

WCT1001A/WCT1003A Automotive A13 Wireless Charging Application User's Guide

1. Key Features

The Automotive A13_Rev3 (A13_Rev3_SCH-28216_A1, A13_Rev3_LAY-28216_A) wireless charging demo is used to transfer power wirelessly to a charged device. A charged device can be any electronic device equipped with a dedicated wireless charging receiver.

The main parameters of the wireless charging transmitter (WCT) are as follows:

- The input voltage ranges from 9 V DC to 16 V DC (automotive range).
- The input voltage can drop down to 6 V DC level during the start-stop function.
- The nominal delivered power to the receiver is 5 W (at the output of the receiver).
- Designed to meet the Qi specification.
- Working frequency: 110 kHz for Qi devices.

Contents

1.	Key Features	1
2.	Hardware Setup	2
3.	Application Operation	6
4.	Hardware Description	7
5.	Application Monitoring and Control Through FreeMASTER	12
6.	Application Monitoring Through Console	19
7.	Program New Software and Calibration	20
8.	Implementation of Additional Functions	39

2. Hardware Setup

2.1 Pack content

1. WCT Automotive A13 board
2. Power supply connector
3. Power supply 12V
4. USB-UART converting board
5. Touch board

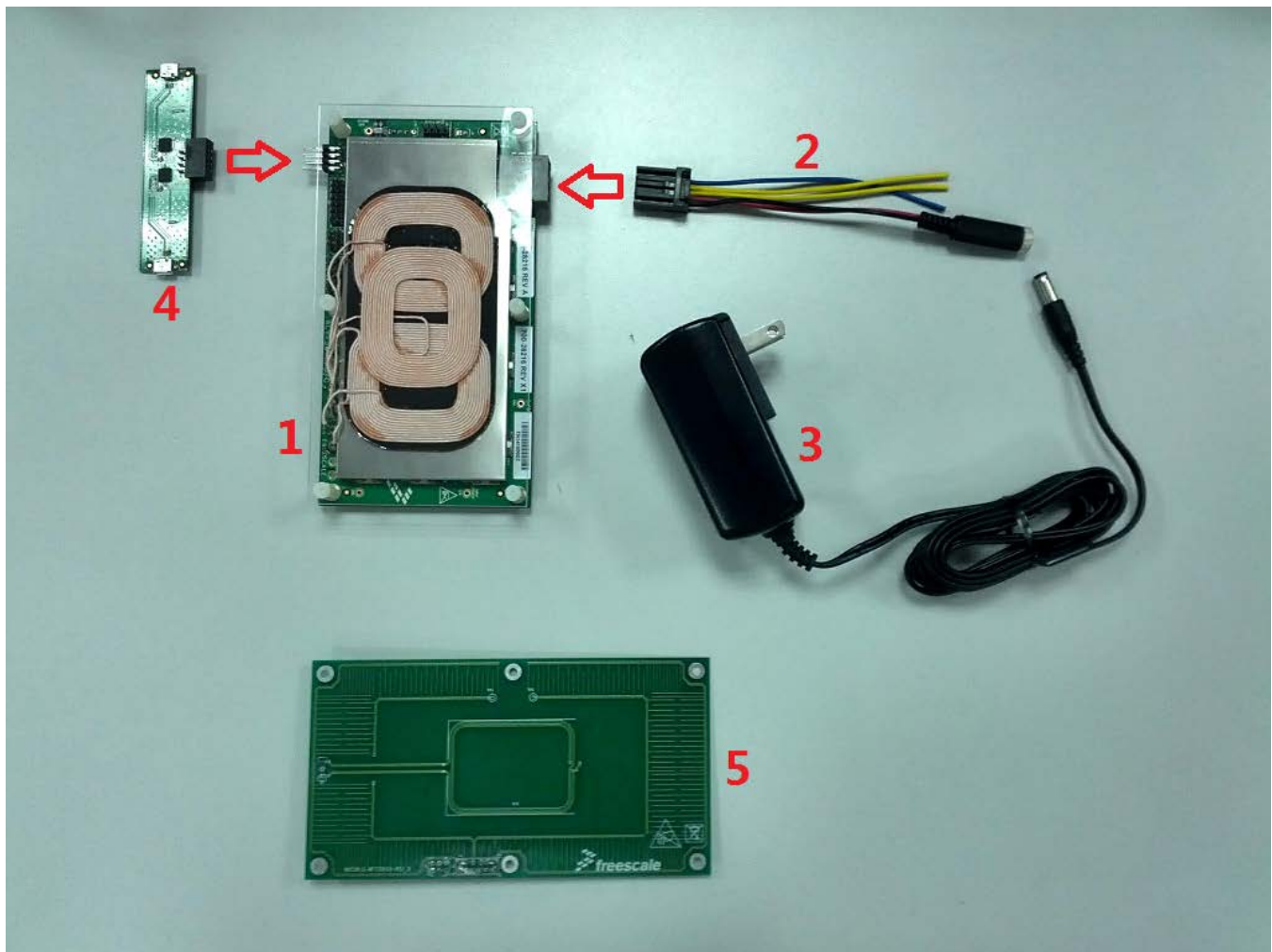


Figure 1 Hardware pack content

2.2 Board description

The WCT is connected to the system by the main power connector. It comprises the automotive battery connection (red wire = +12V line, black wire = GND line), the CAN connection (yellow wires), and the IGNITION (blue wire).

The connectors on the bottom edge of the board provide a JTAG connection for programming and debugging, 2xSCI for the FreeMASTER tool connection for the debug option and the console connection, and the temperature & touch sense to sense the temperature and placed device of the optional Touch Board that covers the coils and is the interface surface of A13.

The circuitry on the board is covered by the metal shield to lower the EMI and fix the coils position. The figure below shows the device.

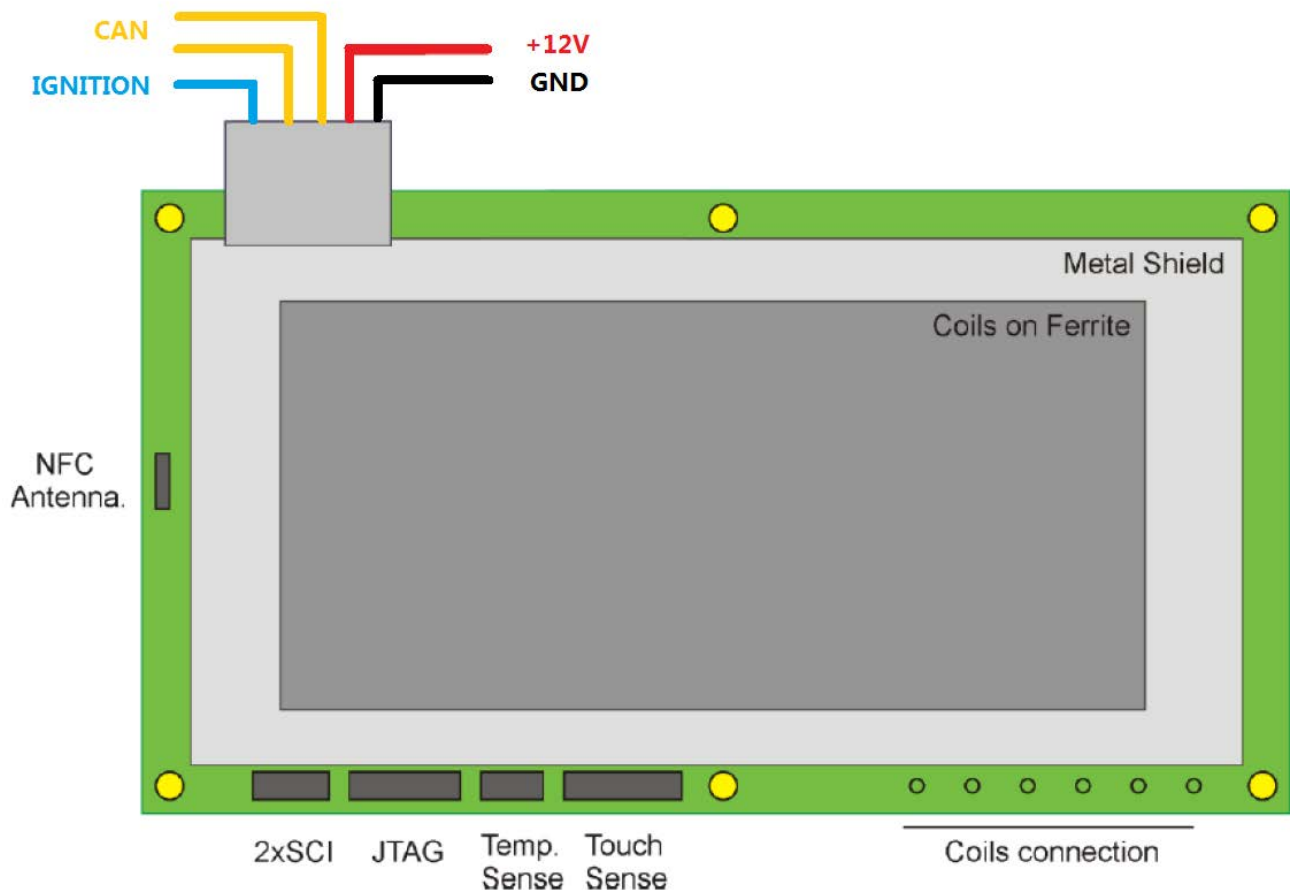


Figure 2 Device

2.3 Turning on a board

To turn on a board, perform the following steps:

1. Plug power supply 12V to the socket.
2. Plug the power supply connector to the board.
3. Connect power supply 12V and power supply connector.

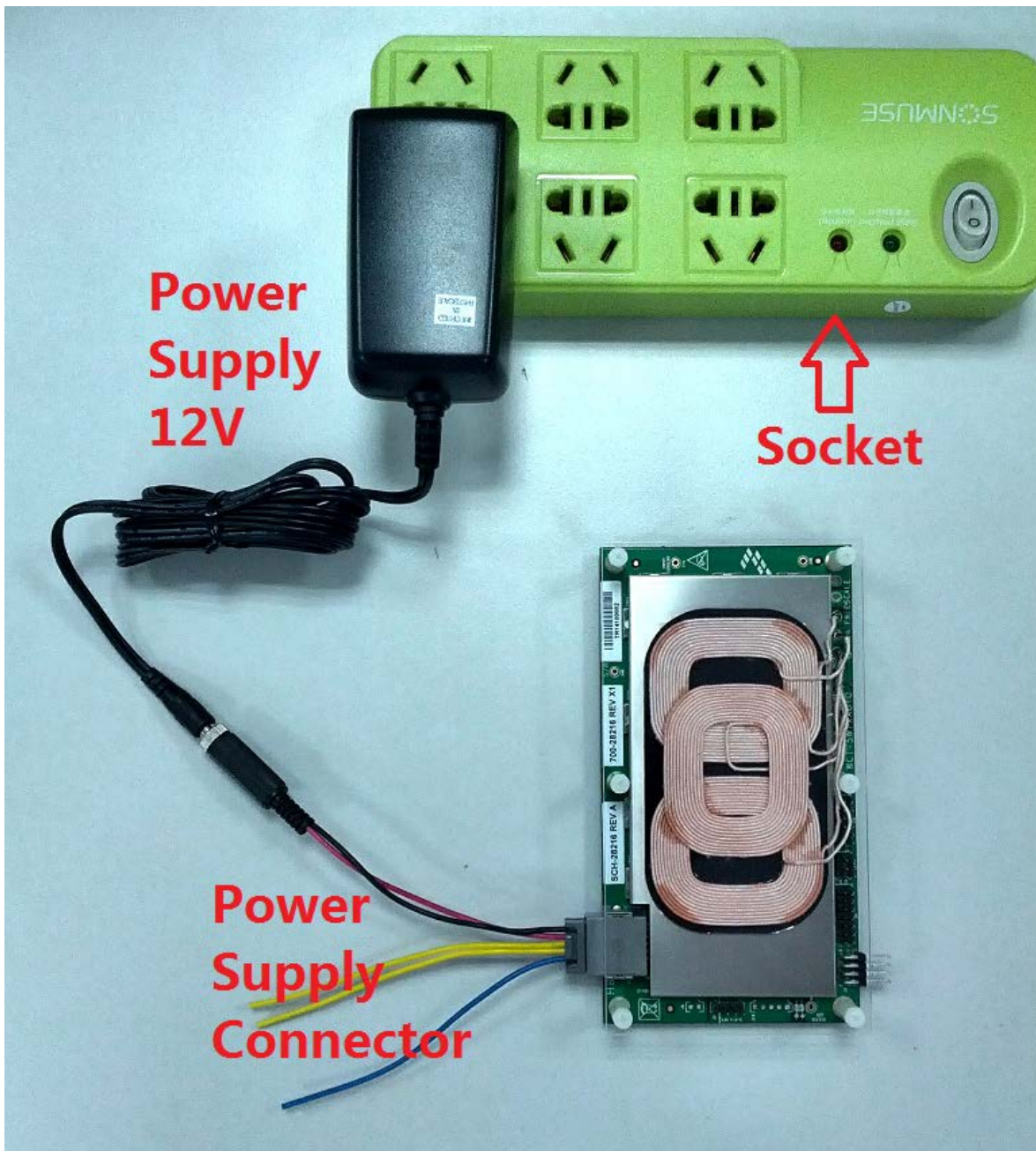


Figure 3 Power supply components

2.4 Hardware setup for FreeMASTER and Console communication

To set up the hardware for FreeMASTER and Console communication, perform the following steps:

1. Download the driver at <http://www.silabs.com/products/mcu/pages/usbtouartbridgevcpdrivers.aspx> and install CP210xVCPInstaller_x86/x64.exe on your computer.
2. Plug the USB-UART converting board to SCI connector J2. The two MicroUSB connectors are for different purposes: FreeMASTER and Console.

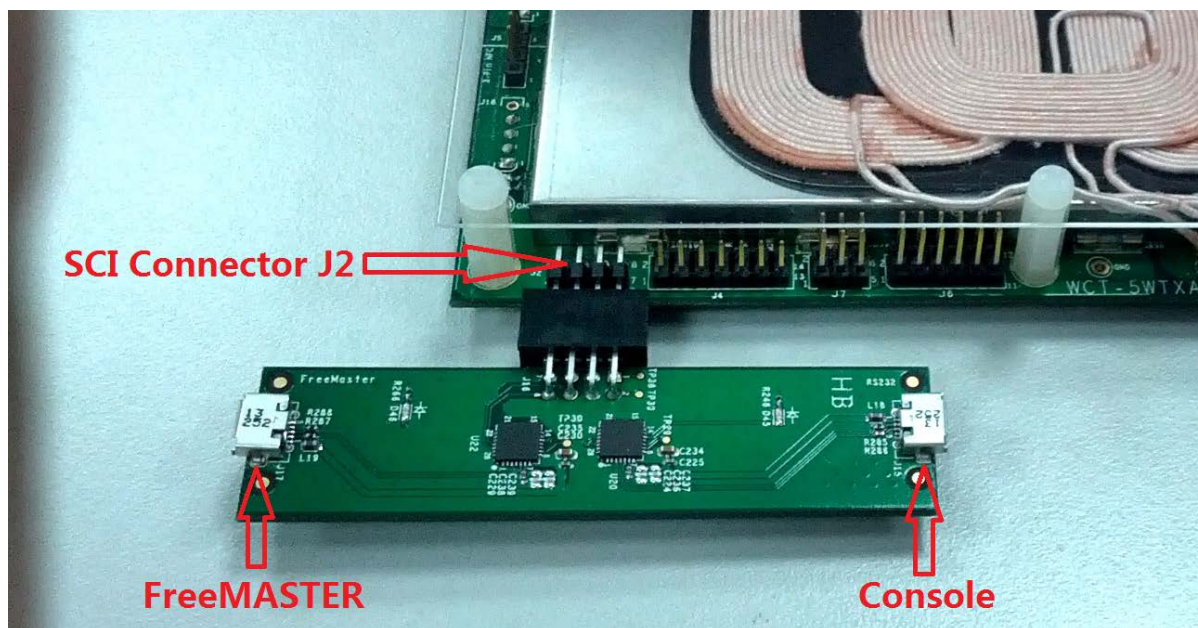


Figure 4 SCI and MicroUSB connectors

3. Application Operation

Connect the demo to the supply voltage +12 V DC. The WCT starts to send periodically the power ping to check the compatible device, wireless charging receiver (WCR), placed on the charging surface.

When the Qi compatible device is placed on the top of the coils area, the WCT starts the charging process. If there is no correct answer from the WCR side, the power transfer will not start.

If the WCR answers properly, the power transfer starts. The actual level of the transferred power is controlled by the WCT in accordance with WCR requirements. The receiver side sends messages to the WCT through the power magnetic field. If the receiver-side device is fully charged, it sends the request for power transfer termination. The power transfer is terminated if the charged device is removed from the WCT magnetic field.

There are two types of Qi WCR devices: Qi_Ver-1.0 and Qi_Ver-1.1 compatible. The main difference is that the later one provides the information to the WCT about the received power. This information serves for the Foreign Object Detection (FOD) on the top of the charging surface. The WCT calculates the difference between the power sent from the WCT and the power received by the WCR. If the difference is greater than the preset limit, the power transfer will be terminated in short time. The FOD power limit can be simply set by the application software.

4. Hardware Description

The following figure shows the block diagram of the automotive wireless charger A13.

Go to the Freescale website to obtain the latest Hardware Design files.

The whole design consists of several blocks, which are described in the following sections.

System Block Diagram

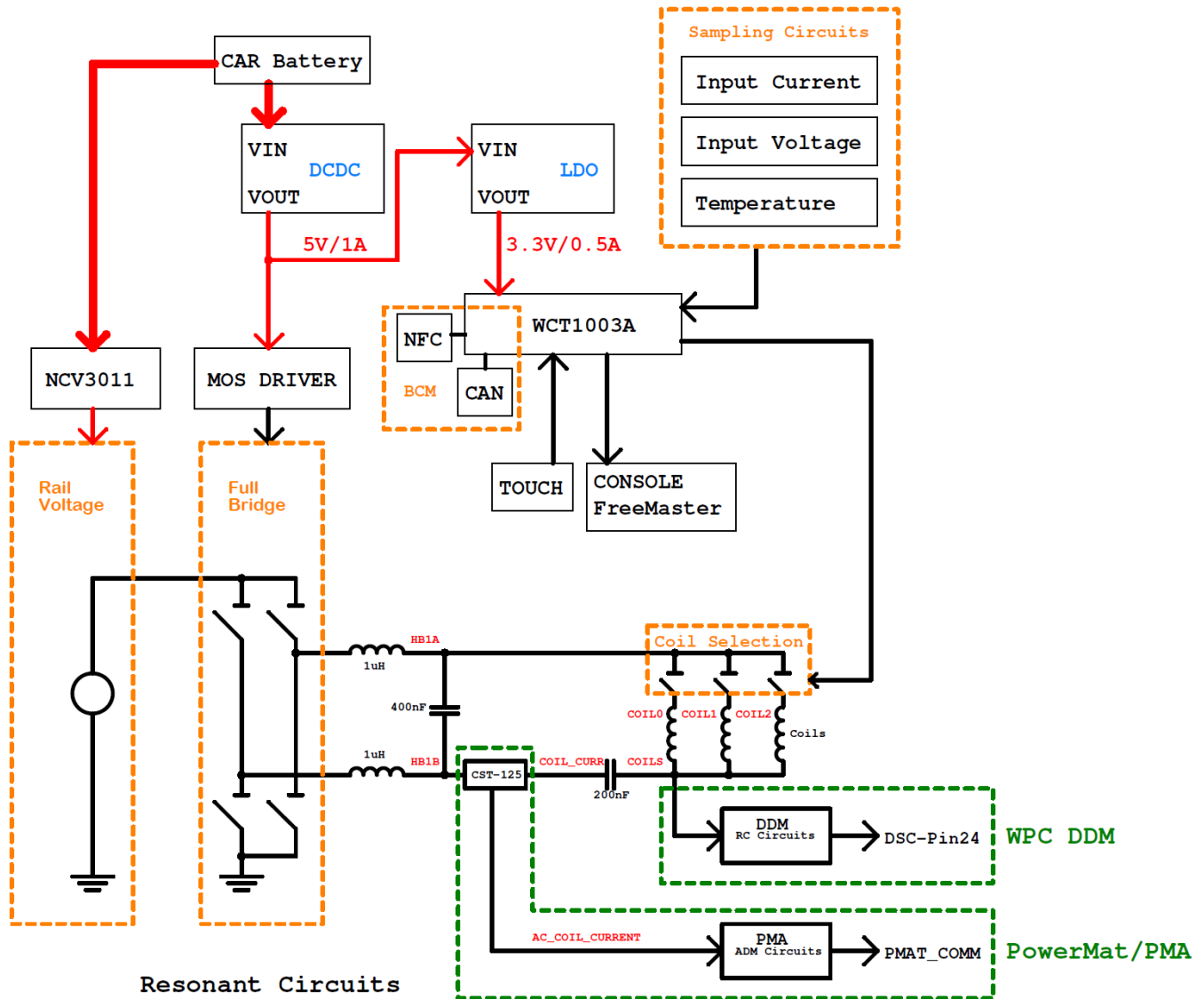


Figure 5 Block diagram of the automotive wireless charger A13

4.1 Input EMI filter

The input connector J1 provides the whole connection to the car wiring. It connects the battery voltage to the WCT and CAN communication interface. The Ignition signal is reserved if MC33907 is used.

The input filter consists of the Common Mode Filter FL1 and the filter capacitors C1, C3, C4, C14 and L1.

The main battery voltage switch is equipped with MOSFET Q1. This stage is controlled by the main controller WCT1001A/WCT1003A. The hardware overvoltage protection (more than 20 V DC) is also implemented by D1 and Q2 to this switch.

4.2 System voltage DCDC and LDO

The 12V Car Battery input is connected to a DCDC U25. Its output is 5 V and supplies LDO U26, MOSFET Driver, and CAN Transceiver. The 3.3V output of LDO is mainly for WCT1001A/WCT1003A and other 3.3 V components.

Mostly the DCDC works at the light-load conditions. High efficiency in light-load is significant important for this DCDC.

4.3 Rail voltage buck

The Qi specification for the A13 topology requires the DC voltage control of the output power. The used buck converter provides the regulated DC voltage in range from 1 V DC to 10 V DC for the full-bridge power supply.

The output Rail voltage of the buck is controlled by the analog signal RAIL_CNTL generated by the WCT1001A/WCT1003A controller. The buck is also controlled by DCDC_EN for enabling or disabling, and is monitored by DCDC_PG for fault detection.

The buck converter uses the automotive grade DC-DC synchronous buck controller followed by the power stage Q5, Q6, L2. The input and output of the DC-DC converter are blocked by the series of the low ESR ceramic capacitors.

4.4 Full-bridge and resonant circuits

The full-bridge power stage consists of two MOSFET Drivers, U8 and U9, as well as four power MOSFETs, Q13, Q15, Q19 and Q20. The MOSFET Drivers are powered by the stable voltage level 5 V DC that decreases the power losses in the drivers and MOSFETs. The full-bridge power stage converts the variable DC voltage VRAIL to the square wave 50% duty-cycle high frequency voltage with the frequency equal to 110 kHz. The range of the used frequency (105 kHz to 115 kHz) is defined in the Qi specification for the A13 topology.

The resonant circuits consist of L10, L11, C113, C114, C115, C116, C111 and C112, all of which are fixed values defined in the Qi specification for the A13 topology. The snubber RC pairs connected in parallel to power MOSFETs are used to lower the high frequency EMI products. The coil discharge circuit Q23, R96, R108 is switched ON while the coils are not energized. This circuit maintains energy-free coils while the power transfer is not active.

The Current Sense Transformer T1 is used only when PowerMat/PMA is employed.

4.5 Demodulation

It is one-way communication between the transceiver side and receiver side. The receiver measures the received power and sends back to transmitter the information about the required power level. This message is amplitude modulated (AM) on the coil current and sensed by A13.

There are two sets of demodulation circuits on A13. One is Digital DeModulation (DDM) used for the Qi compliant Receiver, and the other is Analog DeModulation (ADM) used for the PowerMat/PMA Receiver.

The RC circuits (C210, R116, R118, R224), known as DDM, sample the signals from the coil, compress the signal amplitude, and feed to ADC B-channel of WCT1001A/WCT1003A. The information about the current amplitude and modulated data are processed by the embedded software routine.

The Current Sense Transformer T1 and the classic amplitude demodulator equipped by D21 and followed with circuitry with the U16 Operational Amplifier, form the ADM part to demodulate signals from the PowerMat/PMA Receiver. The digital output signal PMAT_COMM brings the demodulated data information to the control unit WCT1001A/WCT1003A.

4.6 FOD

Required by Qi V1.1, FOD is introduced into A13. The input power to the full-bridge and output power from the coil should be calculated.

Current Sensor U21 plays the role of getting the full-bridge input current. The output power of the coil can be estimated by a specific curve-fitting way.

For details of FOD, see the *WCT1001A/WCT1003A Run-Time Debug User's Guide* (WCT100XARTDUG).

4.7 Coil selection

The Qi specification defines the A13 as the more-than-one coil topology with one coil energized at a time.

The coil selection topology connects one coil only to resonant circuits at a time. The coil is equipped with the dual N-MOSFETs, Q9, Q12, or Q16, controlled by the WCT1001A/WCT1003A controller through the control interface based on the low power bipolar transistors.

4.8 Analog sensing

Some ports of the ADC A-channel of WCT1001A/WCT1003A are used for sensing analog signals, such as temperature, full-bridge input current, input voltage, and Rail voltage.

4.9 Touch sensing

An accessory touch board is included in the hardware package of A13.

After the Plastic Board is replaced by Touch Board, and TOUCH is enabled in the software, any object placed on the top of the Touch Board, can be sensed by the WCT1001A/WCT1003A GPIO port. The GPIO Touch function is based on the capacitance change on the Touch Electrodes.

For a better power consumption consideration, you can use a dedicate Touch Controller to free the WCT1001A/WCT1003A when waiting for a touch event.

4.10 Control unit

The control unit WCT1001A/WCT1003A is the heart of the whole application. This controller runs to code based on the dedicated wireless charging software library. It controls the whole wireless power transfer and runs other customer's tasks.

The following figure shows the Functional Block Diagram of WCT1001A/WCT1003A. The whole control consists of several blocks, which are described in the following sections.

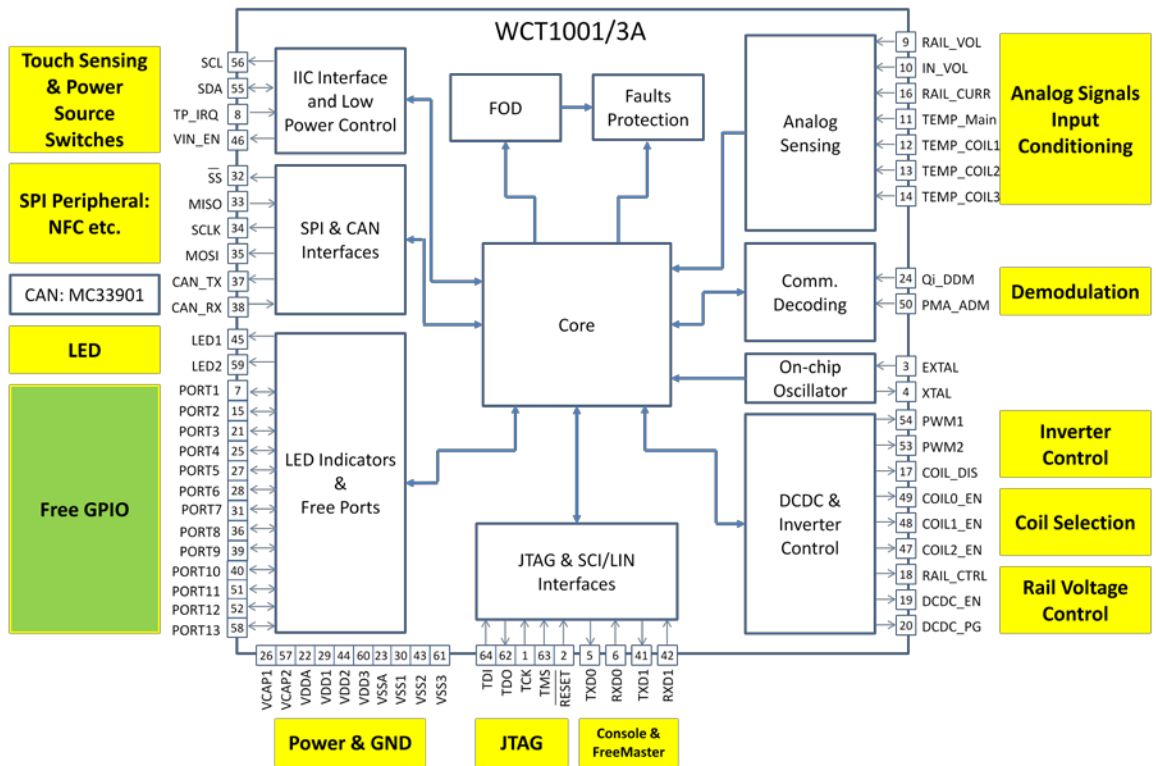


Figure 6 WCT1001A/WCT1003A system functional block diagram

- Touch Sensing: see Section 4.9 “Touch sensing”.
- Power Source Switch: see Section 4.1 “Input EMI filter”.
- SPI Peripheral: used to connect with an SPI peripheral, such as NFC.
- CAN Transceiver: to connect with the CAN transceiver.
- LED: outputs for the signal LEDs.

- Analog Signals Input Conditioning: see Section 4.8 “Analog sensing”.
- Demodulation: see Section 4.5 “Demodulation”.
- Inverter Control: see Section 4.4 “Full-bridge and resonant circuits”.
- Coil Selection: see Section 4.7 “Coil selection”.
- Rail Voltage Control: see Section 4.3 “Rail voltage buck”.
- JTAG: recommended to keep them as JTAG, not to use as GPIO.
- Console and FreeMASTER: serial communication interface for the Console and FreeMASTER.
- Power and GND: VCAP1 and VCAP2 are used for internal core circuits, requiring external capacitors.
- Free GPIOs: can be used freely by customers except for caring use of the ADC input.

5. Application Monitoring and Control Through FreeMASTER

FreeMASTER is a user-friendly real-time debug monitor and data visualization tool for application development and information management. Supporting nonintrusive variable monitoring on a running system, FreeMASTER allows the data from multiple variables to be viewed in an evolving oscilloscope-like display or in a common text format. The application can also be monitored and operated from the web-page-like control panel.

5.1 Software setup

To set up the software, perform the following steps:

1. Install the FreeMASTER V1.4 or later from the Freescale website: freescale.com/freemaster
2. Plug the USB-UART converting board to SCI connector J2, and connect the FreeMASTER MicroUSB port to your computer.
3. Open the Device Manager, and check the number of the COM port.

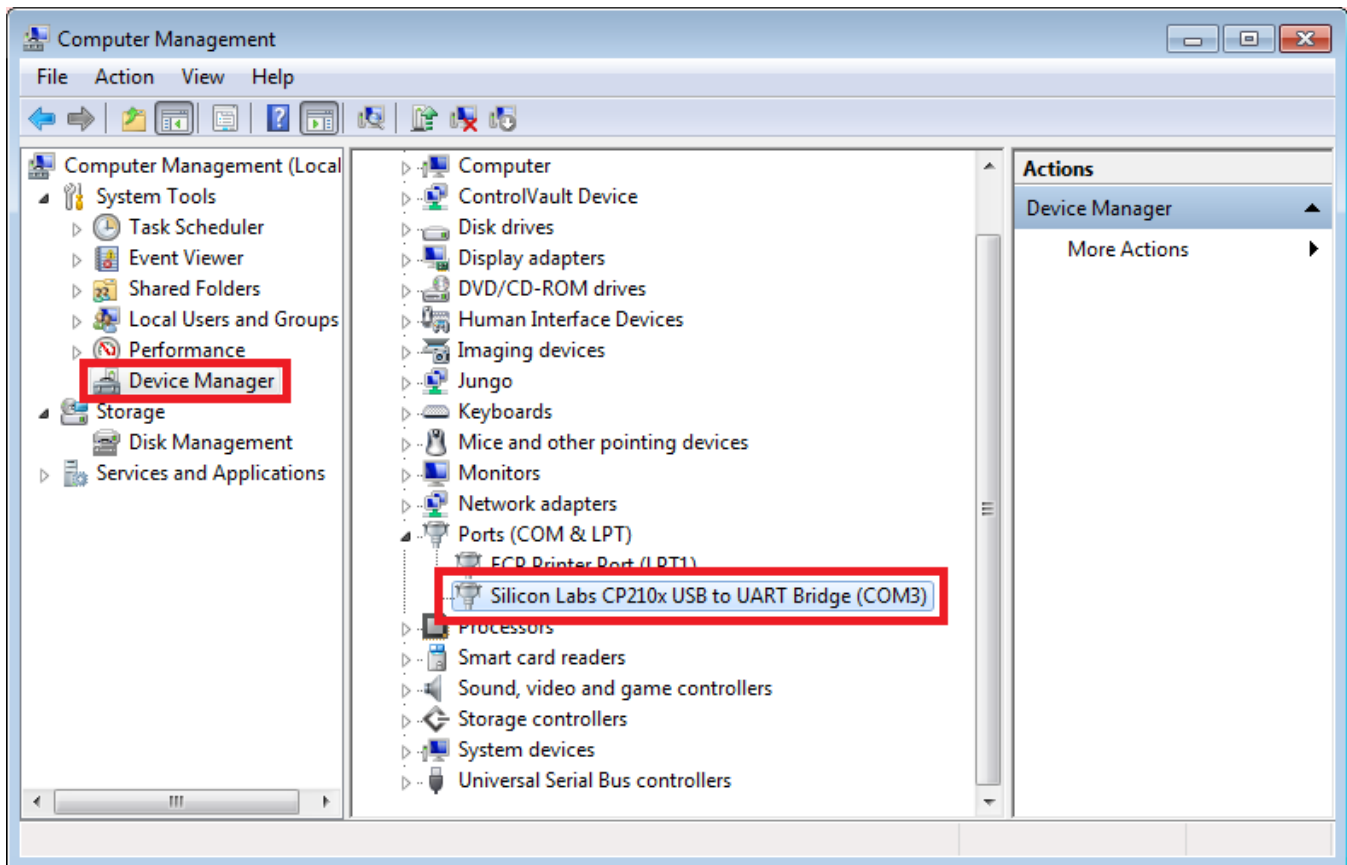


Figure 7 Device manager

4. Unpack the embedded source code to your local disk.
5. Start the FreeMASTER application by opening:

- MWCT1003A
 <unpacked_files_location>/example/WCTxxxx/WCTAutoA13_WCT1003A/WCTAutoA13_WCT1003A.pmp
 - MWCT1001A
 <unpacked_files_location>/example/WCTxxxx/WCTAutoA13_WCT1001A/WCTAutoA13_WCT1001A.pmp
6. Choose **Project** -> **Options**.

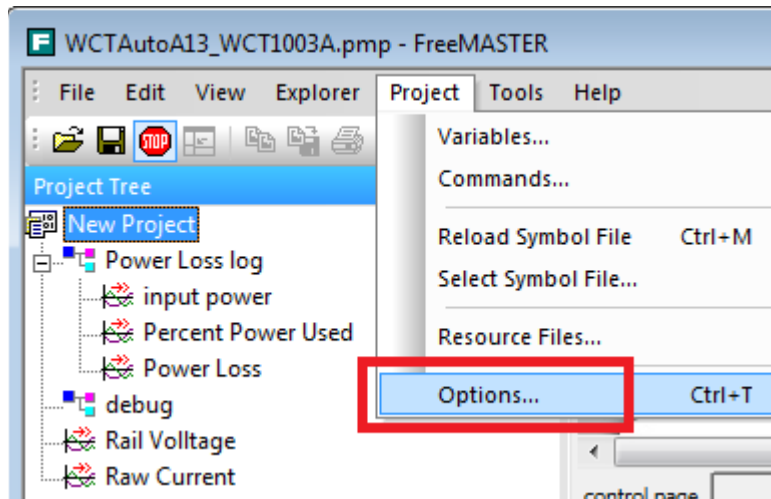


Figure 8 Choosing Options

7. Ensure that the correct virtual **Port** (according to Step 3) and **Speed** is selected.

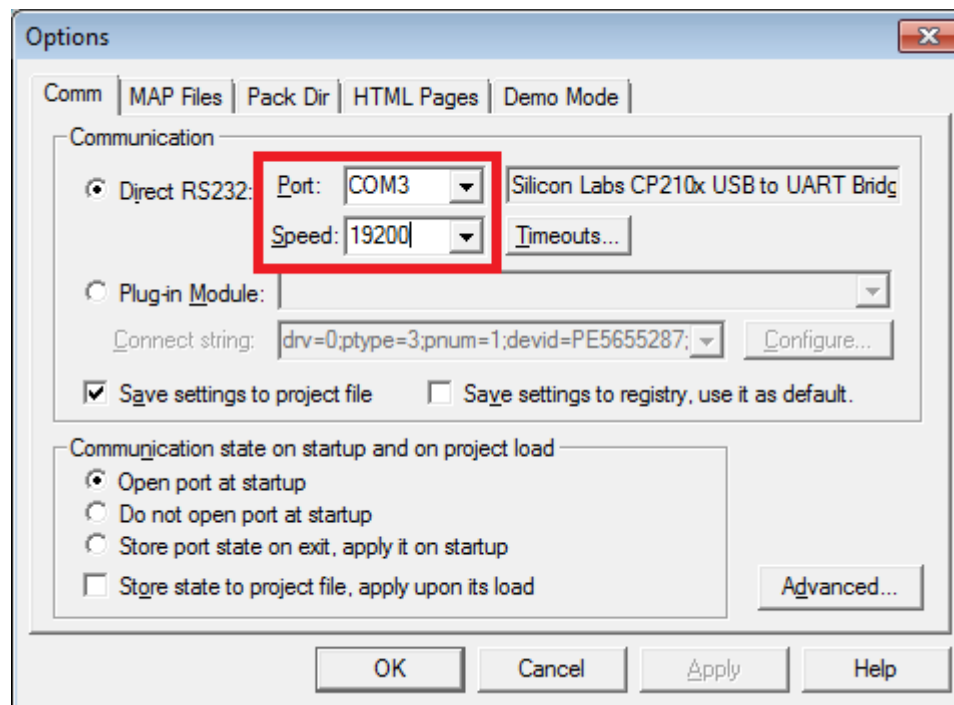


Figure 9 Setting Port and Speed

8. Power on A13, and then start the communication by clicking the **STOP** button on the FreeMASTER.

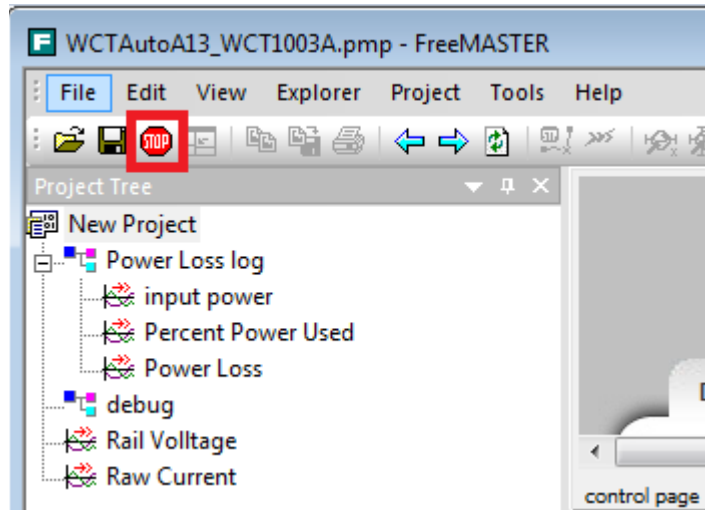


Figure 10 Stop button

5.2 Real-time application variables monitoring

FreeMASTER enables to monitor and update all the application global variables. In this application, several key variables are displayed in the scope windows. The user can observe the following variables in real time during charging or idle mode:

- Input power
- Power used
- Power loss
- Rail voltage
- Raw current

Particular charts are accessible by clicking on the name of the scope window.

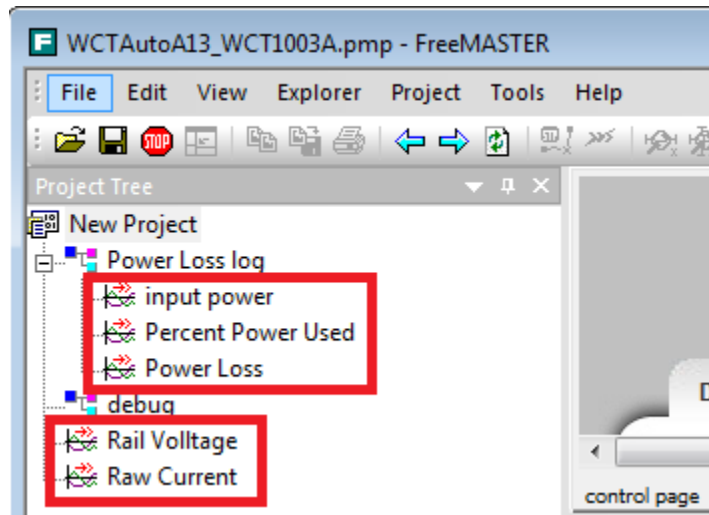


Figure 11 Real-time application variables

5.3 Application parameters modification

The application parameters can be easily viewed and changed on the control panel. The control panel contains the web page elements (buttons, check boxes, text fields) that enable a user-friendly way to visualize and change the application control parameters.

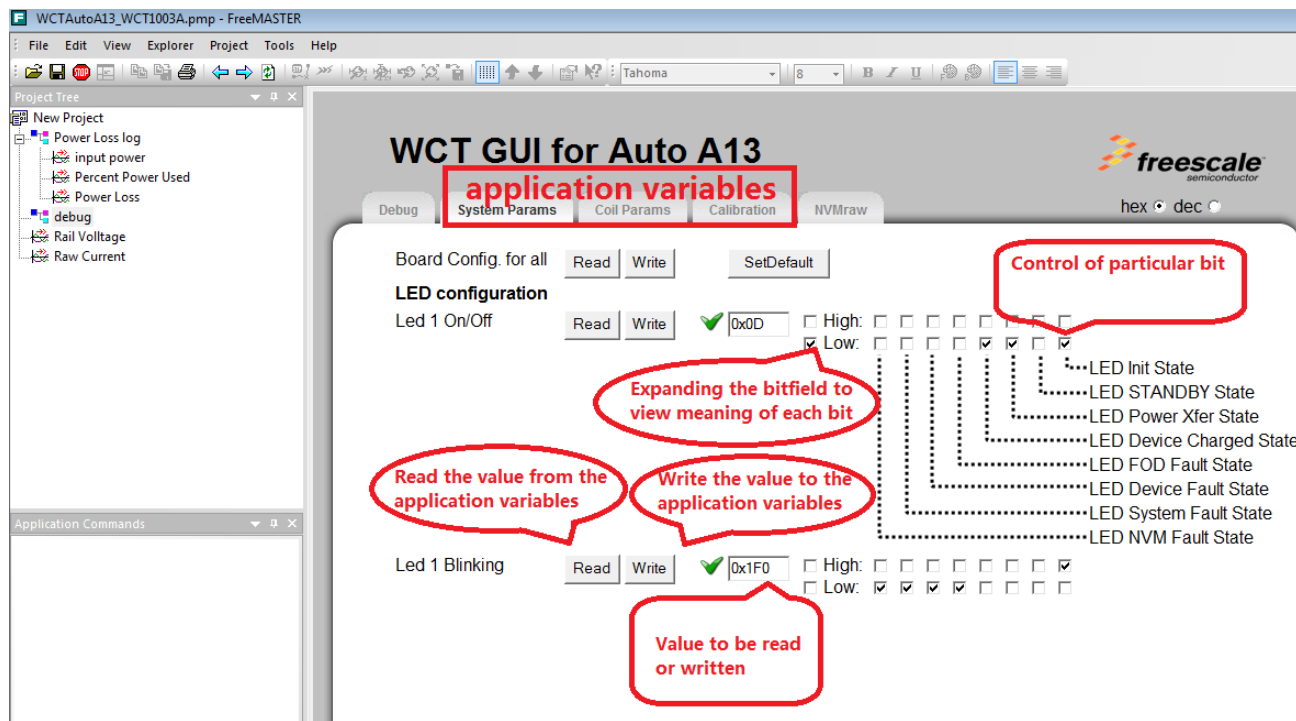


Figure 12 Application variables

The application variables are divided into three tabs:

- **System Params** – group of general system parameters
- **Coil Params** – enables access to the variables related to the coils control
- **Calibration** – group of parameters for calibration of the input current, input voltage, and foreign objects detector

The meaning of each parameter is described next to the text field.

5.4 Debug mode

The most left **Debug** tab contains the control elements to trigger multiple events manually:

- Key Fob Avoidance Trigger
- Turn on/off the battery switch
- Enable debug mode of the WCT library
- Change coil frequency
- Change rail voltage
- Perform rail voltage calibration, input current calibration, or calibration of Foreign Object Detector

Before changing the parameters, click **Enter** next to **Debug Mode** to put the system to debug mode. If Touch Sensing is used, to avoid putting system to sleep, scroll down to the lower part of the window, write **255** to **byTouchTimeout**, touch the Touch Board by finger, and press **Enter** of your keyboard.

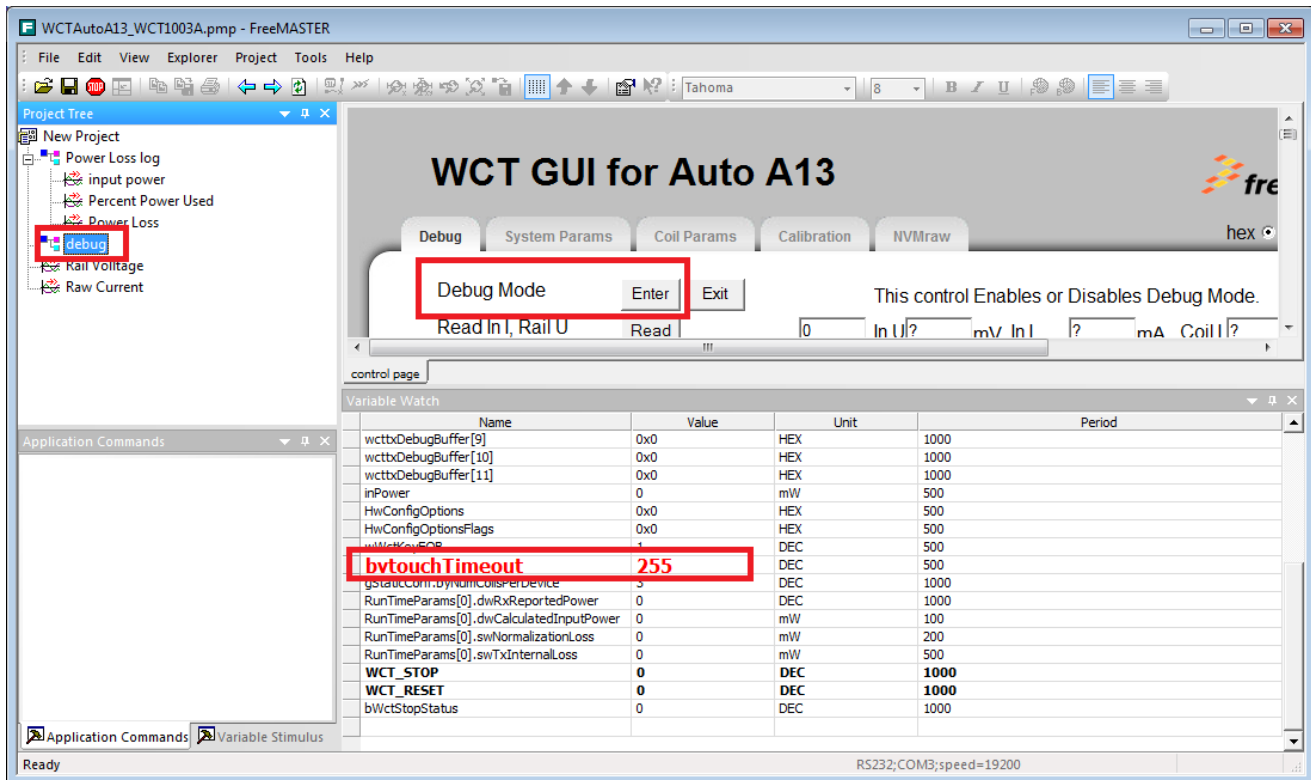


Figure 13 Setting debug mode

The following figure shows the **Debug** tab of the control page. The control elements have the same meaning as on the previously described tabs.

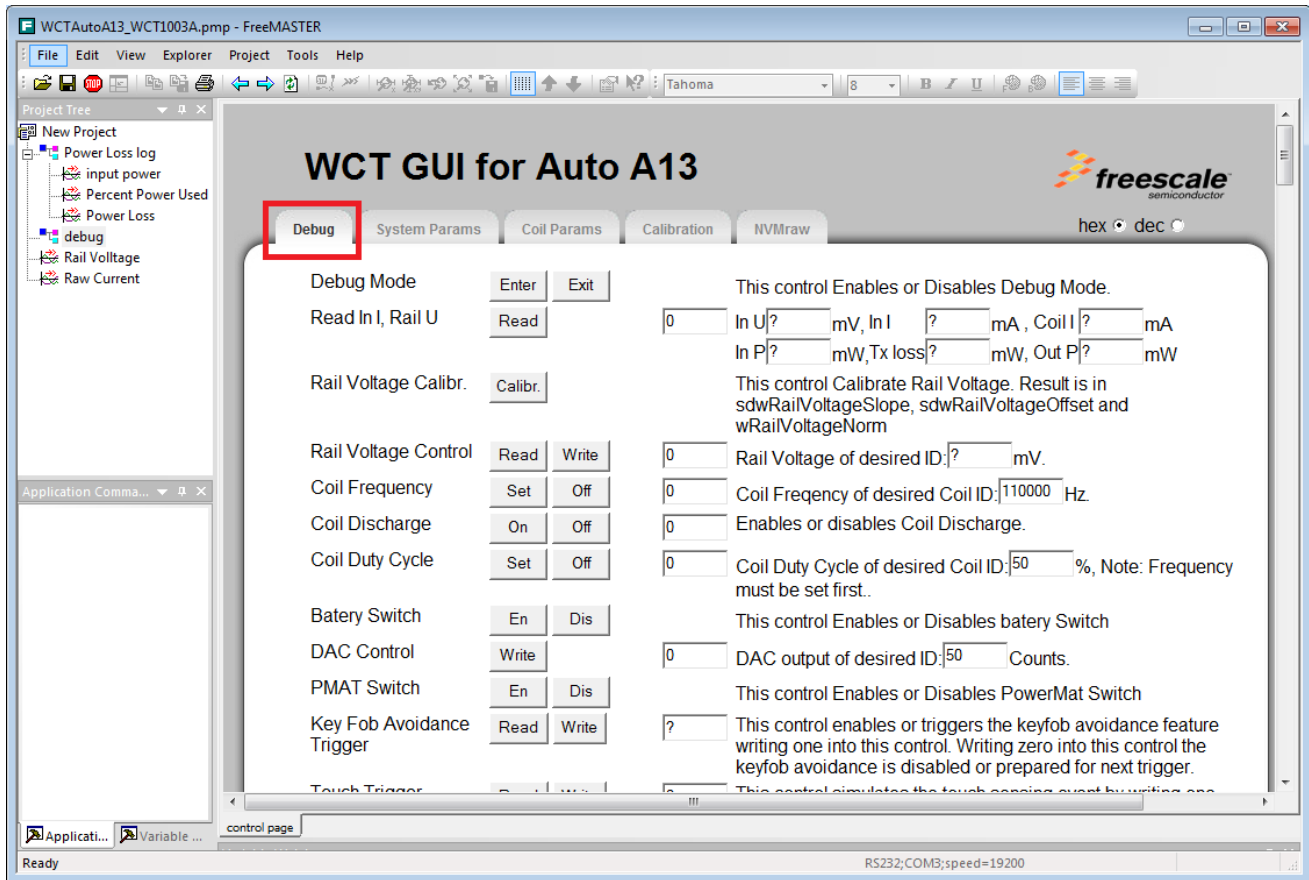


Figure 14 Debug tab

6. Application Monitoring Through Console

The application sends some information and error states through SCI to the console. The information is sent when the board is turned on, when the device is charging, or in case of some error state.

6.1 Software setup

1. Plug the USB-UART converting board to SCI connector J2, and connect the console MicroUSB port to your computer.
2. Open the **Device Manager**, and check the number of the COM port.

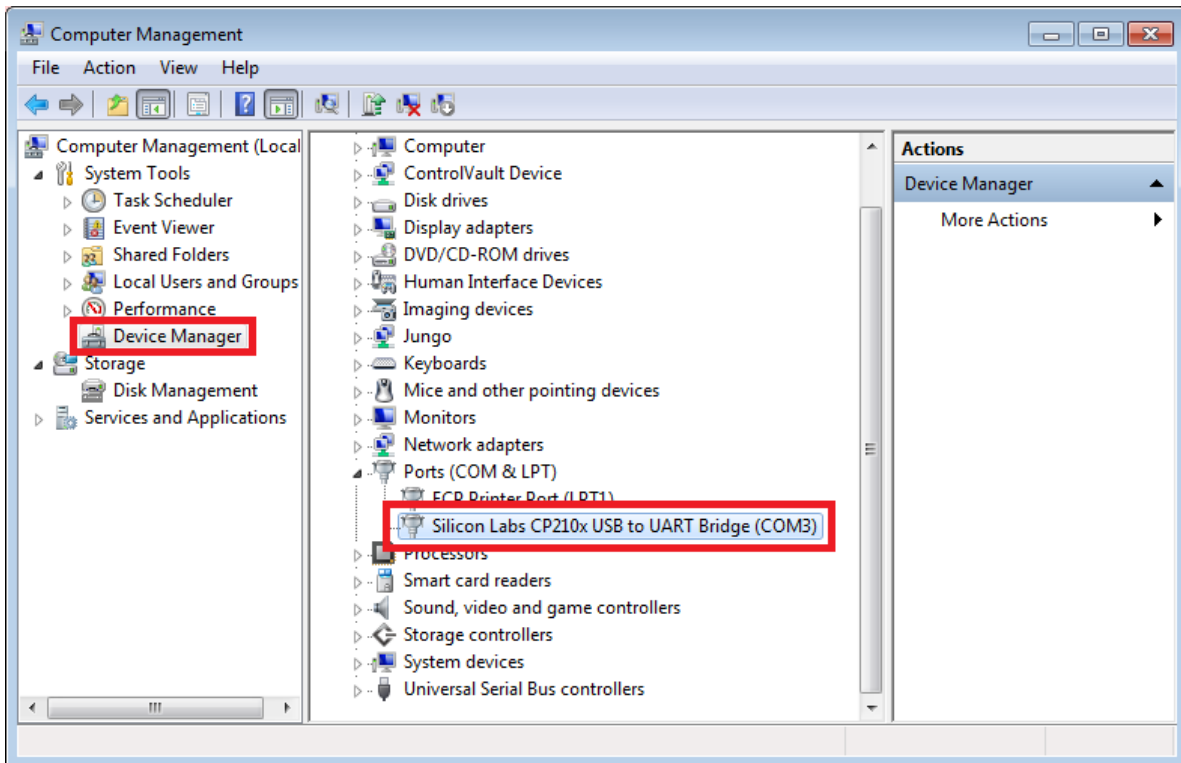


Figure 15 Device Manager

3. Run the communication program supporting console, such as HyperTerminal or RealTerm.
4. The following table shows the communication setup.

Port number	Serial port from Device Manager
Baud	19200
Data Bits	8
Stop Bits	1
Parity	None
Hardware Flow Control	None
Display As	ASCII

- Open the port or start communication, which depends on the used Terminal.

7. Program New Software and Calibration

Freescale provides users a software package, which includes a WCT1001A/WCT1003A project and a Bin file (.S). Users can flash alternative to the board. After flashing new software, board calibration must be carried out.

7.1 Install CodeWarrior 10.6 or later

- Download installation files.

For proper installation of CodeWarrior 10.6 or later, you need to install both **CodeWarrior for Microcontrollers 10.x** and **CW MCU v10.x Wireless Charging MWCT1xxx Service Pack**, and make sure that both files are with the same revision.

- CodeWarrior for Microcontrollers 10.6**

http://www.freescale.com/webapp/sps/site/prod_summary.jsp?code=CW-MCU10&fsp=1&tab=Design_Tools_Tab

- CW MCU v10.6 Wireless Charging MWCT1xxx Service Pack**

http://www.freescale.com/webapp/sps/site/overview.jsp?code=CW_UPDATES_MCU_10_6&fsrch=1&sr=1&pageNum=1

In addition, there are two kinds of CodeWarrior:

- Evaluation Software**

Evaluation Editions are available for free download and work for a limited time, that is, the tools should be installed with all the features available for 30 days.

- Special Edition Software**

Special Editions are fully functional free download versions of the CodeWarrior Development Studio with code size restrictions on the build chain.

- Double-click **Setup.exe** after downloading.

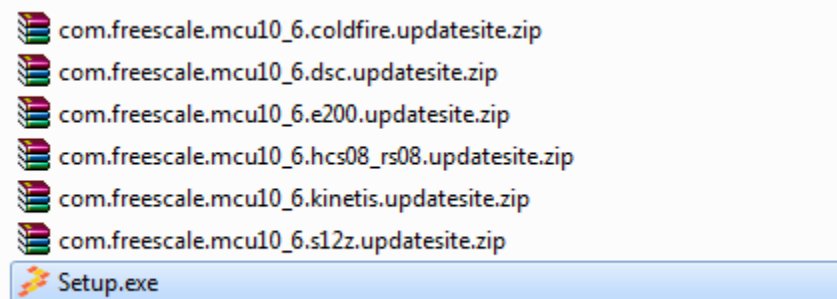


Figure 16 Setup file

3. Make sure that **DSC** is selected.

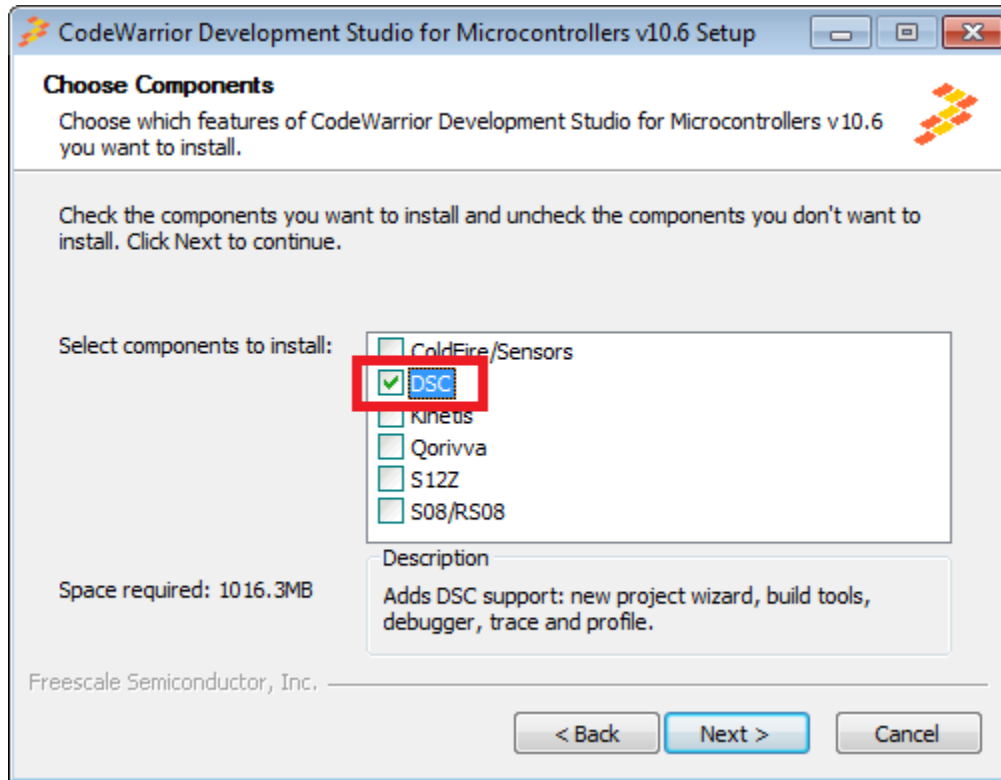


Figure 17 DSC installed

- Click **Allow access** when the **Windows Security Alert** dialog box is displayed.



Figure 18 Windows Security Alert dialog box

- Select **Always trust software from “Freescale Semiconductor”**, and then click **Install**.

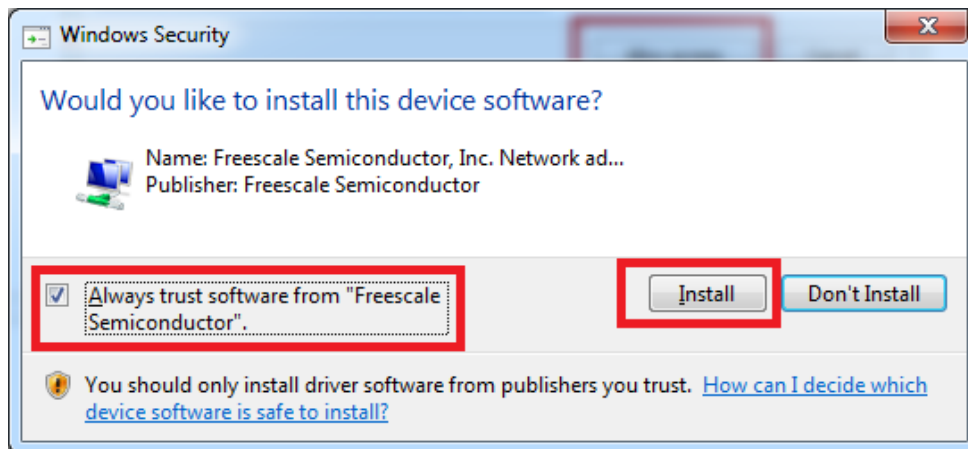


Figure 19 Windows Security dialog box (1)

6. Select **Always trust software from “Jungo LTD”**, and then click **Install**.

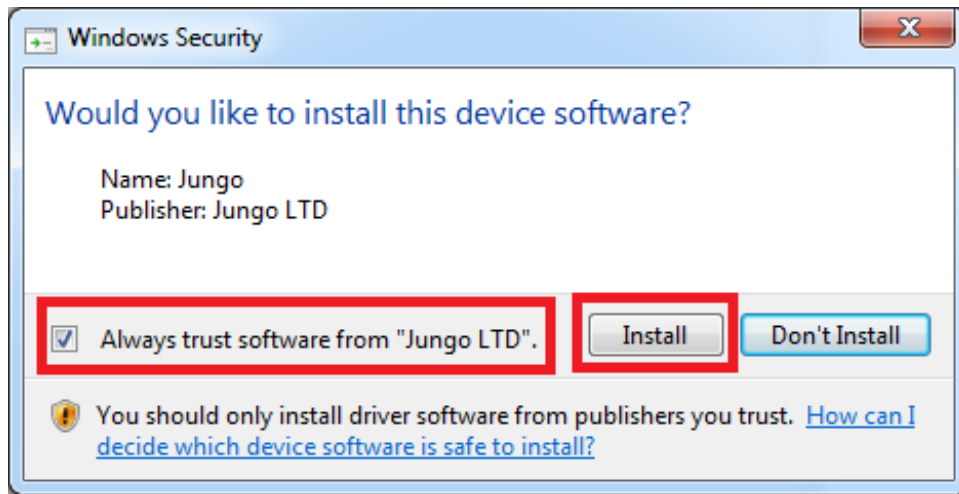


Figure 20 Windows Security dialog box (2)

7. Select **Always trust software from “PE Microcomputer System Inc”**, and then click **Install**.

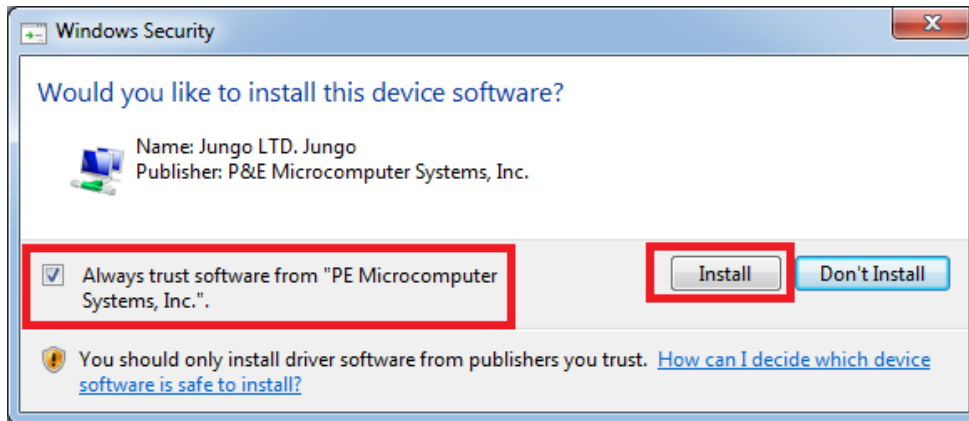


Figure 21 Windows Security dialog box (3)

- Launch CodeWarrior, create a folder workspace, and select it as the default workspace.

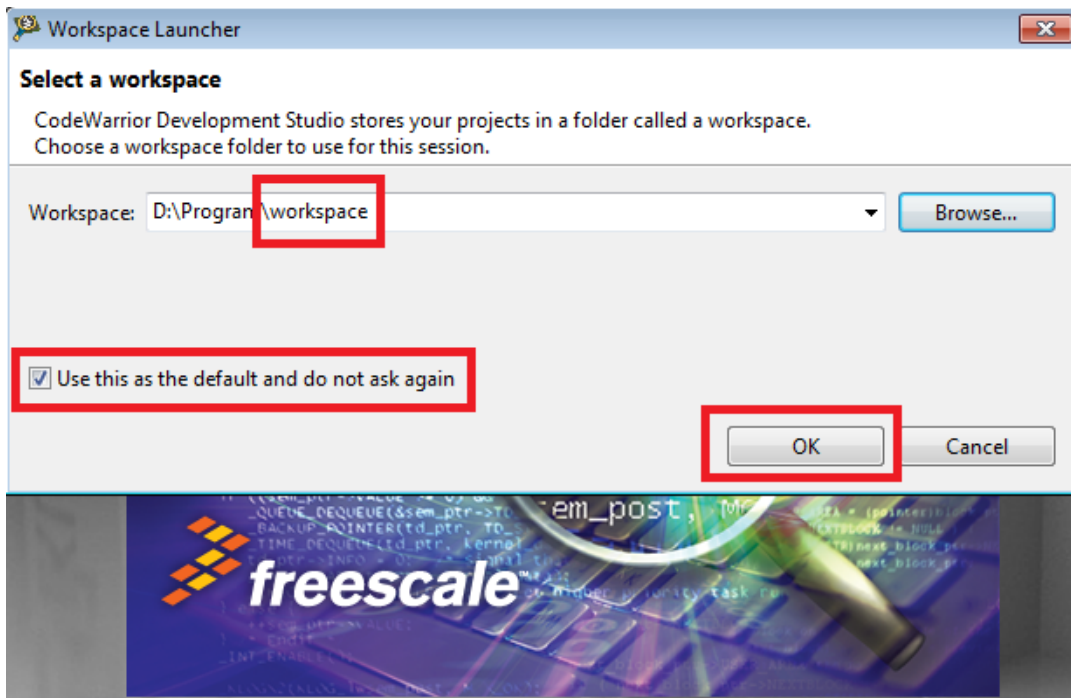


Figure 22 Workspace Launcher dialog box

- Choose **Help** → **Install New Software**.

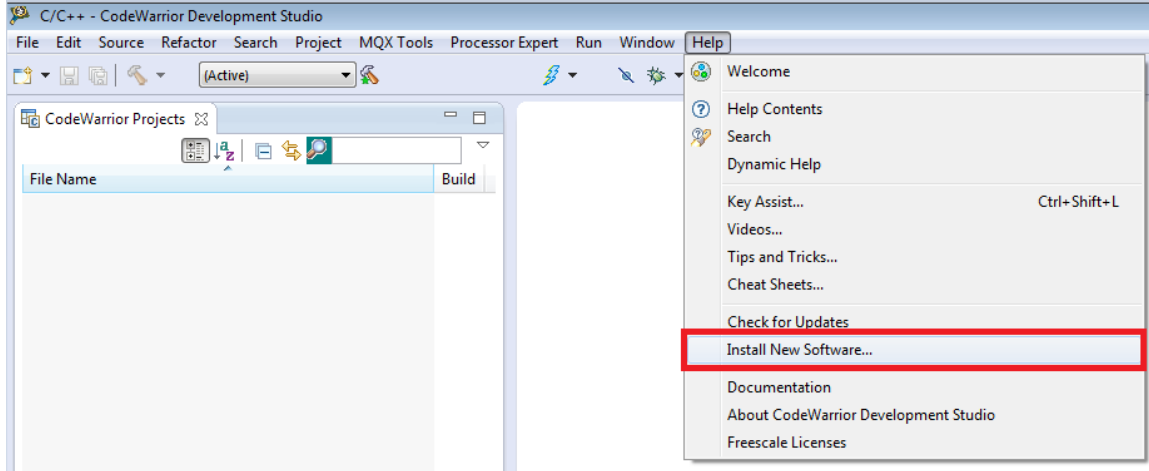


Figure 23 Install New Software

- Click **Add** and **Archive**, and then select **com.freescale.mcu10_6.WCT_WCT100x_WCT1100x_WCT1200x.win.sp.v1.0.1.zip** (CW MCU v10.6 Wireless Charging MWCT1xxx Service Pack).

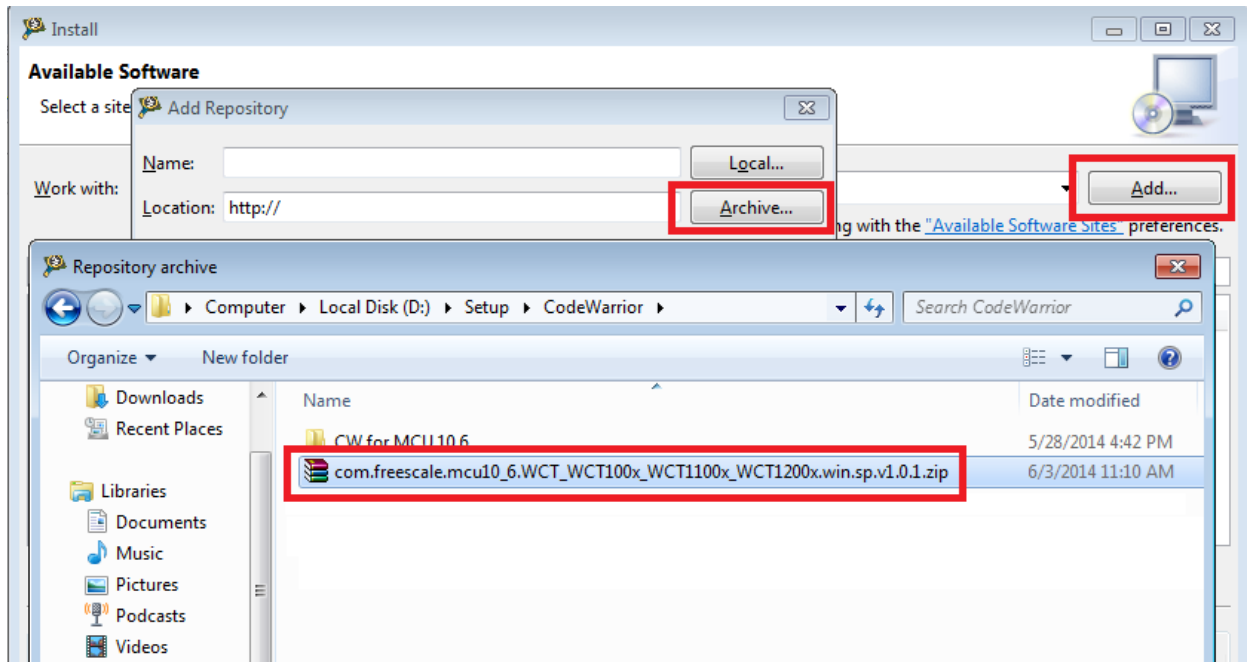


Figure 24 Selecting the service pack

11. Select **MCU v10.6 DSC Service Packs**, and then click **Next**.

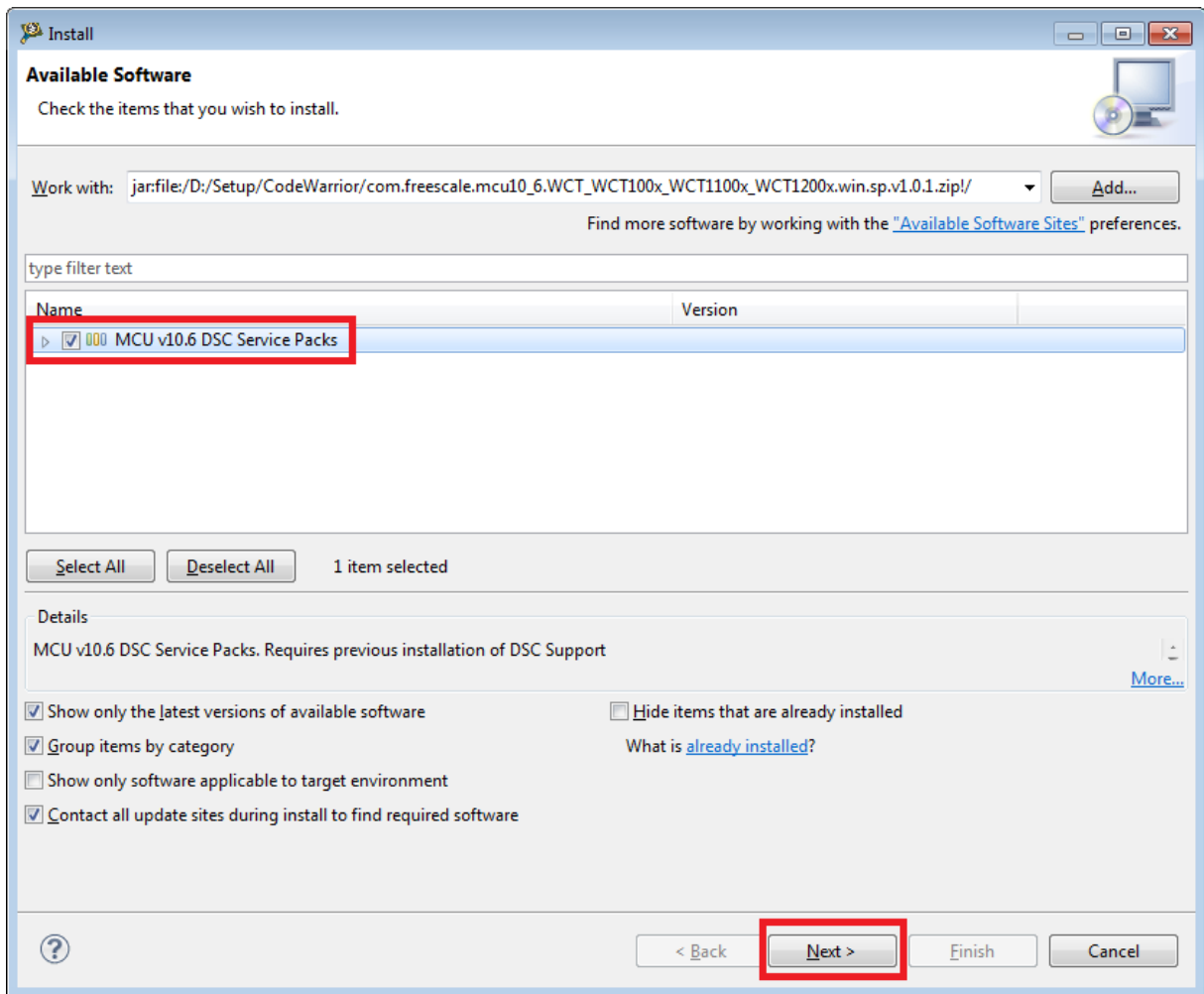


Figure 25 Selecting MCU v10.6 DSC Service Packs

12. Click **Next**.

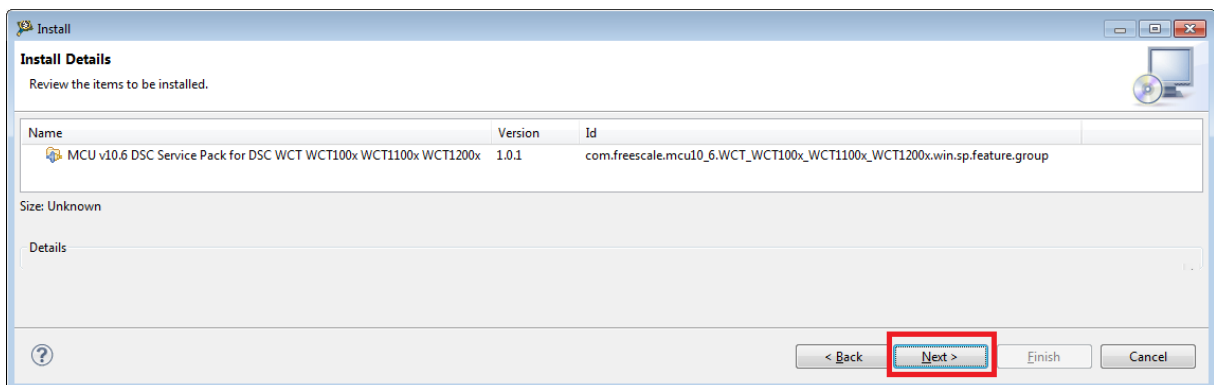


Figure 26 Installing the pack

13. Select **I accept the terms of the license agreement**, and then click **Finish**.

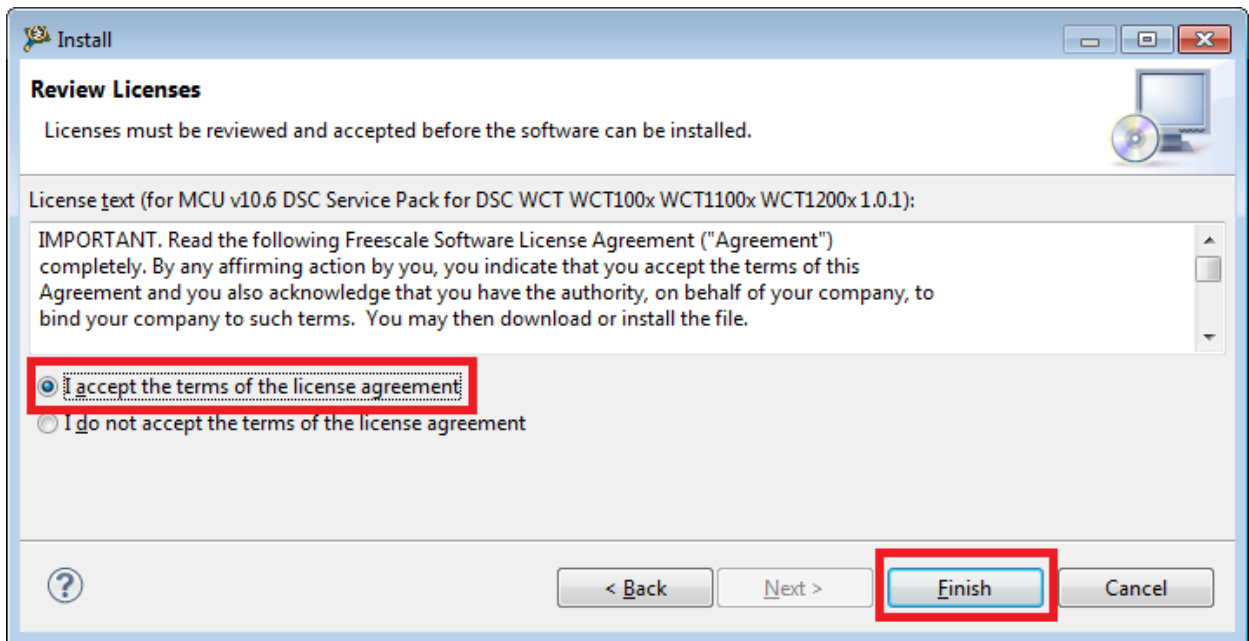


Figure 27 Installation finished

7.2 Board and programmer connection

1. Connect **FSL USB TAP ONCE** or **PnE U-MultiLink** to your PC and install the driver.

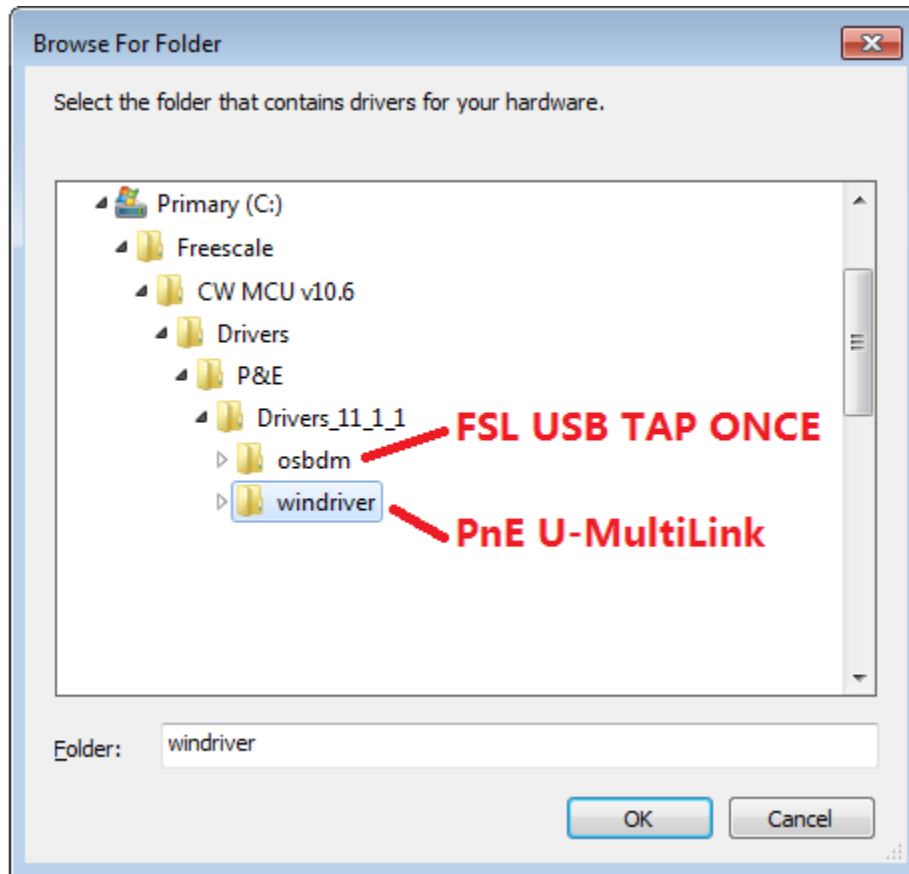


Figure 28 Browse for Folder

2. Connect the 14-pin debug cable to J4 of the board (notice pin-1 position of cable).



Figure 29 Connecting the debug cable to the board

7.3 Program using the project file

1. Drag the **.project** file to the **CodeWarrior Projects**.

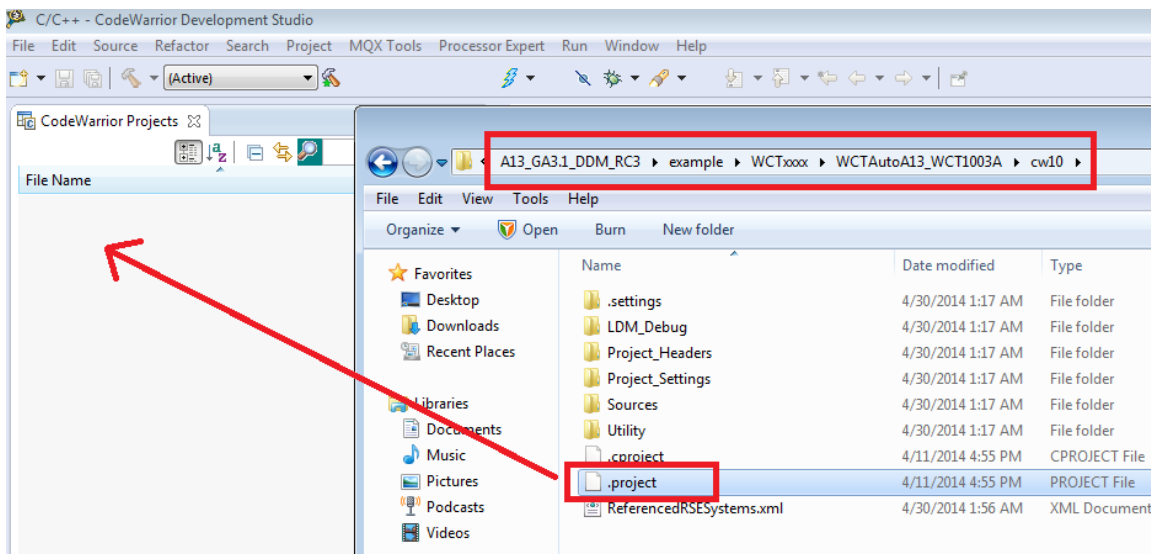


Figure 30 Dragging the project file

- Choose **WCTAutoA13_WCT1003A** → **LDM_Debug**. Select **SDM_Debug** if the chip is **MWCT1001A**.

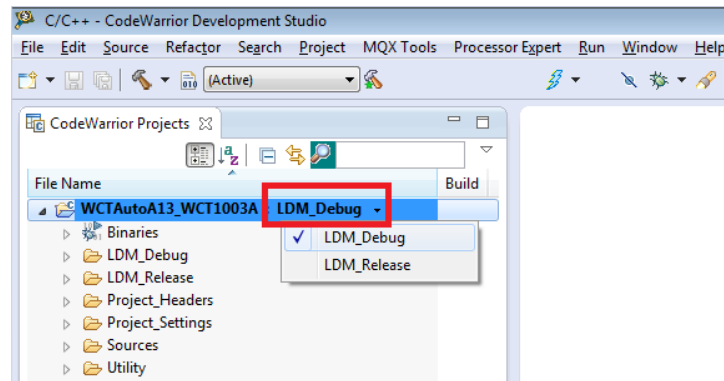


Figure 31 LDM_Debug

- Right-click **WCTAutoA13_WCT1003A**, and then choose **Clean Project** and **Build Project**.

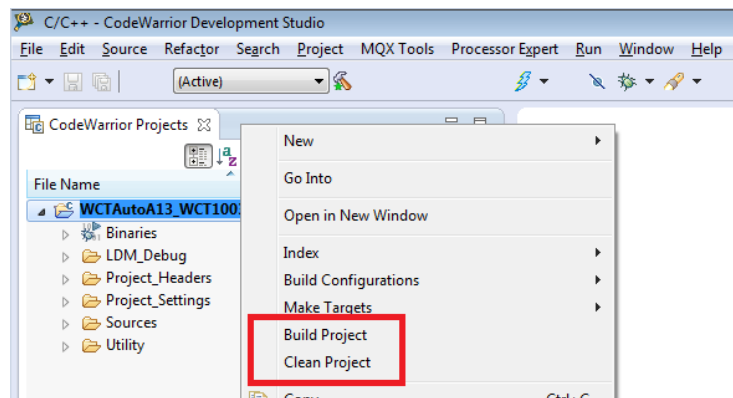


Figure 32 Clean Project and Build Project

- Power on A13 and choose **Run** → **Run Configurations**.

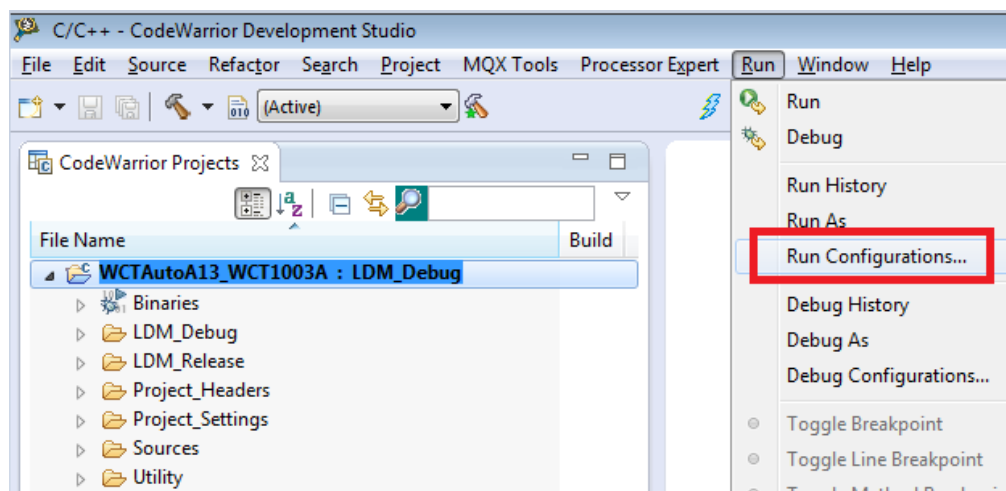


Figure 33 Run Configurations

5. Select **WCTAutoA13_WCT1003A_LDM_Debug_FSL USB TAP**, and click **Run**.
 Select **WCTAutoA13_WCT1001A_SDM_Debug_FSL USB TAP** if the chip is **MWCT1001A**.
 Select **PnE U-MultiLink** if the programmer is **MultiLink**.

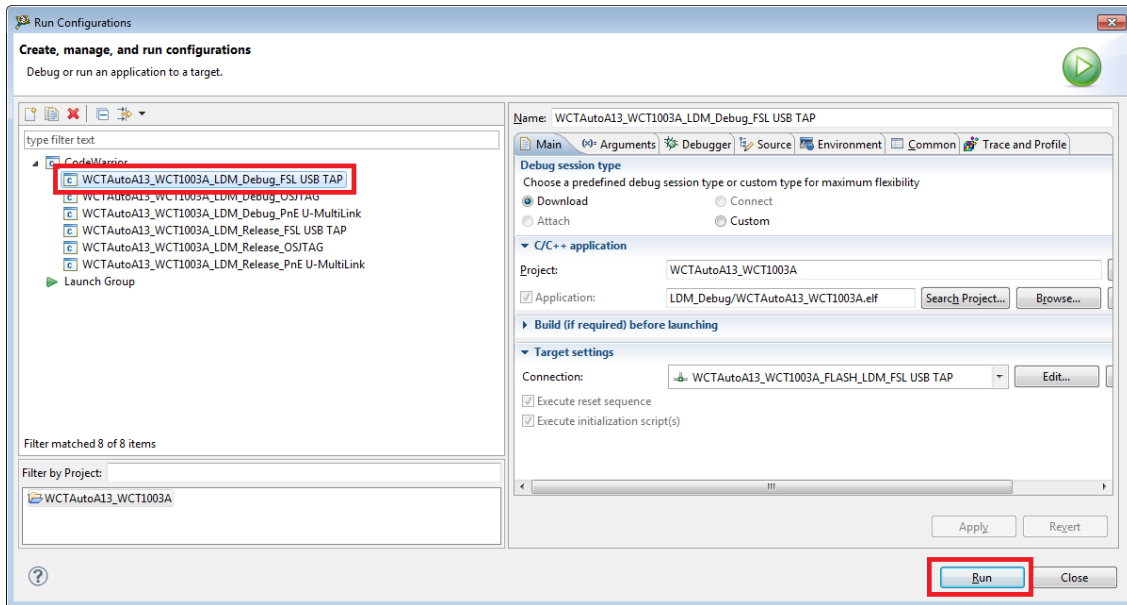


Figure 34 Selecting WCTAutoA13_WCT1003A_LDM_Debug_FSL USB TAP

6. Check the status at the bottom right corner, and wait until the programming finished.

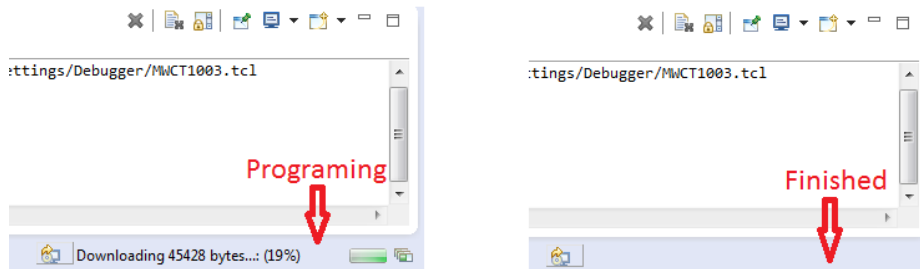


Figure 35 Programming finished

7.4 Program using the Bin file (.S)

1. Choose **Flash Programmer** → **Flash File to Target**.

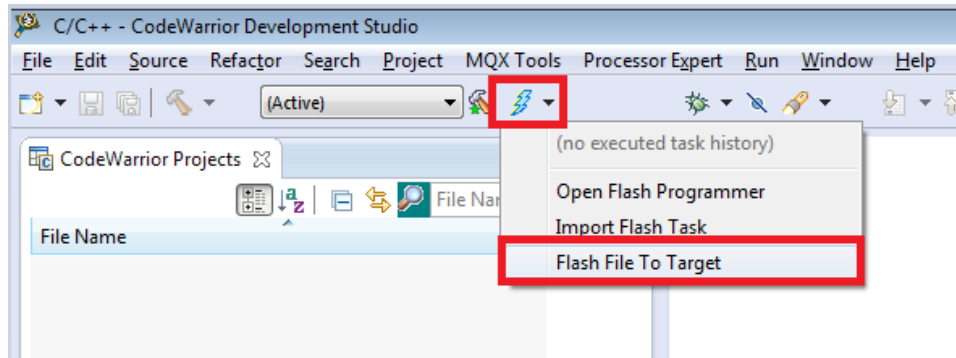


Figure 36 Choosing Flash File to Target

2. Click **New** to create a new connection.

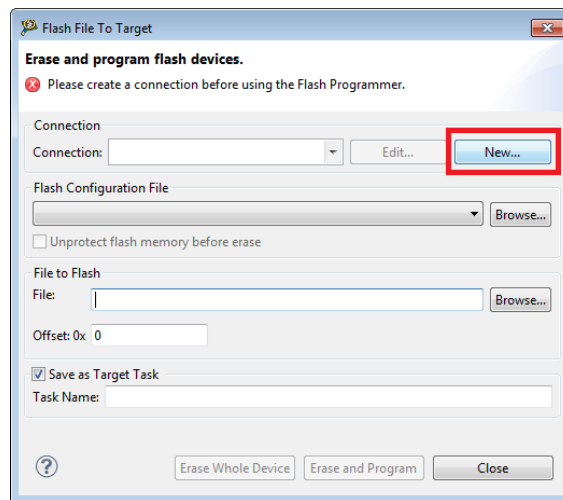


Figure 37 Creating a new connection

- In the **Name** text box, enter a connection name (any name is OK), and click **New** to create a target.

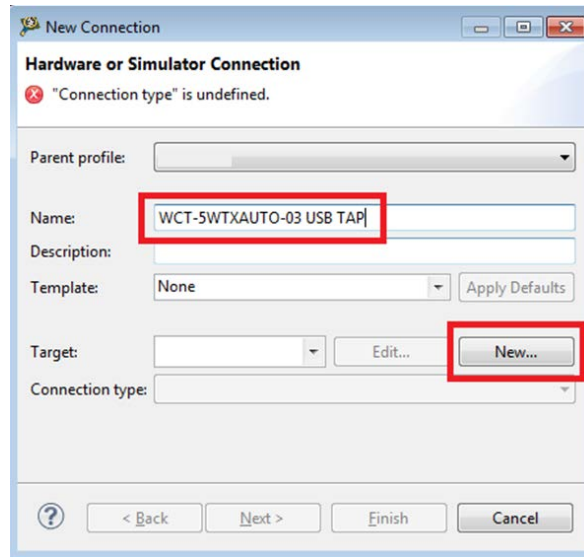


Figure 38 Entering a connection name

- In the **Name** text box, enter a target name (any name is OK but cannot be same with the connection name), and choose **dsc.MWCT10xx** -> **MWCT1003** from the **Target Type** drop-down list.

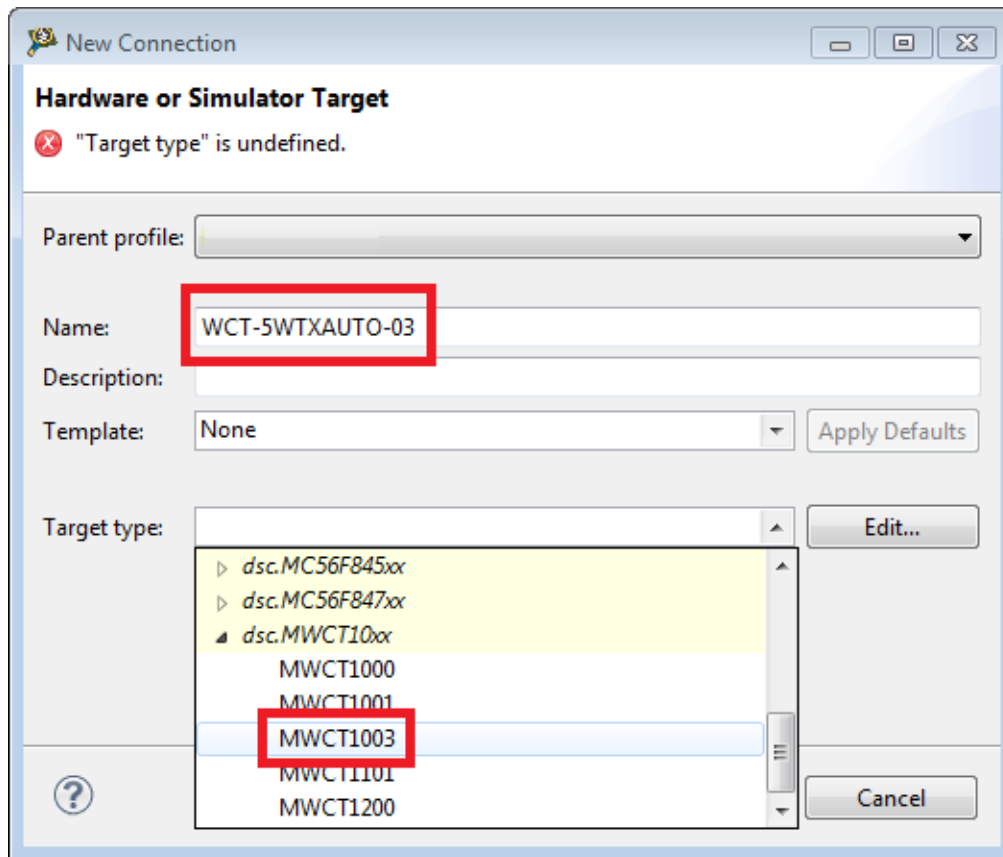


Figure 39 Choosing MWCT1003

5. Select **Execute reset** and **Initialize target**, set the initialization target file path to the CW installation folder, and then select **MWCT1003.tcl**.

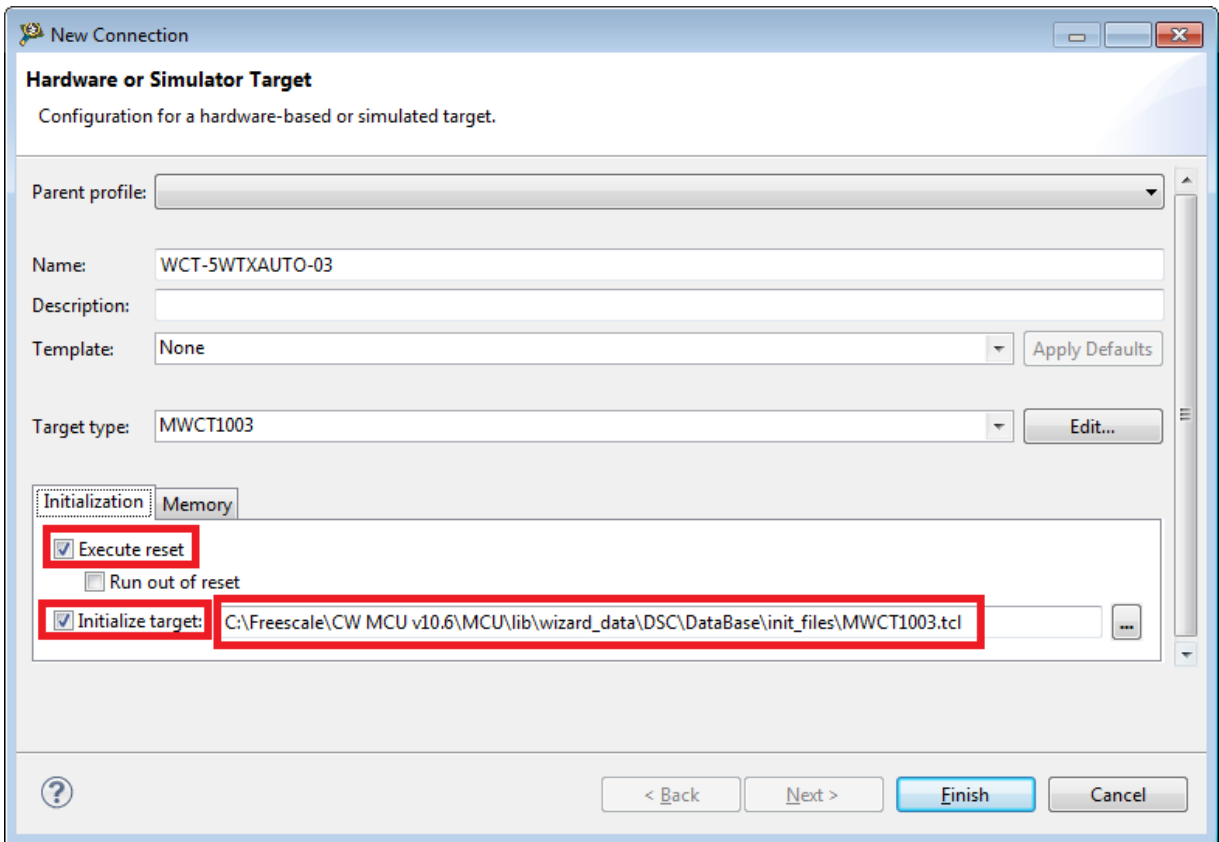


Figure 40 Executing reset and initializing target

- Click the **Memory** tab. Select **Memory configuration**, set the memory configuration file path to the CW installation folder, and then select **MWCT1003.mem**. Click **Finish**.

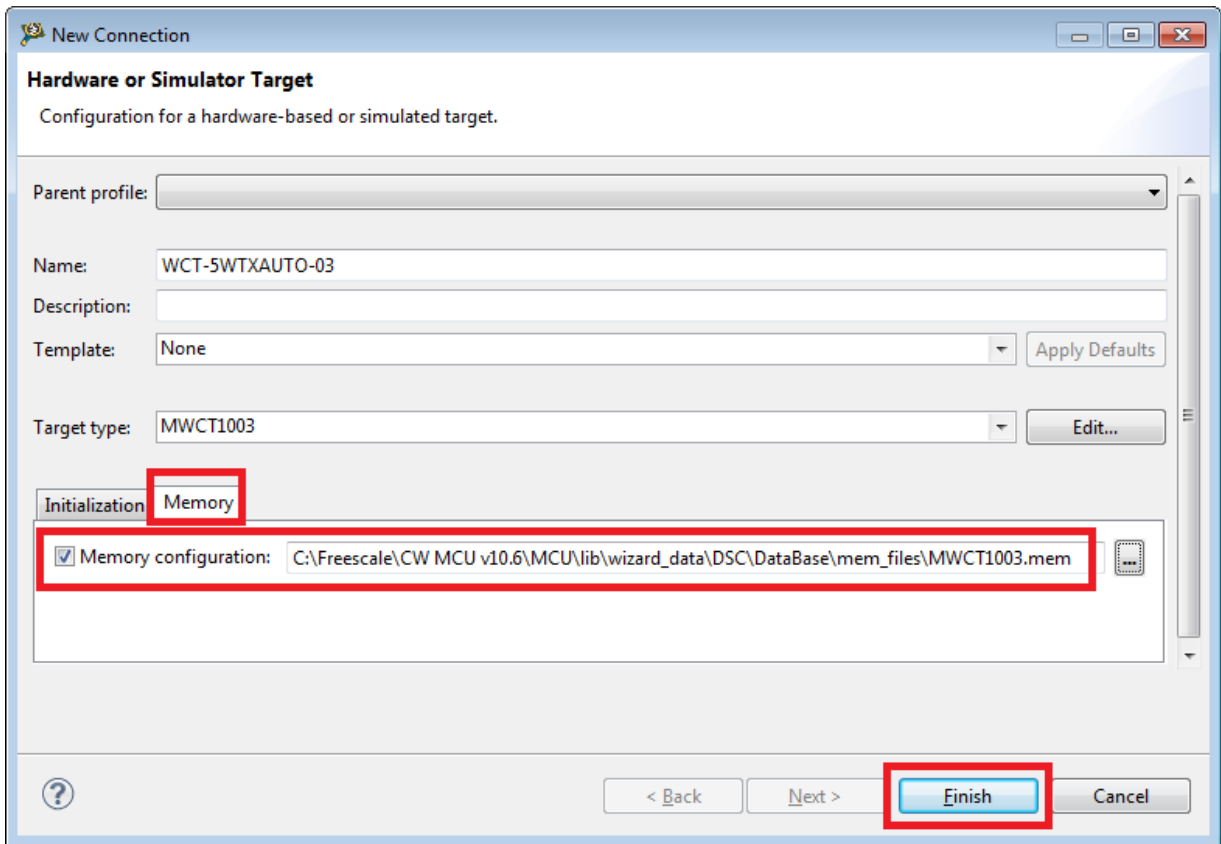


Figure 41 Memory configuration

7. Select **USB TAP** for the **Connection type**, and then click **Finish**.

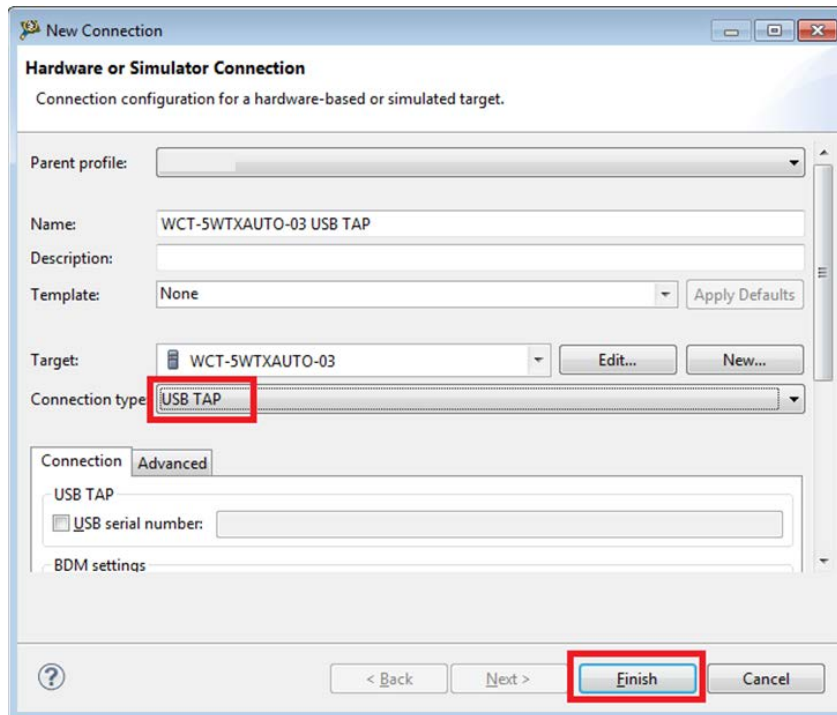


Figure 42 Setting the connection type

8. Set the .S file to be **File to Flash**. Select **Save the Target Task** for future programming. Power on A13 and click **Erase and Program**.

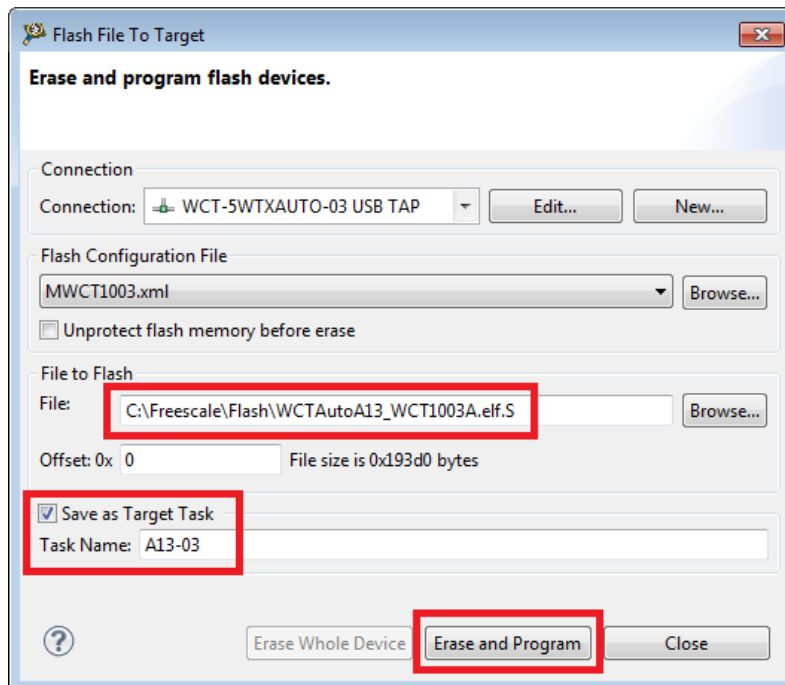


Figure 43 Erase and Program

9. Select the task path to save the task.

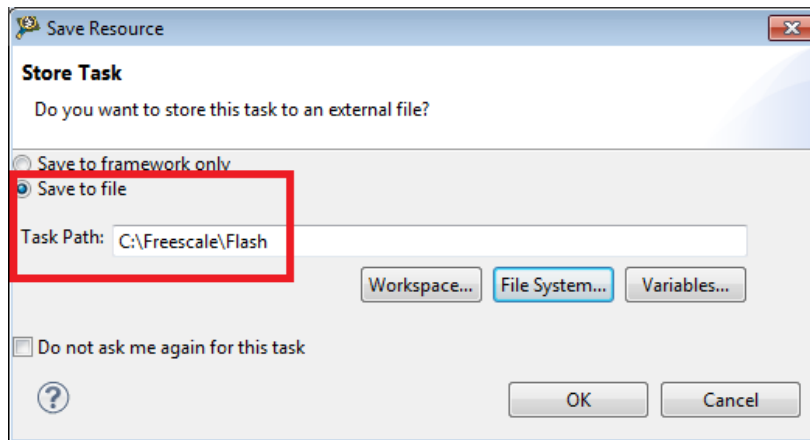


Figure 44 Selecting the task path

10. When program is finished, the **Console** window displays the following log.

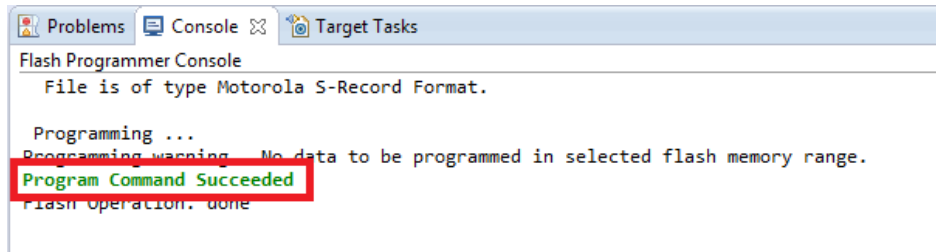


Figure 45 Programming finished log

11. For future programming, just select **A13-03** and wait until the programming is finished.

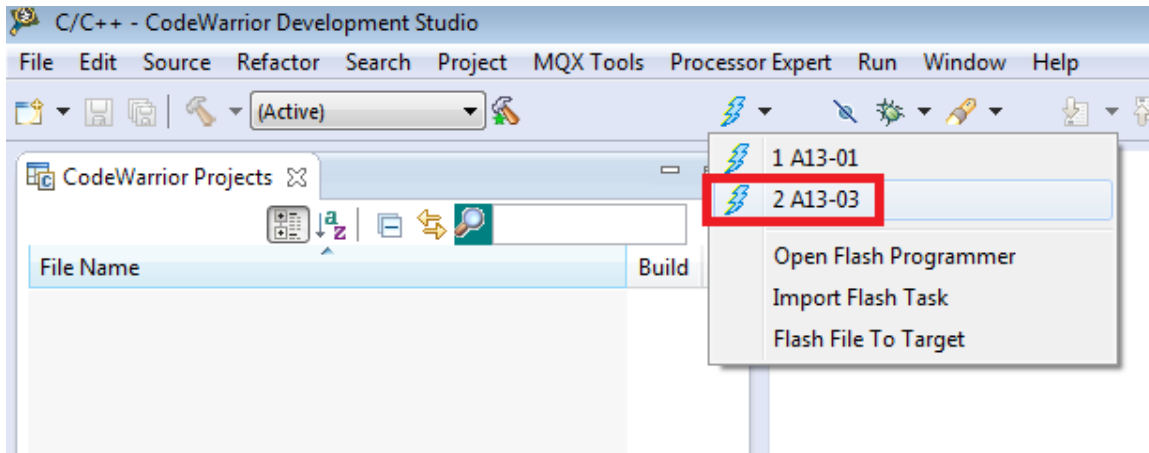


Figure 46 Future programming

7.5 Board calibration

Freescale provides the FreeMASTER GUI tool for calibration. For board calibration, see the *WCT1001A/WCT1003A Run-Time Debug User's Guide* (WCT100XARTDUG).

8. Implementation of Additional Functions

8.1 Touch function

8.1.1 Software setup

1. Enable the touch function

In application_cfg.h, configure macro TOUCH_USED to TRUE as follows:

```
#define TOUCH_USED TRUE
```

2. Recompile the project after reconfiguration.

8.1.2 Demonstration

When touch is enabled, install the accessory Touch Board to replace the Plastic Board (make sure the following three connectors are connected).

When the device is put on the charging surface, A13 sends the power signal out to ping for receiver (Rx) detection. Otherwise, A13 does not send any power signal out.

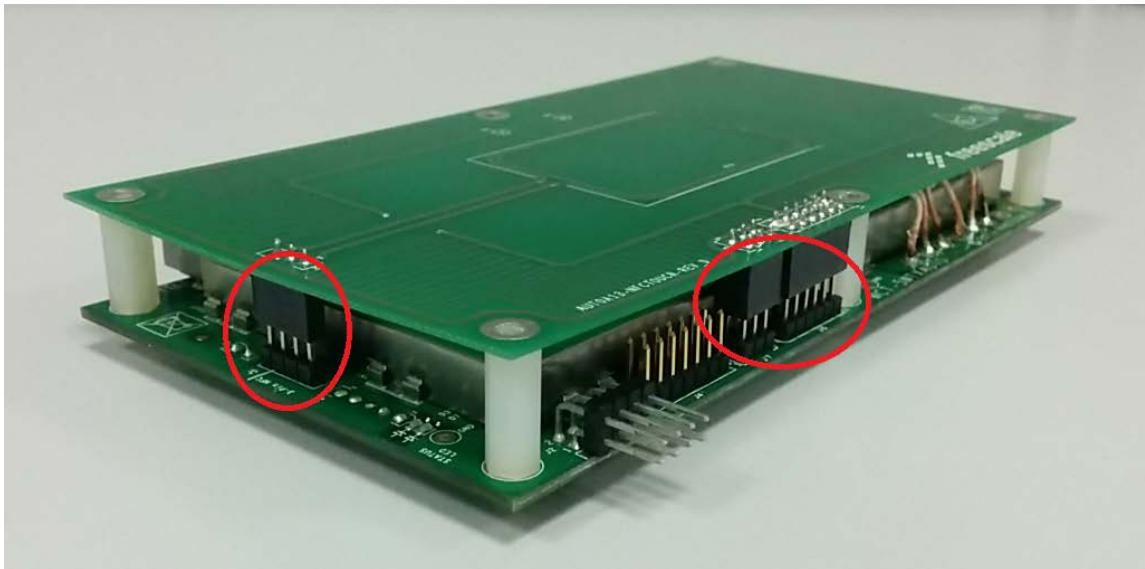


Figure 47 Three connectors connected

How to Reach Us:

Home Page:

www.freescale.com

Web Support:

www.freescale.com/support

Information in this document is provided solely to enable system and software implementers to use Freescale products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits based on the information in this document.

Freescale reserves the right to make changes without further notice to any products herein. Freescale makes no warranty, representation, or guarantee regarding the suitability of its products for any particular purpose, nor does Freescale assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. "Typical" parameters that may be provided in Freescale data sheets and/or specifications can and do vary in different applications, and actual performance may vary over time. All operating parameters, including "typicals," must be validated for each customer application by customer's technical experts. Freescale does not convey any license under its patent rights nor the rights of others. Freescale sells products pursuant to standard terms and conditions of sale, which can be found at the following address: freescale.com/SalesTermsandConditions.

Freescale and the Freescale logo are trademarks of Freescale Semiconductor, Inc., Reg. U.S. Pat. & Tm. Off. All other product or service names are the property of their respective owners.

© 2014 Freescale Semiconductor, Inc.