 32 33 34 35 36 37 38 39 3a 3b 3c 3d 3e 3f 40
41 42 43 44 45 46 47 48 49 4a 4b 4c 4d 4e 4f 50 51 52 53 54 55 56
notif_cb(87): 00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d
1e 1f 20 21 22 23 24 25 26 27 28 29 2a 2b 2c 2d 2e 2f 30 31 32 33 34 35 36 37 38 39 3a 3b 3c 3d 3e 3f 40
41 42 43 44 45 46 47 48 49 4a 4b 4c 4d 4e 4f 50 51 52 53 54 55 56
notif_cb(87): 00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d
1e 1f 20 21 22 23 24 25 26 27 28 29 2a 2b 2c 2d 2e 2f 30 31 32 33 34 35 36 37 38 39 3a 3b 3c 3d 3e 3f 40
41 42 43 44 45 46 47 48 49 4a 4b 4c 4d 4e 4f 50 51 52 53 54 55 56
```

- **Running example on WISE-1021**

1. Compile example and programming.

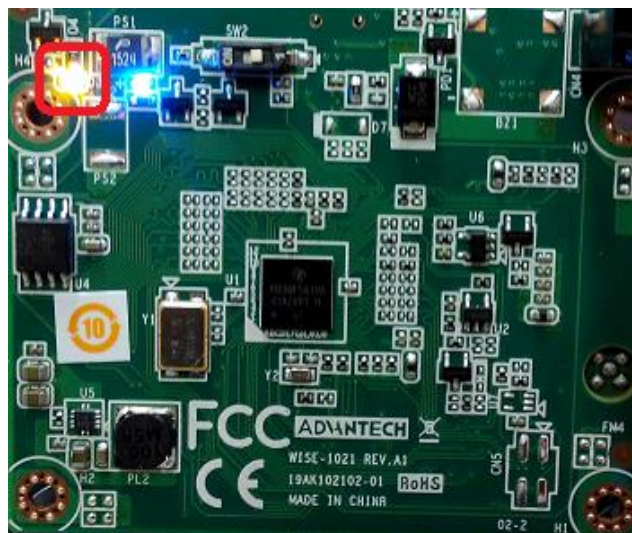
- ① Select the “wise-1021-senddata” project in Project Explorer, and select Project -> Build Project to build it.
- ② Click the debug icon as below to programming to the device.



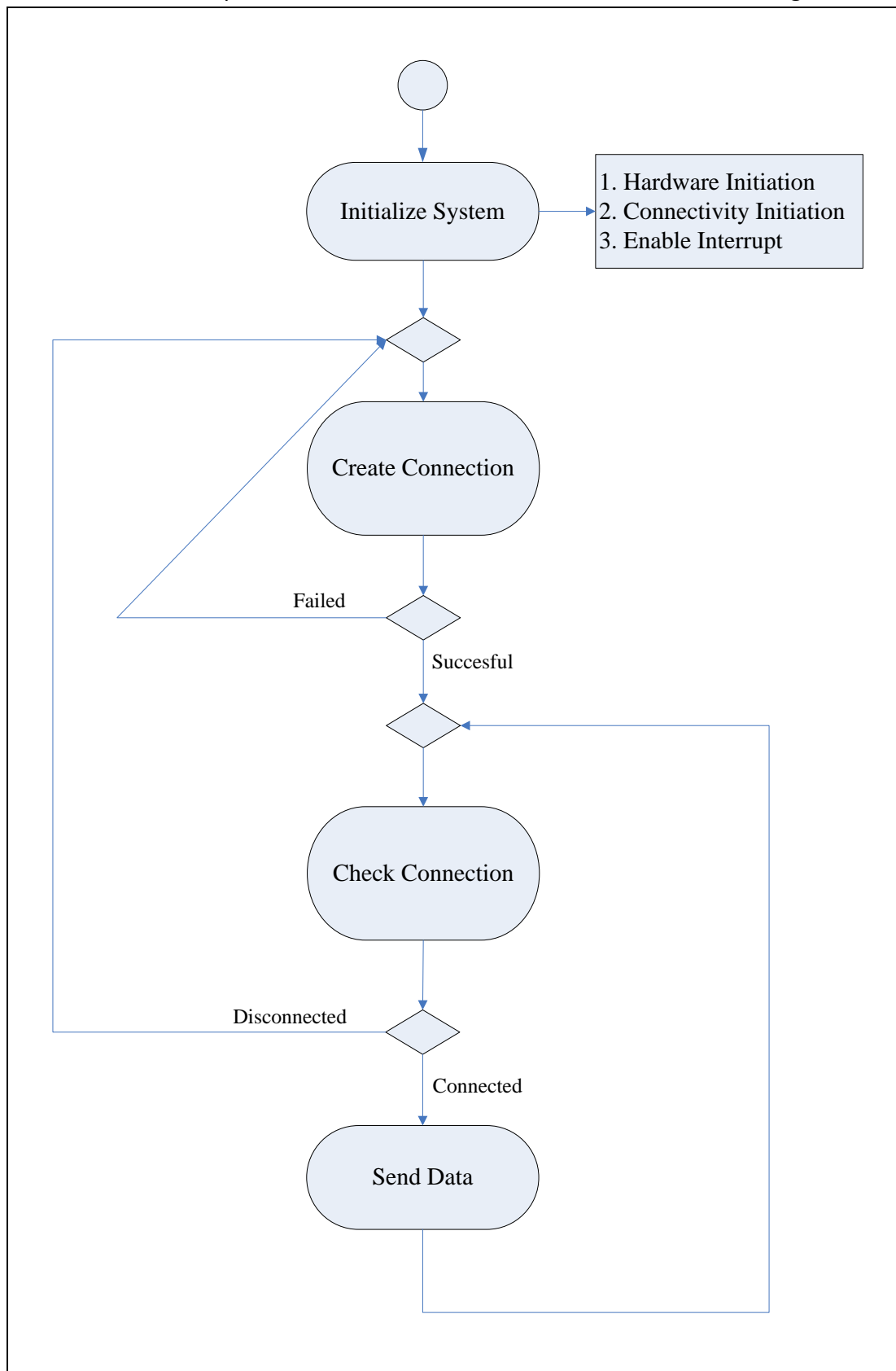
2. Check status with LED.

Waiting until the yellow LED is turned on. It's mean that WISE-1021 begin sending data to gateway every seconds defined by macro “SCHEDULE_TIME” in sample code.

Note: The yellow LED will be turned off when disconnected between WISE-1021 and gateway.



The flowchart is example “Send Data” how it runs on WISE-1021 as below figure.



4.3. Exchange Data with Sensor Format

This example is demonstrated how it could exchange data with sensor format between mote and gateway. The sensor format defined by Advantech combines IPSO data format with CoAP like protocol. By implementation the user needs to study “WISE sensor API” to know how to add sensor by self. In example it adds fake temperature sensor with sensor format and exchanges data between mote and gateway. Finally the fake temperature will be transfer to WISE-PaaS and it is presented by UI on browser, but WISE-PaaS has no more detailed in this document. Please get contact about WISE-PaaS detailed with Advantech if needed.

4.3.1. How to Handle Data between Mote and Gateway

Based on sensor format, the brief in bidirectional explain how it has been implemented in example as below:

From mote to gateway :

- ① The data accessed from sensor are encoded as packet with sensor format by mote.
- ② The packet is sent from mote to gateway.
- ③ The packet is received and decoded by gateway and then the received information will be transfer to WISE-PaaS.

From gateway to mote :

- ① The data and command are encoded as packet by gateway.
- ② The packet is sent from gateway to mote.
- ③ The packet is received and decoded by mote.
- ④ Depend on command/data received from gateway, the information about WISE-1021 and sensor data in real time are encoded by mote and send it back to gateway.

4.3.2. Running Example

Please refer to the following to run example between mote and gateway:

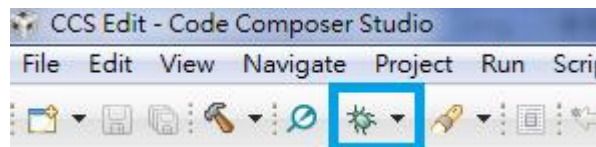
- **Running on Gateway**

There is nothing to do. This example is supported as default on WISE-3310.

- **Running on WISE-1021**

1. Compile example and programming.

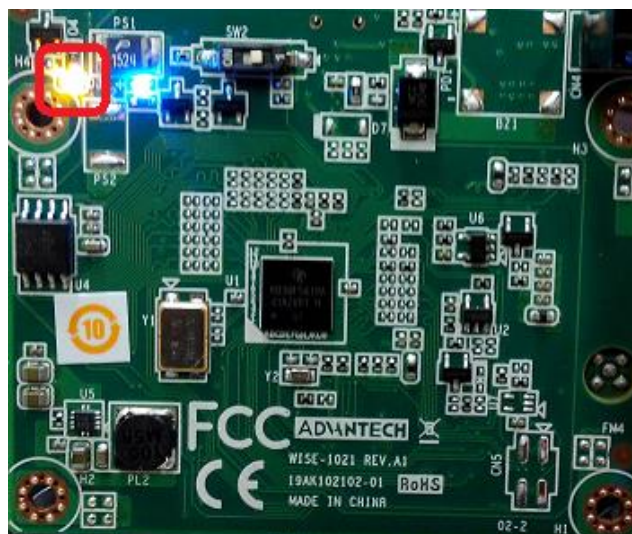
- ① Select the “[wise-1021-sensorformat](#)” project in Project Explorer, and select Project -> Build Project to build it.
- ② Click the debug icon as below to programming to the device.



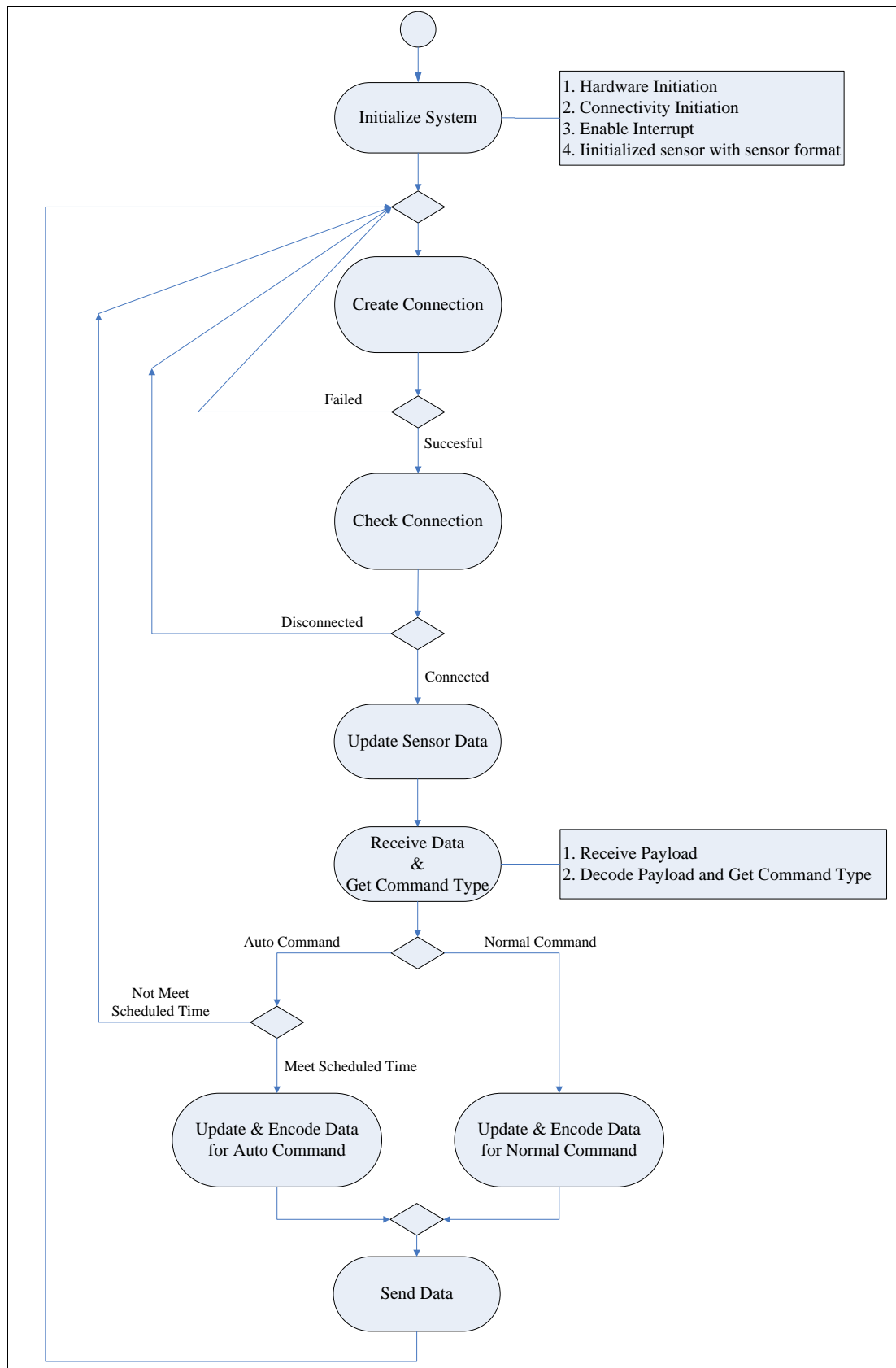
3. Check status with LED.

Waiting until the yellow LED is turned on. It's mean that WISE-1021 is getting ready to communicate with gateway.

Note: The yellow LED will be turned off when disconnected between WISE-1021 and gateway.



The flowchart is example “Exchange Data with Sensor Format” how it runs on WISE-1021 as below figure.



5. Appendix

5.1. References

5.1.1. TI

About examples and documents for TI MSP430F5419A:

© Code examples the user can find it at link

<http://www.ti.com/product/MSP430F5419A/toolssoftware>

About development tools:

© Code Composer Studio(CCS) Integrated Development Environment (IDE) at link

http://processors.wiki.ti.com/index.php/Category:Code_Composer_Studio_v6

5.1.2. Linear Dust

About SmartMesh IP Networks:

© [SmartMesh IP Mote CLI Guide](#)

© [SmartMesh IP Mote API Guide](#)