AVR-IoT WG User Guide

AVR-IoT WG Development Board User Guide

Preface

The AVR-IoT WG development board is a small and easily expandable demonstration and development platform for IoT solutions, based on the AVR® microcontroller architecture using Wi-Fi® technology. It was designed to demonstrate that the design of a typical IoT application can be simplified by partitioning the problem into three blocks:

- Smart - represented by the ATmega4808 microcontroller
- Secure - represented by the ATECC608A secure element
- Connected - represented by the WINC1510 Wi-Fi controller module

The AVR-IoT WG development board features a USB interface chip Nano Embedded Debugger (nEDBG) that provides access to a serial port interface (serial to USB bridge), a mass storage interface for easy 'drag and drop' programming, configuration and full access to the AVR microcontroller UPDI interface for programming and debugging directly from Microchip MPLAB® X IDE and the Atmel® Studio 7.0 IDE. The AVR-IoT WG development board comes preprogrammed and configured for demonstrating connectivity to the Google Cloud IoT Core.

The AVR-IoT WG development board features two sensors:

- A light sensor
- A high-accuracy temperature sensor - MCP9808

Additionally, a mikroBUS™ connector is provided to expand the board capabilities with 450+ sensors and actuators offered by MikroElektronika (www.mikroe.com) via a growing portfolio of Click boards™.
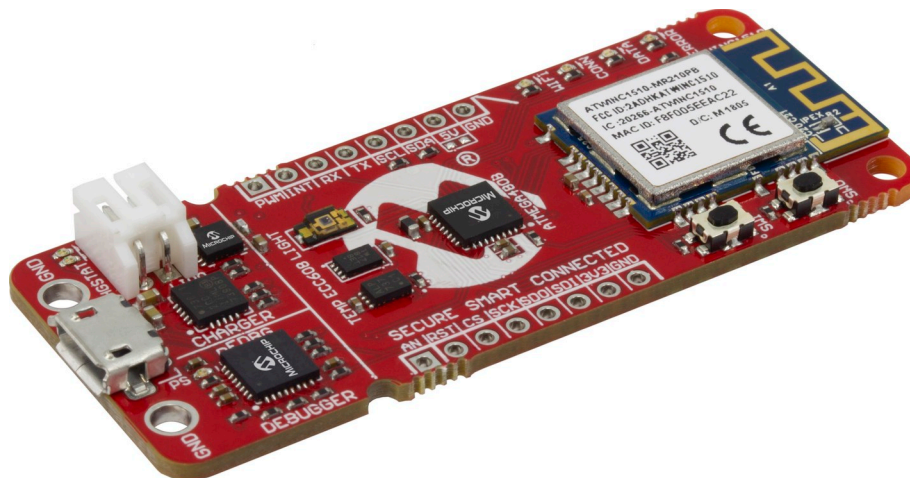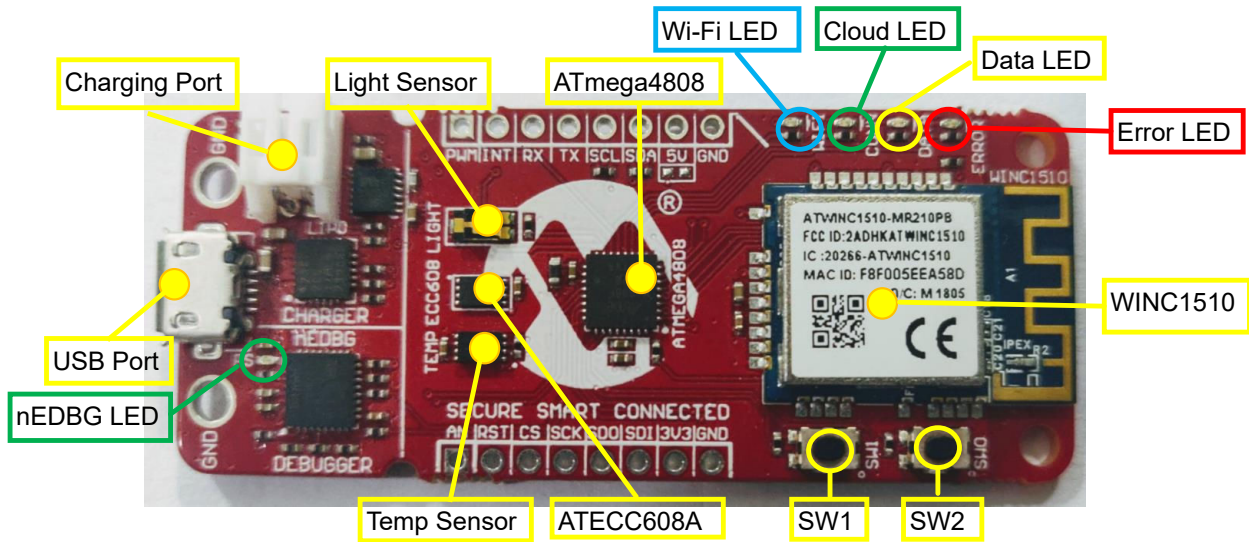
Table of Contents

1. Chapter 1: Overview

1.1 Board Layout

The AVR-IoT WG development board layout can be seen below.

1.2 LED Indicators

The development board features four LEDs that the demo code uses to provide diagnostic information as represented in the table below.

Table 1-1. LED Indicators

| LED Color	Label	System Element Monitored	Details
Blue	WIFI	Wi-Fi® Network Connection	Indicates a successful connection to the local Wi-Fi® network.
Green	CONN	Google Cloud Connection	Indicates a successful connection to the Google Cloud servers.
Yellow	DATA	Data Publication to Servers	Indicates that a packet of sensor data has been successfully published to the Google Cloud MQTT servers.
Red	ERROR	Error Status	Indicates that an error happened after the last step.

2. Chapter 2: Getting Started

2.1 Connecting the board to the PC

First, connect the AVR-IoT WG development board to the computer using a standard micro-USB cable. Once plugged in, the LED array at the top right-hand corner of the board should flash in the following order twice: Blue->Green->Yellow->Red. If the board is not connected to Wi-Fi, the Red LED will light up. The board should also appear as a Removable Storage Device on the host PC, as shown in the figure below. Double click the **CURIOSITY** drive to open it and get started.

Note: All procedures are the same for Windows®, Mac OS®, and Linux® environments.

Figure 2-1. Curiosity Board as Removable Storage
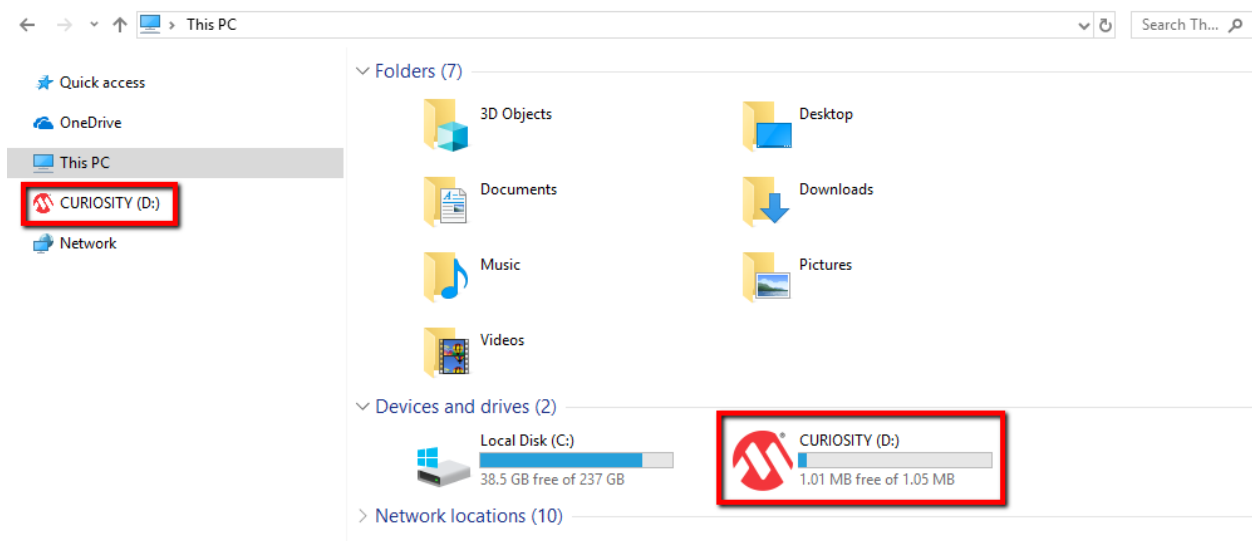
2.1.1 The AVR-IoT WG Experience

The CURIOSITY drive should contain the following five files:

- CLICK-ME.HTM - redirects the user to the AVR-IoT web demo application
- KIT-INFO.HTM- redirects the user to a site containing information and resources about the board
- KIT-INFO.TXT - a text file with details about nEDBG firmware and the board's serial number
- PUBKEY.TXT - a text file containing the public key used for data encryption
- STATUS.TXT - a text file containing the status condition of the board.

Double click on the **CLICK-ME.HTM** file to go to the dedicated webpage to access the Google Cloud sandbox account. Figure 2-3 shows an image of the AVR-IoT WG webpage. On this page, the user can quickly see sensor data, reconfigure the Wi-Fi credentials of the board, download additional example codes and customize the application. The status markers at the middle of the page, as shown in Figure 2-2, indicate the progress of the system setup. These markers will light up once each stage is completed successfully. The leftmost marker indicates if the board is connected to the host PC. Next to this, the Wi-Fi marker lights up once the board is connected to a Wi-Fi network, turning on the Blue LED of the board. To the right of the Wi-Fi marker, the Google Cloud MQTT marker can be found, indicating the status of the connection to the Google Cloud server; this corresponds to the Green LED on the board. Finally, the lighting up of the rightmost marker signifies that data is streaming from the board to the server, by blinking

the Yellow LED on the board. If there is no data streaming, the lower right-hand side of the page will be showing the video demonstration of the setup instructions.
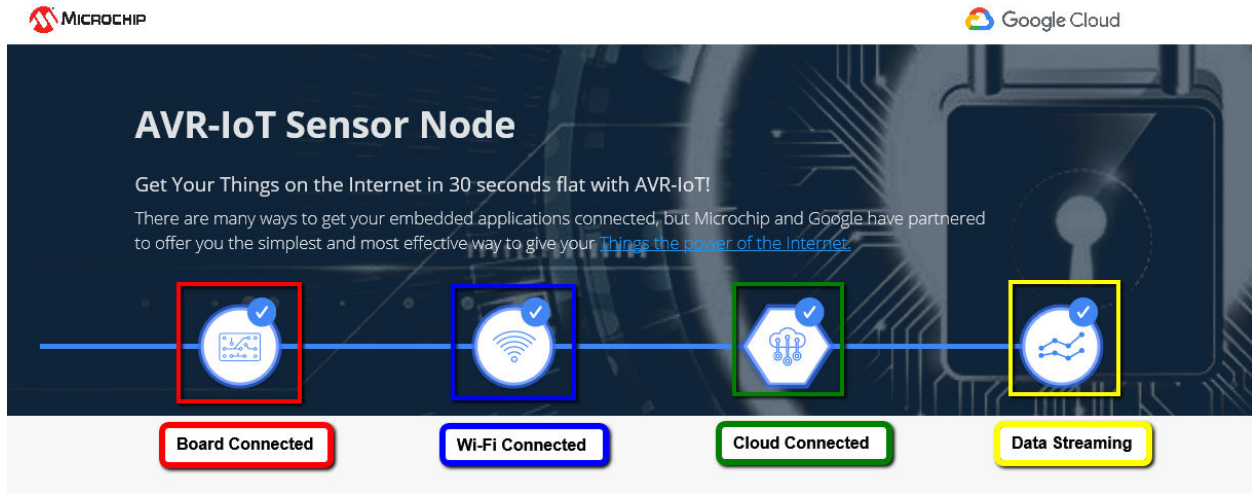
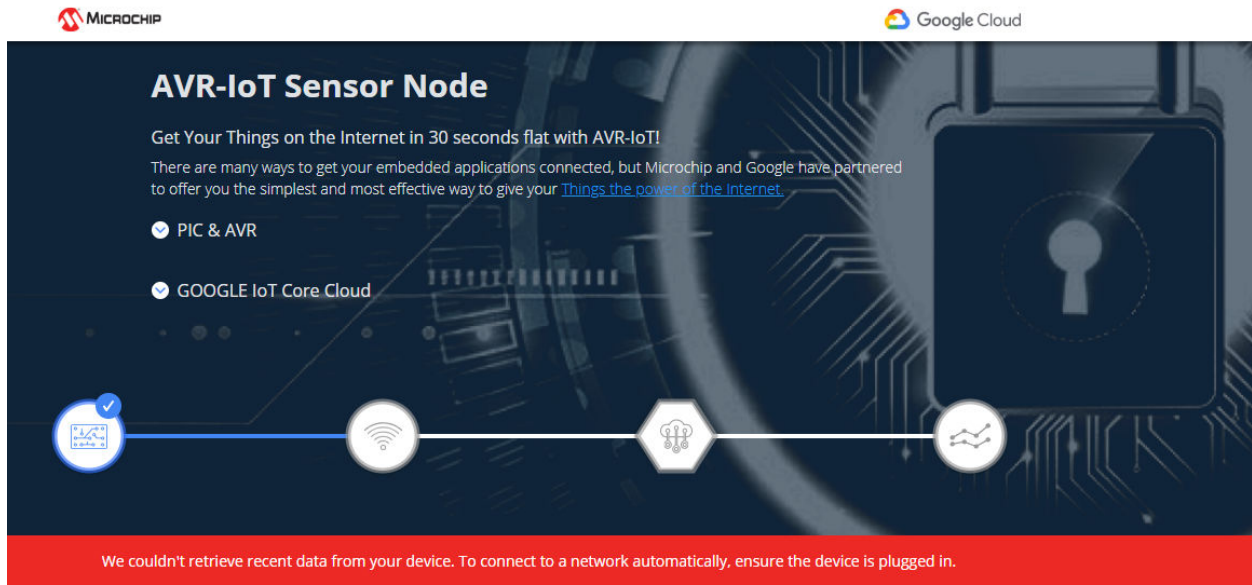**Figure 2-2. Webpage Status Indicators**

Figure 2-3. AVR-IoT WG Webpage (No Wi-Fi Connection)

2.1.2   Connecting to the Wi-Fi Network

When the connection has not been established, the lower left-hand corner of the Microsite will show a wireless network connection window where the user can choose to connect to an Open (no password required) network or enter the credentials for a password protected Wi-Fi network. For this live demonstration, the user needs to fill in the text fields shown in Figure 2-4. These are the details for the Wi-Fi network setup used during the class. For other means of connection to the internet like mobile hotspots, the user may fill these fields with the SSID and password of their own Wi-Fi network .

Note: The Wi-Fi network SSID and password are limited to 19 characters. Avoid using quotation marks, names or phrases that begin or end in spaces.

Figure 2-4. Entering Wi-Fi Credentials in Microsite

Once the required details are entered, click the **Download Configuration** button. This will download the WIFI.CFG (text) file on the host PC. From the WIFI.CFG's download location, drag and drop the file to the CURIOSITY drive to update the Wi-Fi credentials of the board. The Blue LED will light up to show a successful connection. Otherwise, refer to Chapter 3 to troubleshoot any board issues.

Note: Any information entered in the SSID and password fields is not transmitted over the web or to the Microchip or Google servers. Instead, the information is used locally (within the browser) to generate the WIFI.CFG file.

2.1.3 Security Provisions

The secure element (ATECC608A), present on the AVR-IoT WG boards, comes pre-registered within the MCHP AVR-IoT (sandbox) account on Google Cloud. Each secure element provides an 18-digit hexadecimal Unique Identification Number (UID) and a public or private key pair, pre-generated using Elliptic Curve cryptography. The UID can be seen on the URL of the webpage application or via the serial command line interface (discussed later on in the document). The private key is never revealed by the

secure element but the public key can be viewed in the PUBKEY.TXT file or through the serial command line interface.

Figure 2-5. Device UID
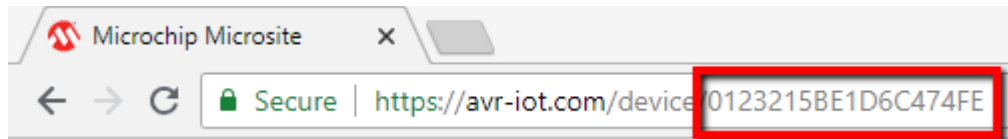

2.1.4 Visualizing Cloud Data in Real Time

Out of the box, all AVR-IoT development boards are pre-registered to Microchip's Google Cloud sandbox account. This account is set up for demonstration purposes only. All data gathered by the sensors of the AVR-IoT development boards are published on the Microchip sandbox account and can be identified by the following details:

Project ID	avr-iot
Region	us-central1

There is no permanent storage or collection of the data published by the boards connected through the Microchip sandbox account. The full storage of the Google Cloud features will be available to the user after the board is removed from the demo environment and migrated to a private account.

Once the board is connected to the Wi-Fi and to the Cloud, the avr-iot.com webpage will show a real-time graph of the data gathered from the on-board light and temperature sensors. Data are transferred and transformed from the sensor to the cloud through a JSON object: an ASCII string formatted as follows: { 'Light' : XXX, 'Temp': YYY }, where XXX and YYY are numerical values expressed in decimal notation.

Figure 2-6. Real-Time Data on the Microsite
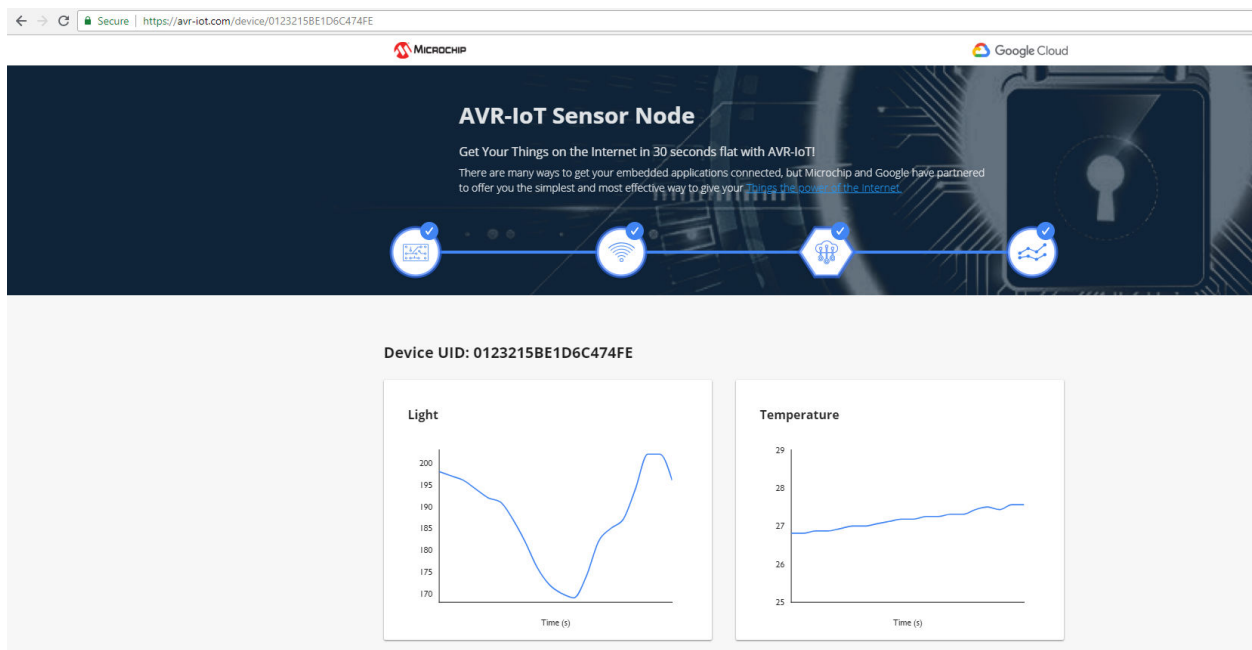
2.1.5 The USB Interface

While the AVR-IoT WG development board comes out of the box fully programmed and provisioned, the user can still access the firmware through the USB interface. There are three methods to do this: through

drag and drop, the serial command line interface, or through the on-board programmer/debugger using Atmel Studio 7.0.

I. USB Mass Storage ('Drag and Drop')

One way to program the device is to just drag and drop a `.hex` file into the CURIOSITY drive. The AVR C compiler tool chain generates a `.hex` file for each project it builds. This `.hex` file contains the code of the project. The AVR-IoT WG board facilitates putting code into the board by having this drag and drop feature. This feature does not require any USB driver to be installed and works in all major OS environments. Alternative application example `.hex` files for the board firmware will be available for download from the downloads section at the bottom of the avr-iot.com webpage.

II. Serial Command Line Interface

The AVR-IoT WG development board can also be accessed through a serial command line interface. This interface can be used to provide diagnostic information. To access this interface, use any preferred serial terminal application (i.e. Teraterm, Coolterm, PuTTy) and open the serial port labeled Curiosity Virtual COM port, with the following settings:

Baud Rate	9600
Data	8-bit
Parity Bit	None
Stop Bit	1 bit
Flow Control	None
Additional Settings	Local Echo: On
Transmit to the Microcontroller	CR+LF (Carriage Return + Line Feed)

Note: For users of the Windows environment, the USB serial interface requires the installation of an USB serial port driver.

The user can control the board by typing the command keywords, listed in Table 2-1.

Table 2-1. Serial Command Line Commands

Command	Arguments	Description
reset	-	Reset the settings on the device
device	-	Print the unique device ID of the board
key	-	Print the public key of the board
reconnect	-	Re-establish connection to the Cloud
version	-	Print the firmware version of the serial port user interface
cli_version	-	Print the command line interface firmware version of the serial port user interface

..........continued		
Command	**Arguments**	**Description**
wifi (see Figure 2-8 for example)	<Network SSID>, <Password>, <Security Option*>	Enter Wi-Fi®network authentication details
debug (see Figure 18/19 for sample of debug messages)	<Debug Options**>	Print debug messages to see status of board operation

*- Type in one of these three numbers to choose among the following security options:

1. Open - Password and Security option parameters are not required. See Figure 2-8 for an authentication example for open networks.
2. WPA/WPA2 - Security Option Parameter not required. See Figure 2-9 for an authentication example for the WPA/WPA2 networks.
3. WEP

**- Type in a number from 0 to 4; for the number of debug messages with 0 - the result is printing no messages and with 4 for printing all the messages.

Figure 2-7. Serial Command Line Interface

Figure 2-8. Wi-Fi Authentication for Open Networks
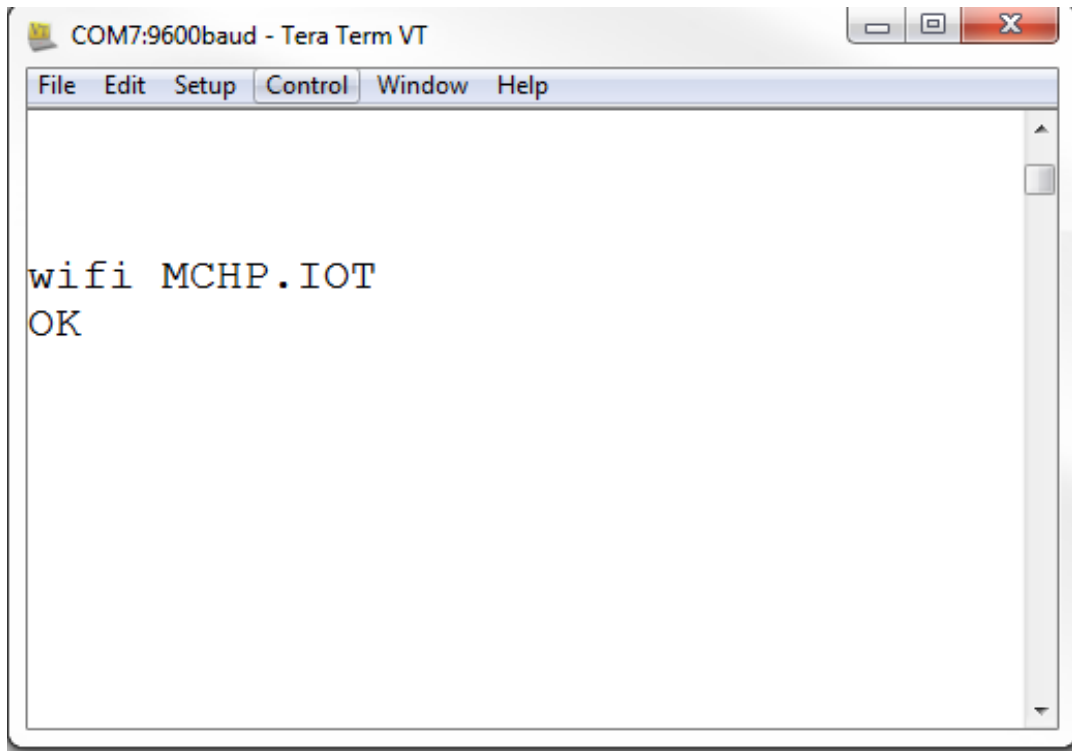
Figure 2-9. Wi-Fi Authentication for Password-Protected Networks
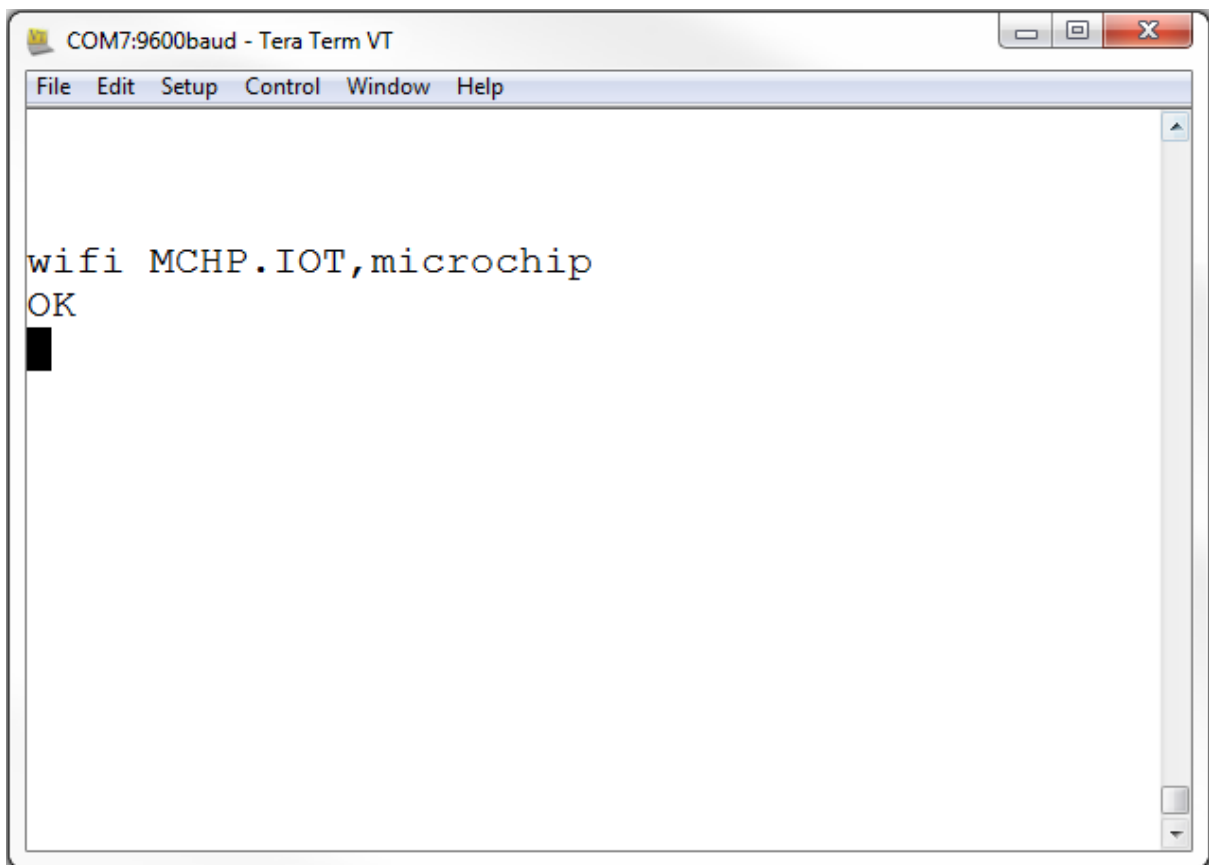
III. USB Programmer/Debugger interface

For users familiar with the Atmel Studio interface, the AVR microcontroller can also be programmed and debugged directly via the Atmel Studio 7.0 IDE. The AVR-IoT development board is automatically detected by the Atmel Studio IDE, enabling full programming and debugging through the on-board nEDBG interface.

2.2 AVR-IoT Development on START

Atmel START, a quick development tool, can be used to select and customize additional code examples including single-click support for 100+ Click sensor boards (out of the 450 models available so far). The codes can be downloaded by clicking **Browse Examples** on the Atmel START page, as shown in Figure 2-9.

I. Generate the AVR-IoT Development Board Demo

To generate the microcontroller code used on the AVR-IoT development board, select **Browse Examples** from the Atmel START home page and follow these simple steps:

1. Search and select the **AVR-IoT WG Sensor Node**.
2. To download the demo code as it is, click **Download Selected Example**. To make modifications to the code, click **Open Selected Examples**.
3. To make changes to the configuration, such as the Google Cloud project details, scroll down the page to the AVR-IoT WG Sensor Node panel, as shown in Figure 2-13. The user can find their Google Cloud Project details like Project ID at: https://console.cloud.google.com/cloud-resource-manager.
4. Once these changes are made, the following options are available: preview the code, save the configuration for later use, or export the project to a selected development environment. To select one of the options, click the corresponding tab on the top of the page shown in Figure 2-14.
5. If the project is to be exported, click on the **EXPORT PROJECT** tab and select which IDE or tool will be used. Then click the **DOWNLOAD PACK** button. Once downloaded, follow the "Getting Started With Atmel Studio 7" Guide to import the Start project to Atmel Studio (see Figure 2-15).
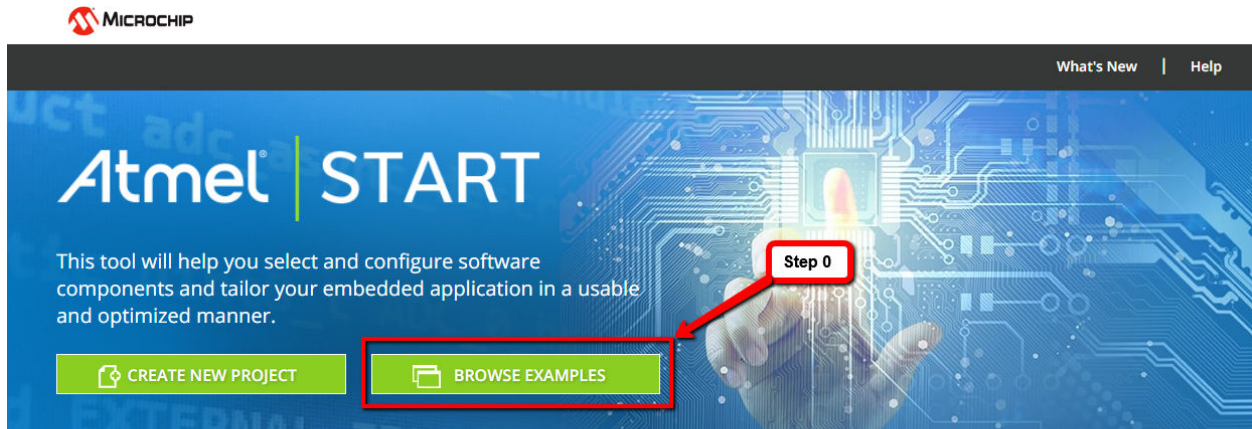
II. Generate AVR-IoT WG Sensor Node with supported mikroElektronika Click Boards

Atmel START can also generate example codes for two supported MikroElektronika Click Boards: Weather Click and Air Quality Click. To generate code for either of these, select the corresponding project in the examples list in Atmel START and follow steps 2 to 4 to regenerate the AVR-IoT WG development demo code. Additional code examples will be posted in future releases of the Atmel START tool.

III. Exporting AVR-IoT WG START Project to Atmel Studio

After generating an AVR-IoT WG project in Atmel START, export it to Atmel Studio to be compiled, linked and eventually programmed into the AVR microcontroller. For instructions on how to import Atmel START projects into Atmel Studio and program them onto the board, refer to the Atmel START User Guide.

Figure 2-10. ATMEL START Homepage

Figure 2-11. ATMEL START Browse Examples Page

Figure 2-12. AVR-IoT WG Firmware Map

Figure 2-13. AVR-IoT WG Configuration Section

Figure 2-14. User Options Tabs

Figure 2-15. Exporting a Picture
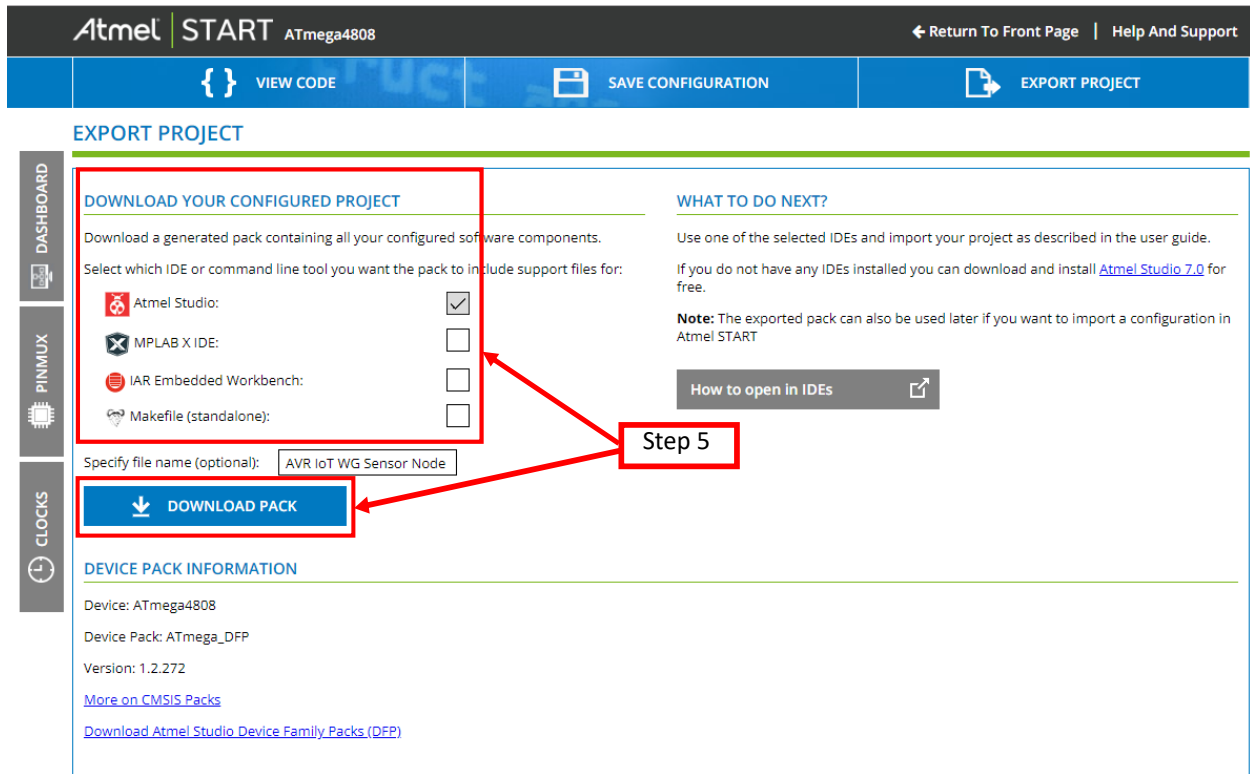
2.3 AVR-IoT Development on MCC (MPLAB® Code Configurator)

The source code of the AVR-IoT WG board is also available as an example code in MPLAB® Code Configurator (MCC). To generate the codes, the following should be installed on the user's machine:

Table 2-2. Software for MCC Code

Software	Download link
MPLAB® X IDE v5.10 or later	https://www.microchip.com/mplab/mplab-x-ide
AVR GCC Compiler v5.4.0 or later or XC8 v2.05 or later	• https://www.microchip.com/mplab/avr-support/avr-and-arm-toolchains-c-compilers • https://www.microchip.com/mplab/compilers
MCC Plugin v3.66 or later	https://www.microchip.com/mplab/mplab-code-configurator
avr8bit_v1.1.0 or later	Bundled with MCC v3.66
AVR_IoT v1.00 or later	Bundled with MCC v3.66

1. Generate the AVR-IoT Development Board Demo

Once the board is connected to the host machine, follow these steps to generate microcontroller code for it:

a. Creating a new MPLAB X project

1. Create a new Standalone project (see Figure 2-16) in MPLAB X 5.10 or later using the ATmega 4808 as device (see Figure 2-17); the nEDBG as programming tool (see Figure 2-18); and the AVR GCC Compiler as compiler (see Figure 2-19). Finally, name the MPLAB project and its location (seeFigure 2-20). The Start page will then appear (see Figure 2-21).

Figure 2-16. Create New Project
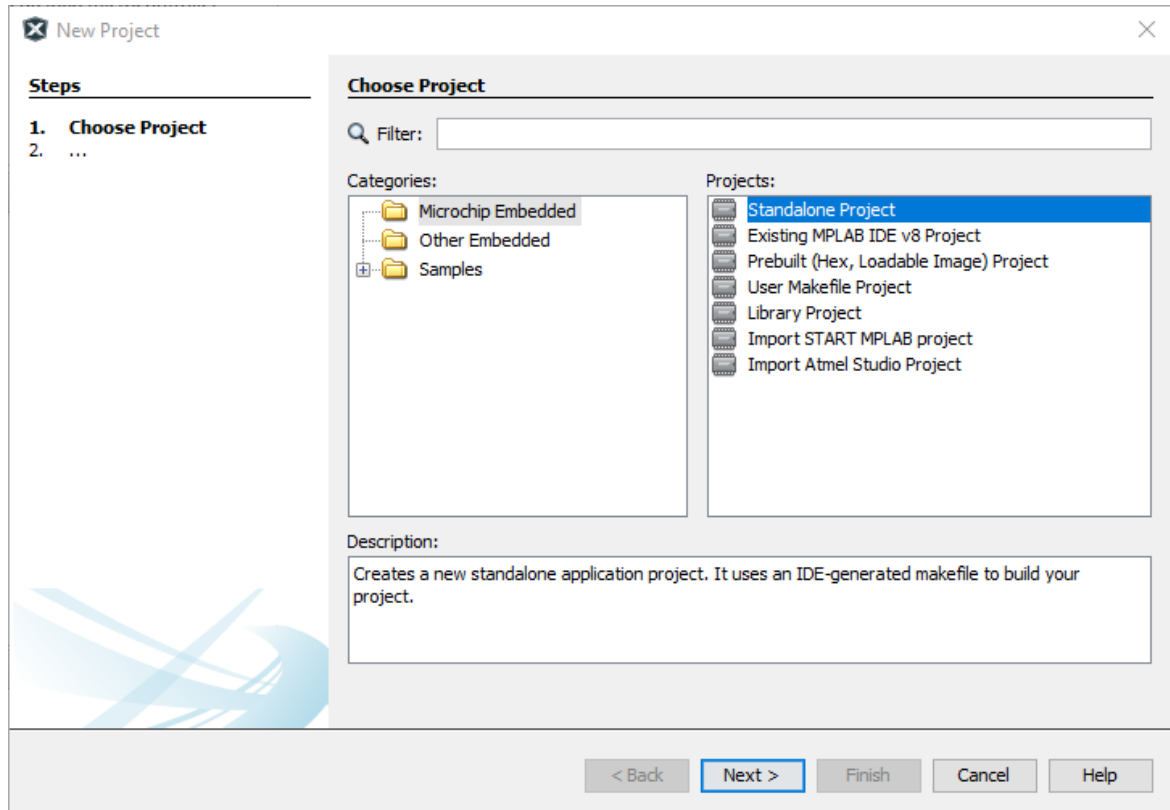
Figure 2-17. ATmega4808
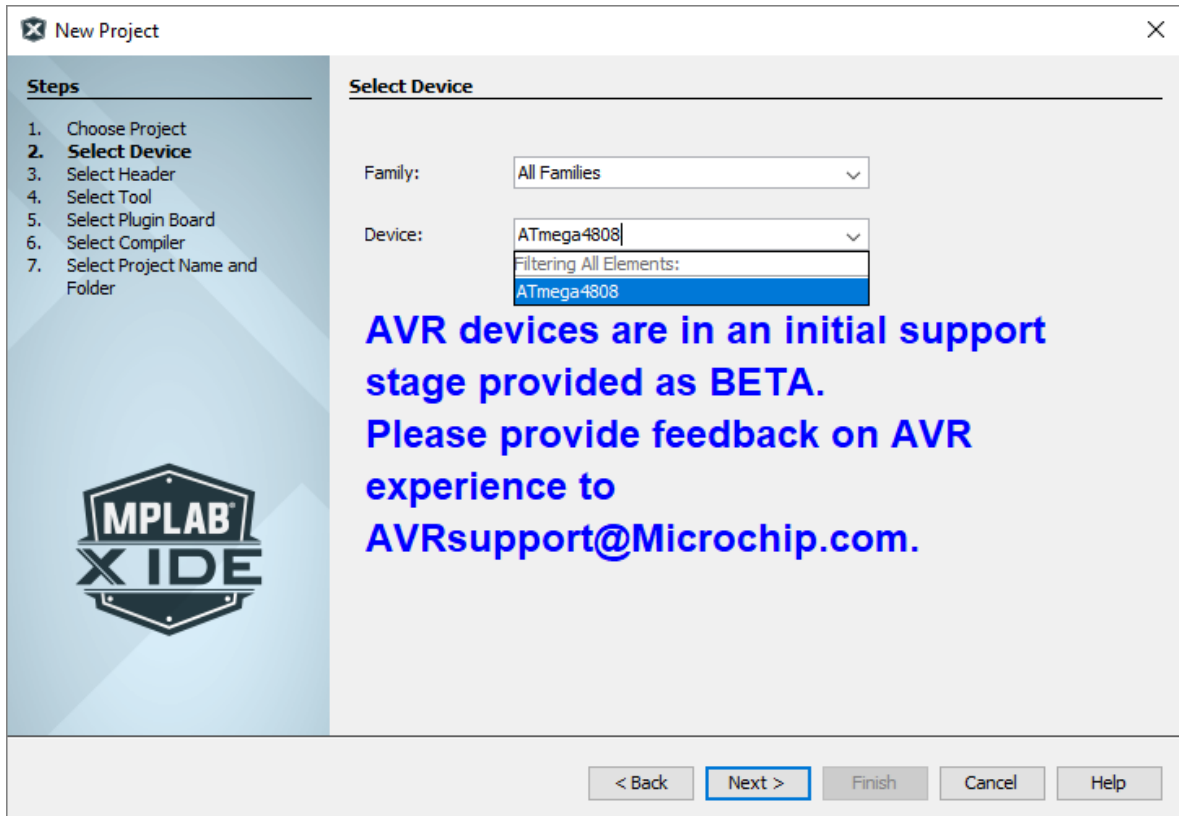
Figure 2-18. nEDBG
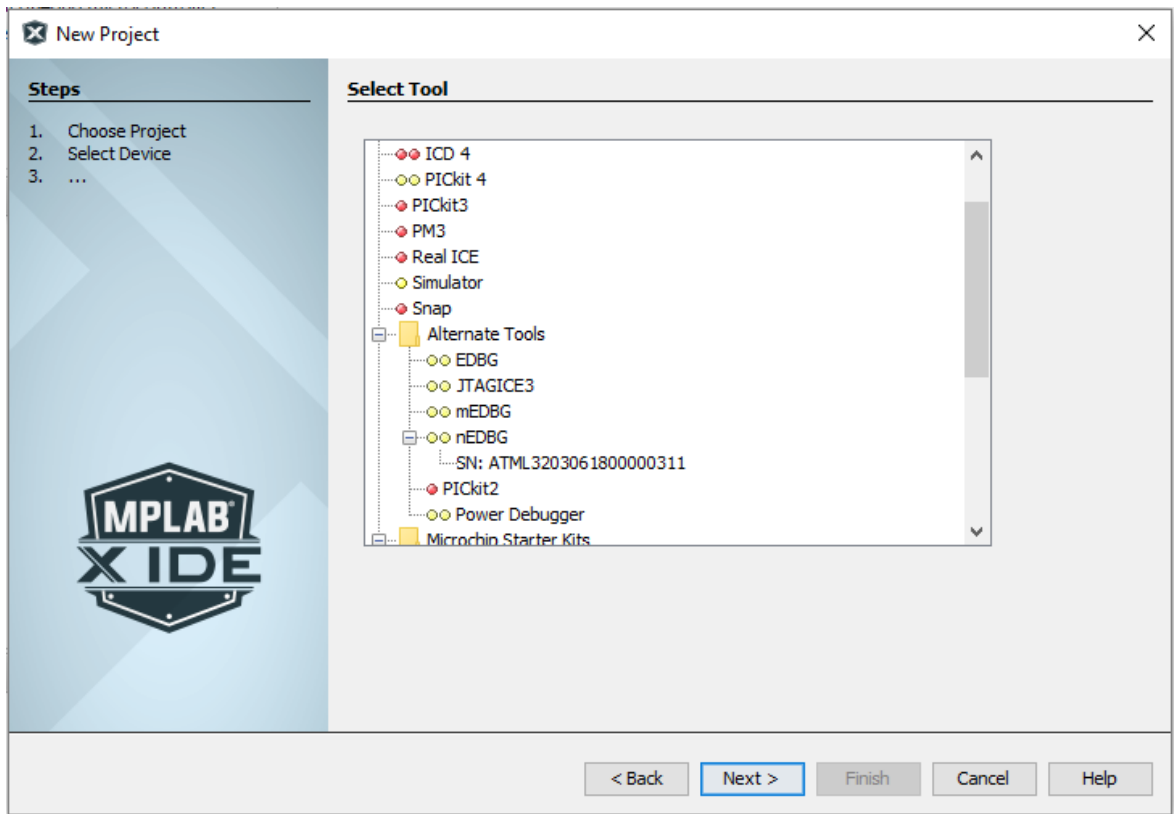
Figure 2-19. The AVR GCC Compiler
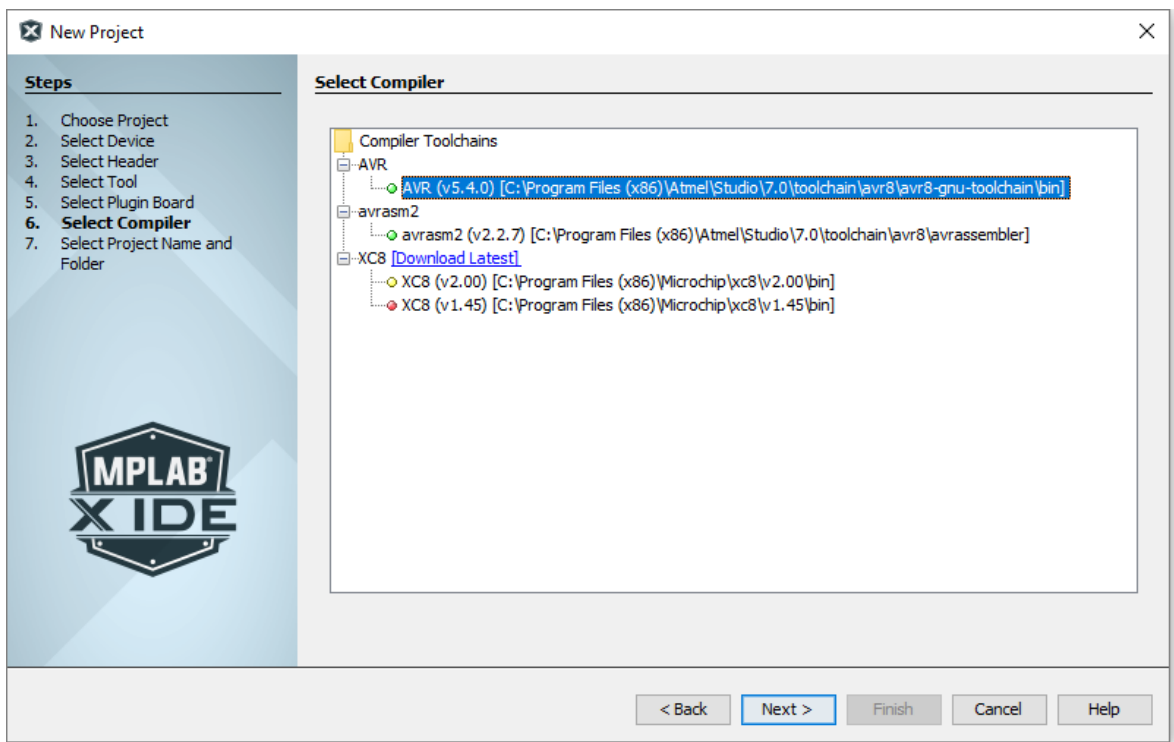
Figure 2-20. Project Name and Location
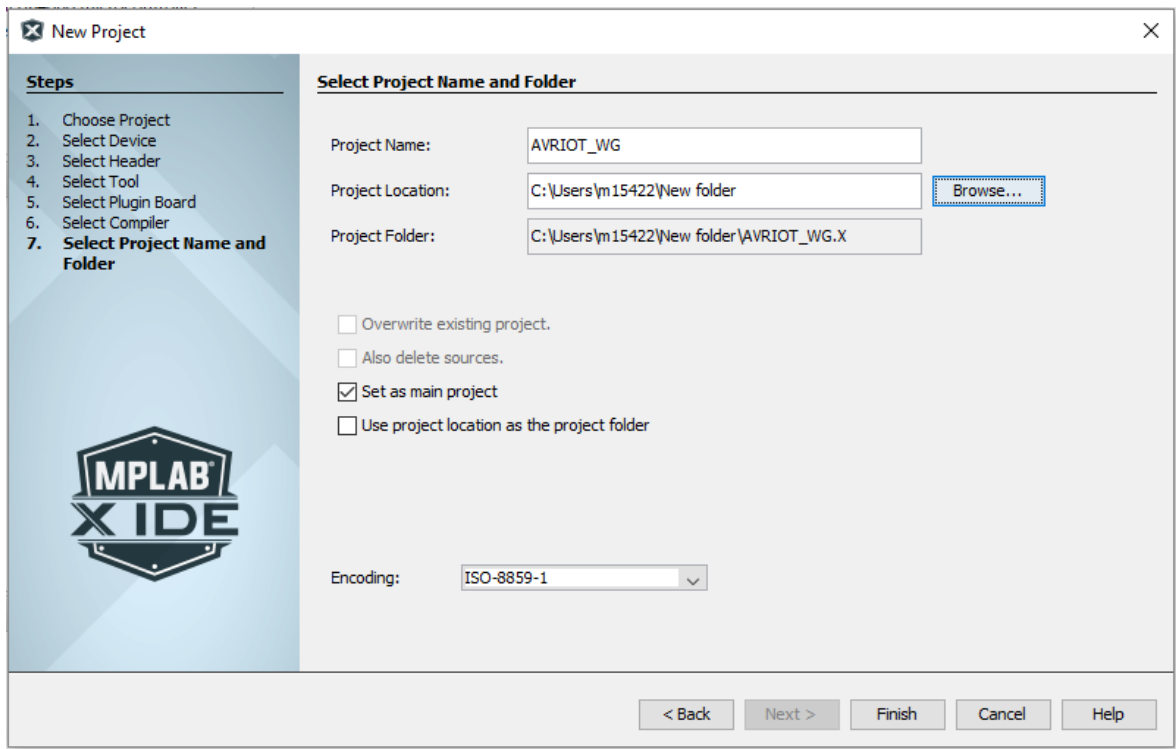
Figure 2-21. Start Page
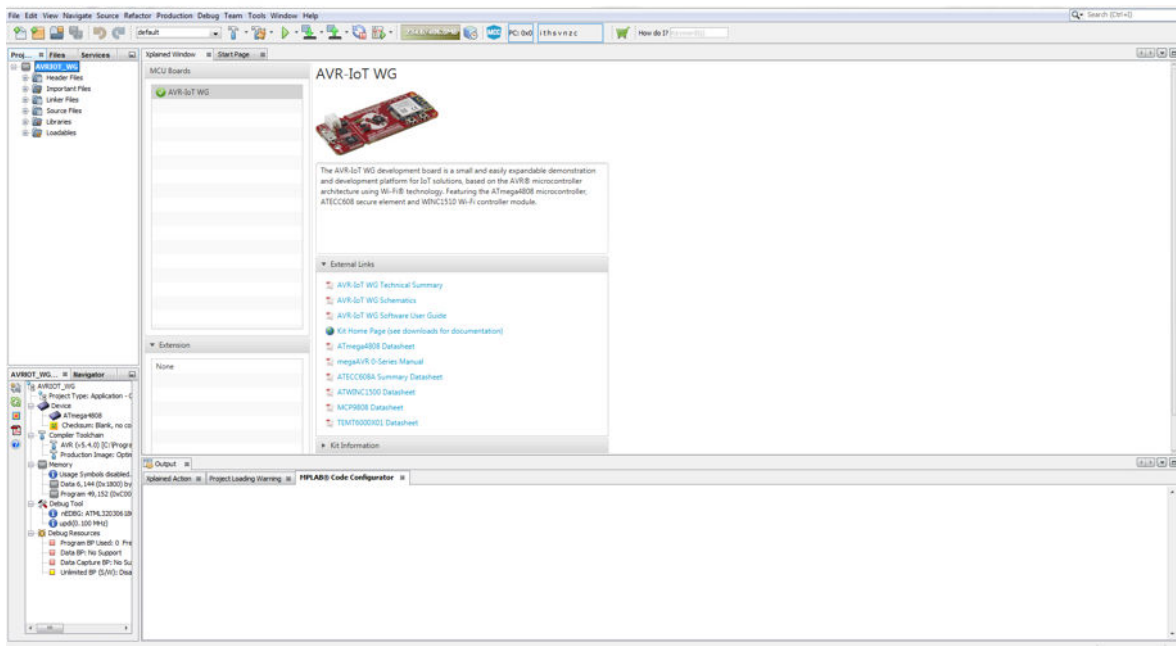
2. On the MPLAB X toolbar, look for and click the MCC Icon ( ) or click *Tools>Embedded>MPLAB X Code Configurator v3 Open/Close*(see Figure 2-22).

Figure 2-22. MCC location path
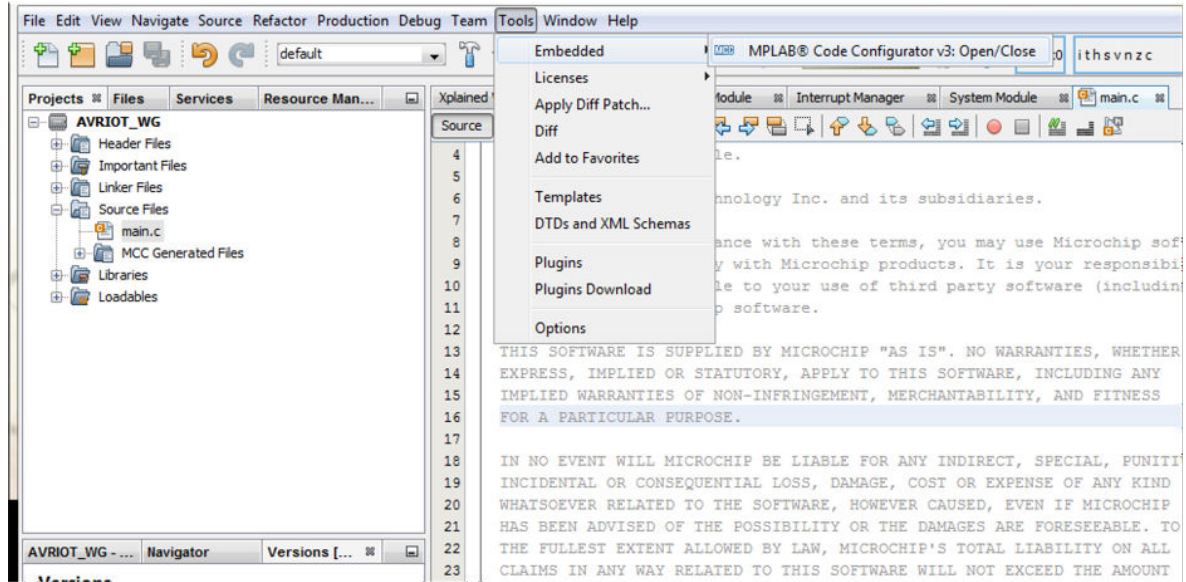
3. Under Device Resources, scroll down to the 'Internet of Things' header. Under Examples, double-click on '**AVR-IoT WG Sensor Node**'(see Figure 2-23).

Figure 2-23. AVR-IoT WG Sensor Node
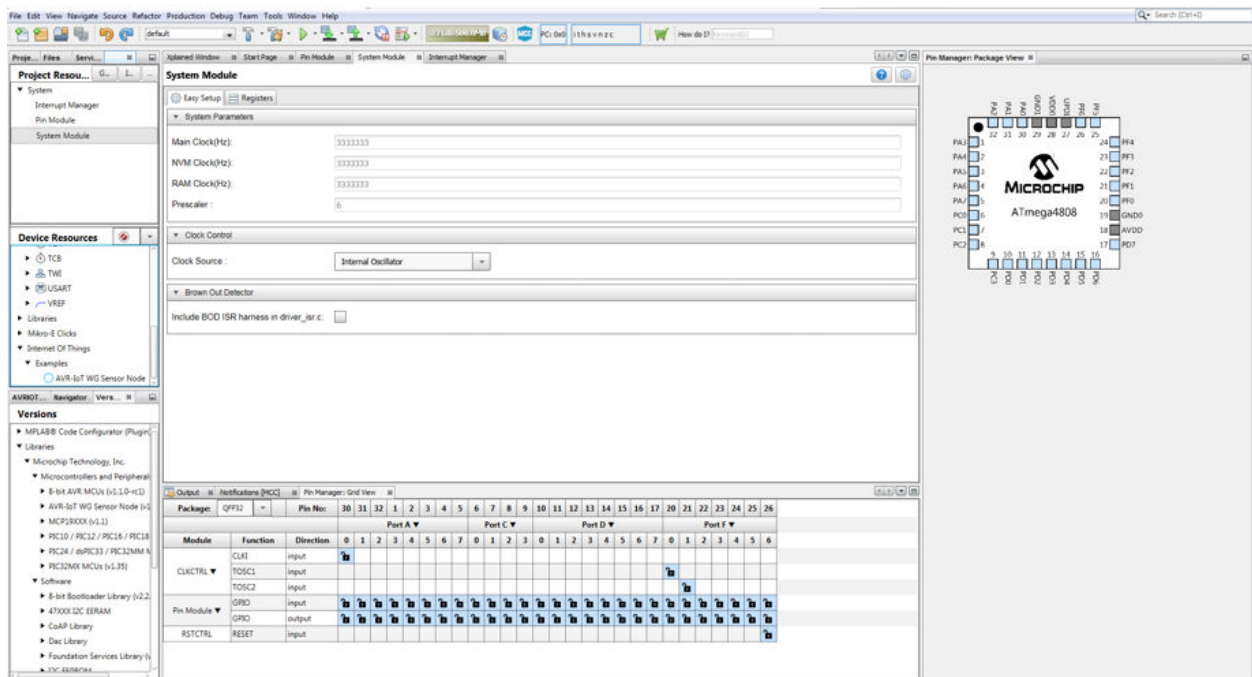
b. Configuring the settings of the project

The AVR-IoT WG Sensor Node module makes use of multiple libraries and peripherals. To configure the libraries, double-click on each library in the Device Resources window (see Figure 2-24) to view their setup windows.

Figure 2-24. AVR-IoT WG Sensor Node Libraries
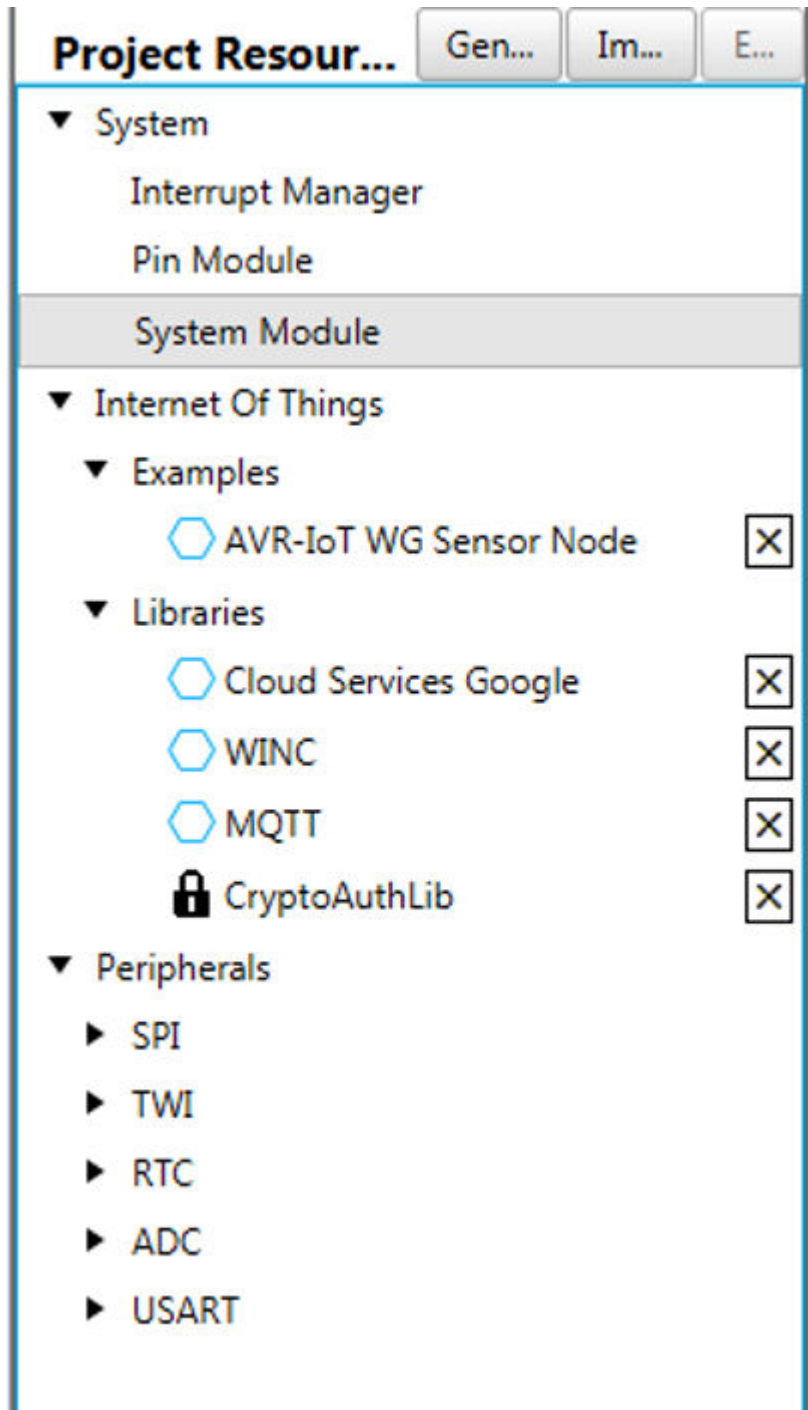
c. CryptoAuthLib

The Crypto Authentication Library (CryptoAuthLib) is not available for user modification but it shows the macros that need to be enabled for the Crypto Authentication functionalities of the AVR-IoT WG board to work. It also indicates the communication settings between the ECC608 chip and the ATmega4808 microcontroller on board (see Figure 2-25).

Figure 2-25. Crypto Authentication Library (CryptoAuthLib)
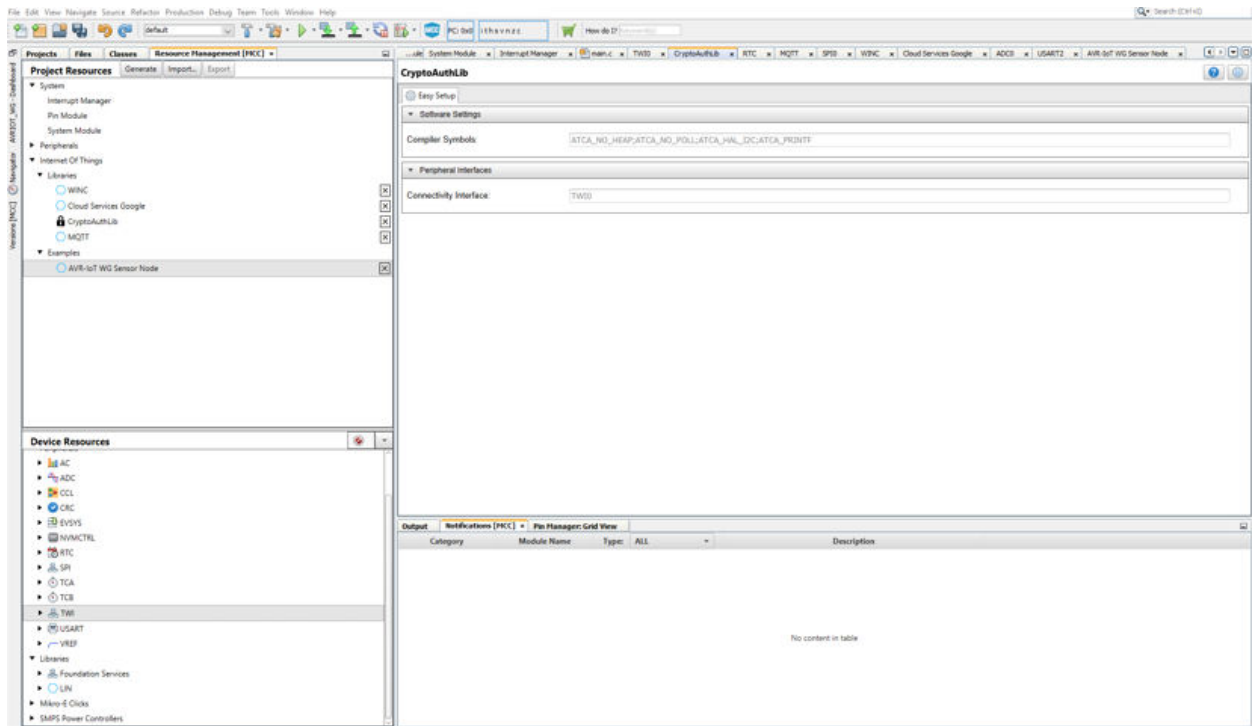
d. WINC

Under the WINC library, the user can change the settings for the SSID, password and the authentication type of the network to which the board will connect (see Figure 2-26).

Figure 2-26. WINC
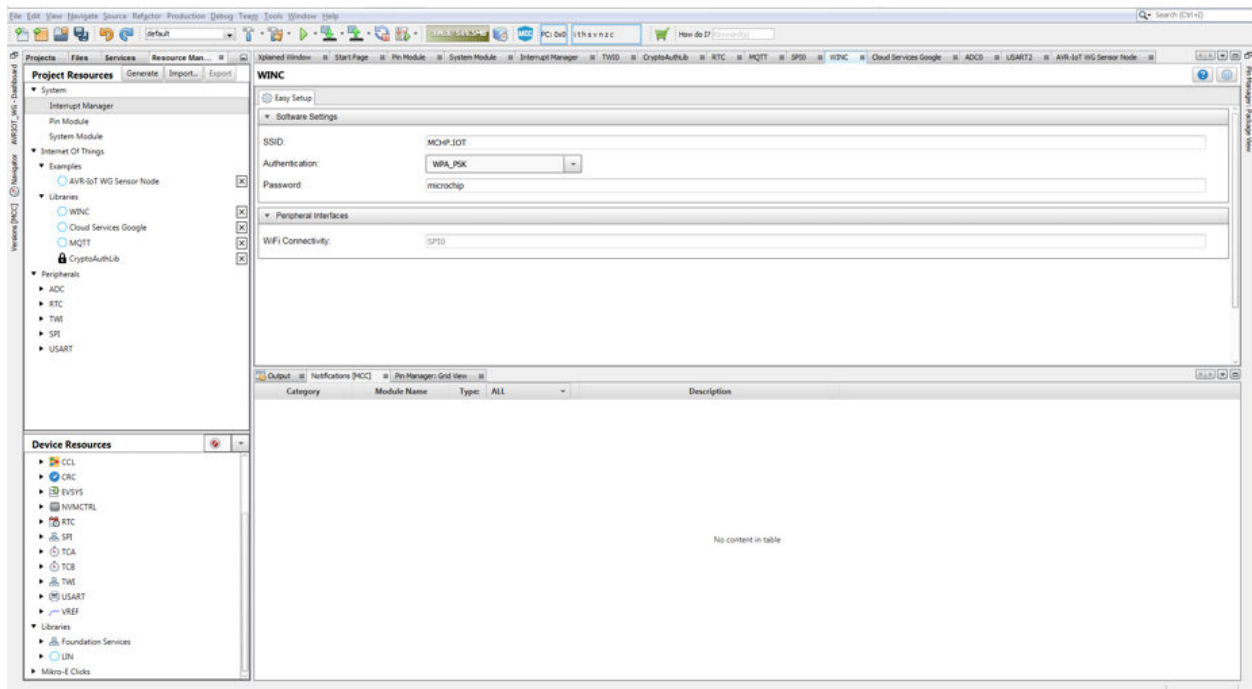
e. Cloud Services Google

The Cloud Services Google library contains settings for users to use their own Google Cloud Project by entering details such as Project ID, Project Region, and Registry ID. The default details used are from the public Microchip sandbox project (seeFigure 2-27).

Figure 2-27. Cloud Services Google
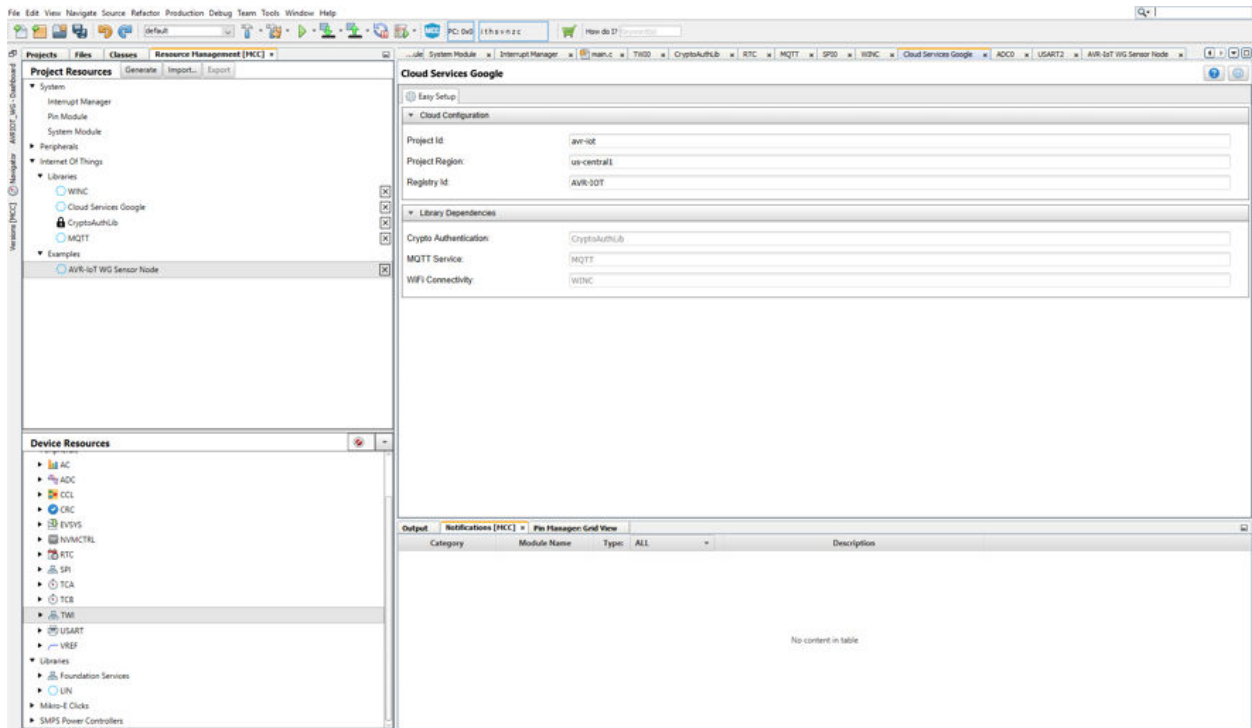
f. MQTT

The AVR-IoT WG relies on MQTT to transport data to the Cloud. In MCC, the user can change their MQTT host and connection time-out duration (see Figure 2-28).

Figure 2-28. MQTT
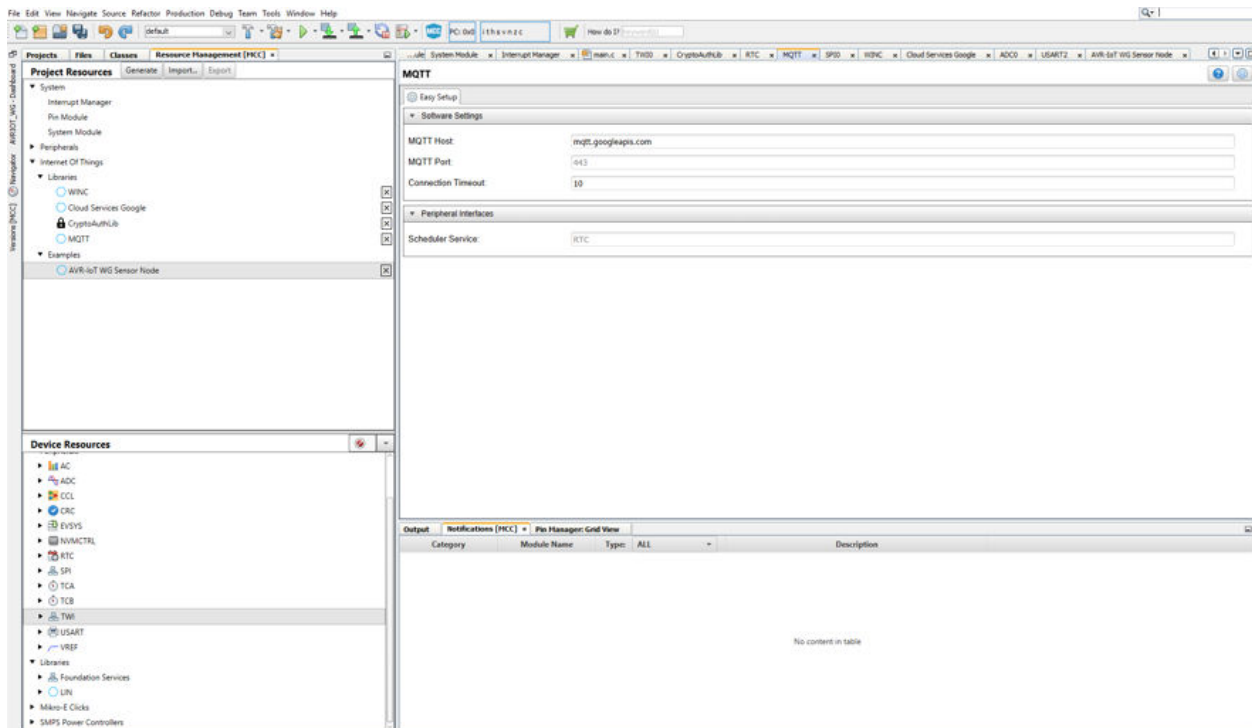
g. Generating MCC files and programming the board

Once the changes are made, click the '**Generate**' button on the left-hand corner of the window (see Figure 2-29) and wait for the generation to complete. For the code to work at optimal level, the user needs to change the optimization settings for the compilers. Right-click on the project name and select '**Properties**'. If the AVR GCC compiler is used, click '**avr-gcc**' under AVR-GCC Global Options (see Figure 2-30). If XC8 2.05 is used, click '**XC8 compiler**' under XC8 Global Options (see Figure 2-31). From there, select '**Optimizations**' in the Categories drop-down menu (see Figure 2-32 for AVR GCC and Figure 2-33 for XC8). Select "**s**" in the drop-down menu beside the 'Optimization Level' label option (see Figure 2-34 for AVR GCC and Figure 2-35 for XC8). Click the '**Apply**' button then '**OK**'. From there, click the '**Make and Program Device**' button near the middle of the toolbar (see Figure 2-36). Make sure the board is connected while programming.

Figure 2-29. Project Resources Generate
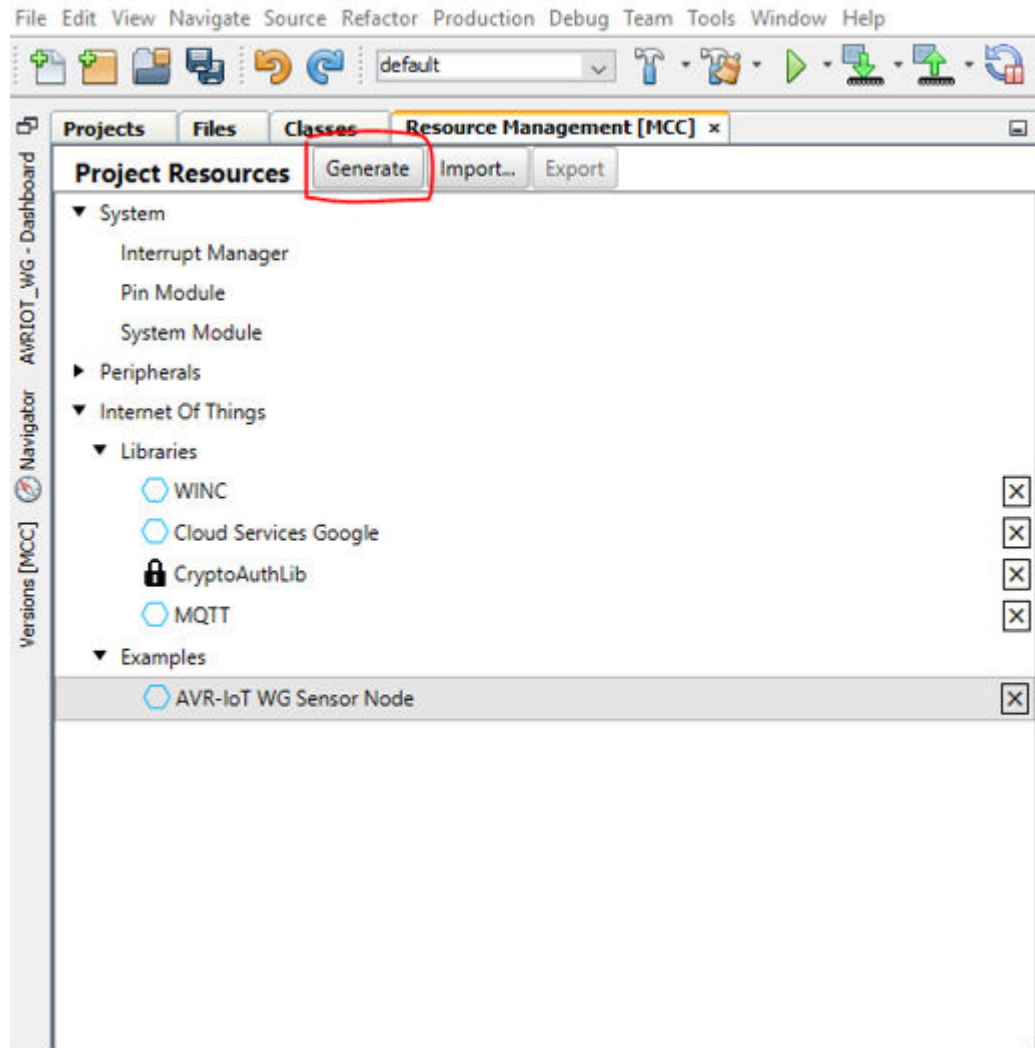
Figure 2-30. AVR GCC Settings
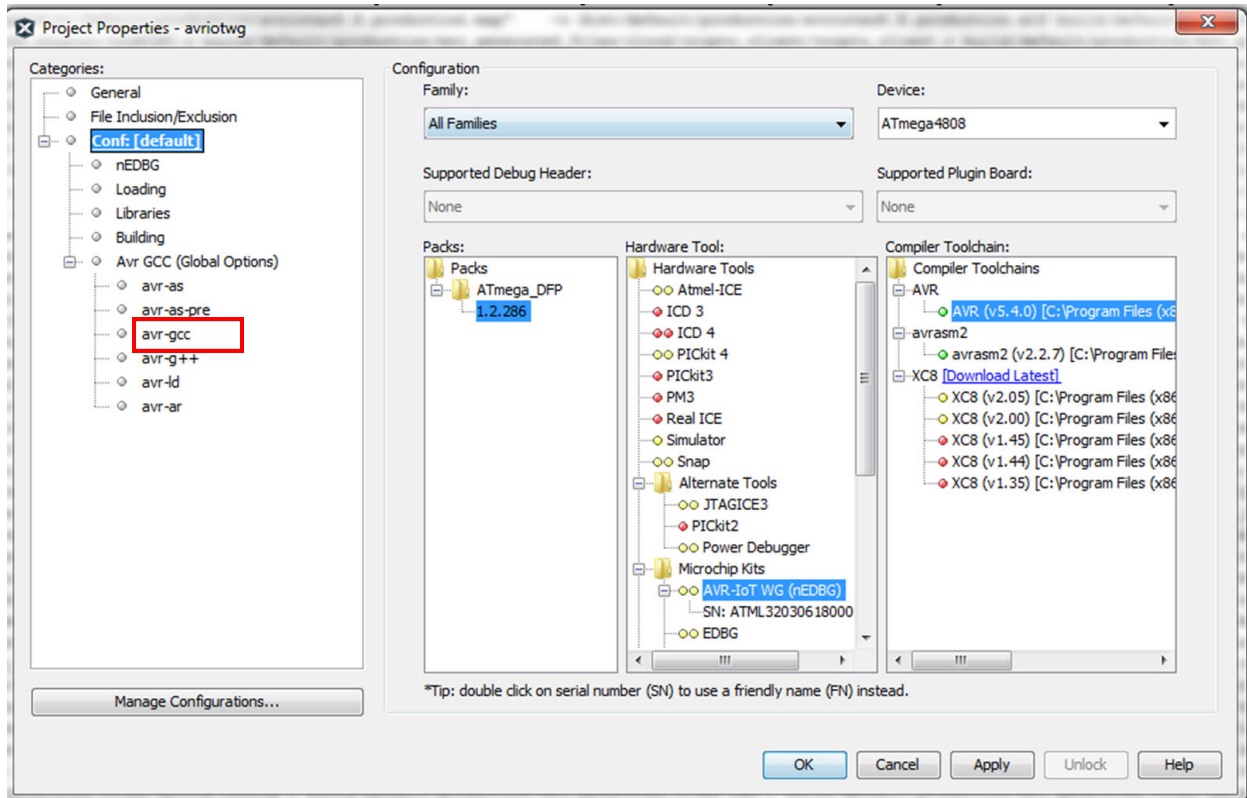
Figure 2-31. XC8 Settings
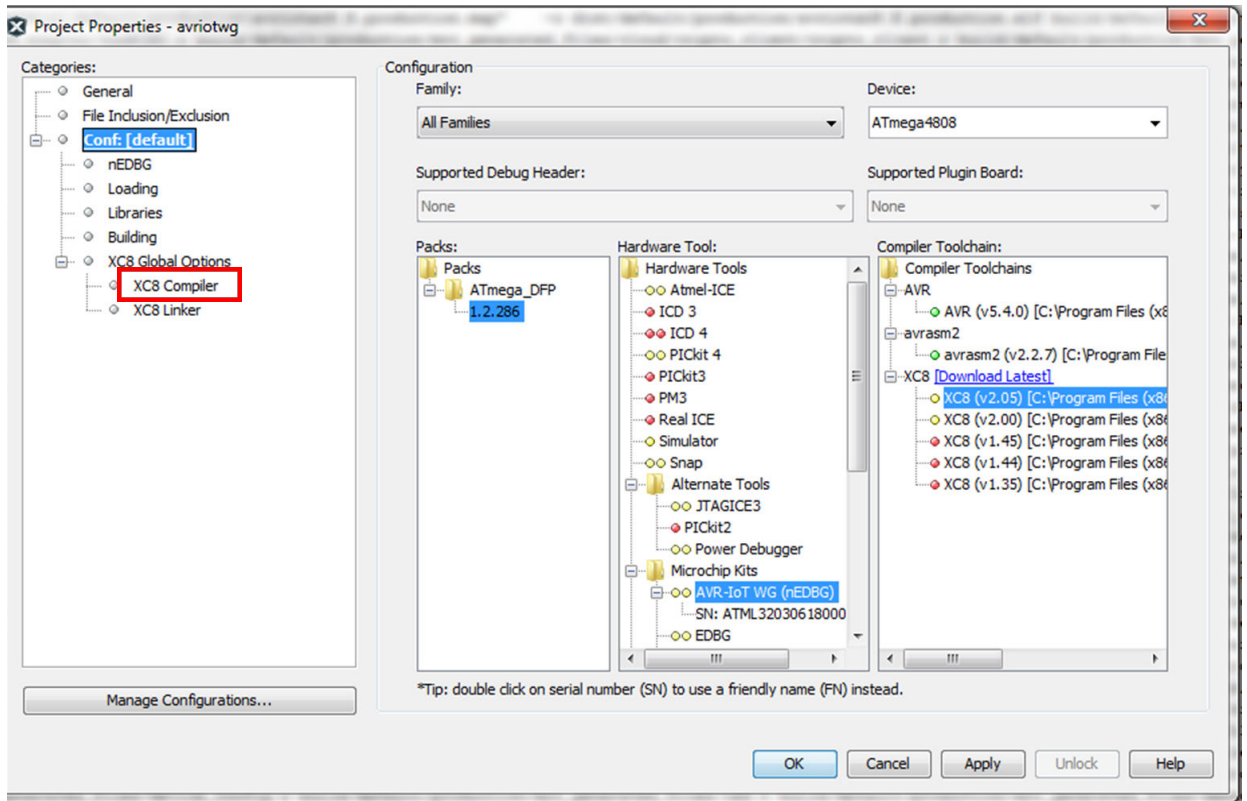
Figure 2-32. AVR GCC Option Categories
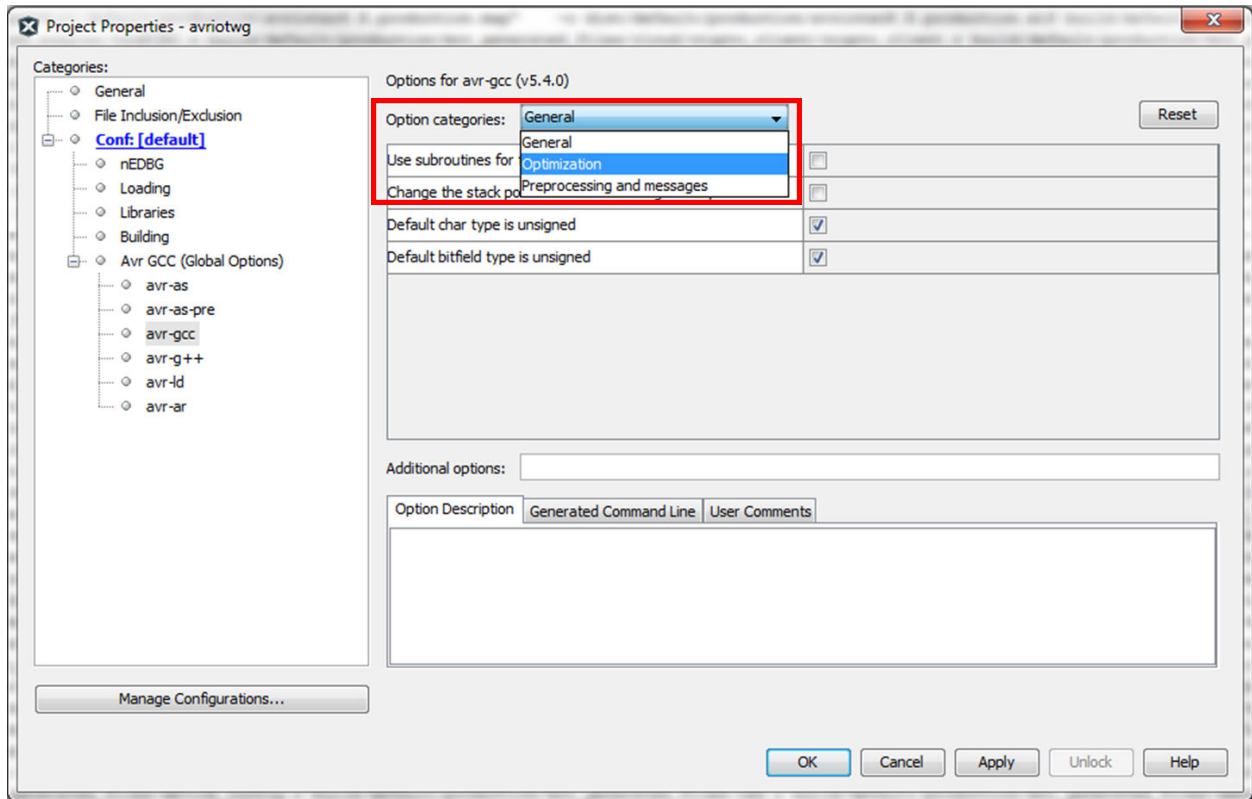
Figure 2-33. XC8 Option Categories
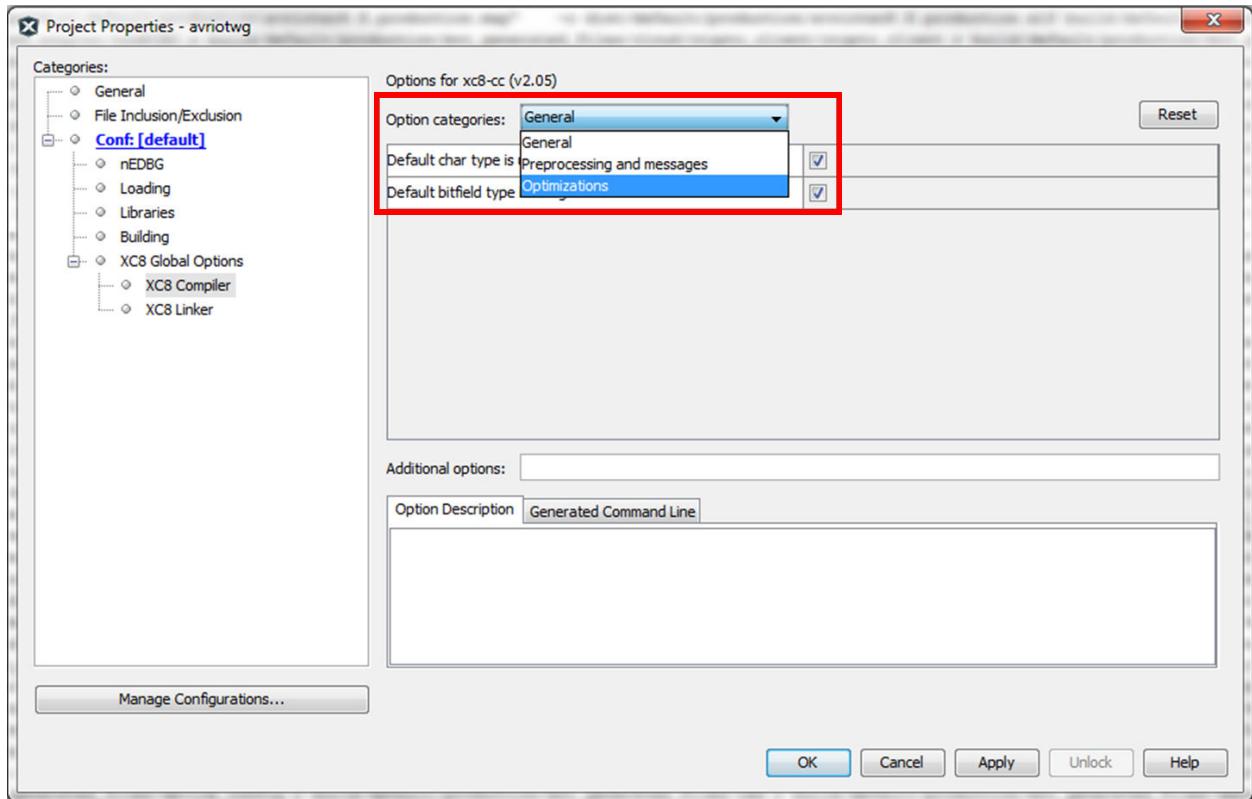
Figure 2-34. AVR GCC Optimization Level
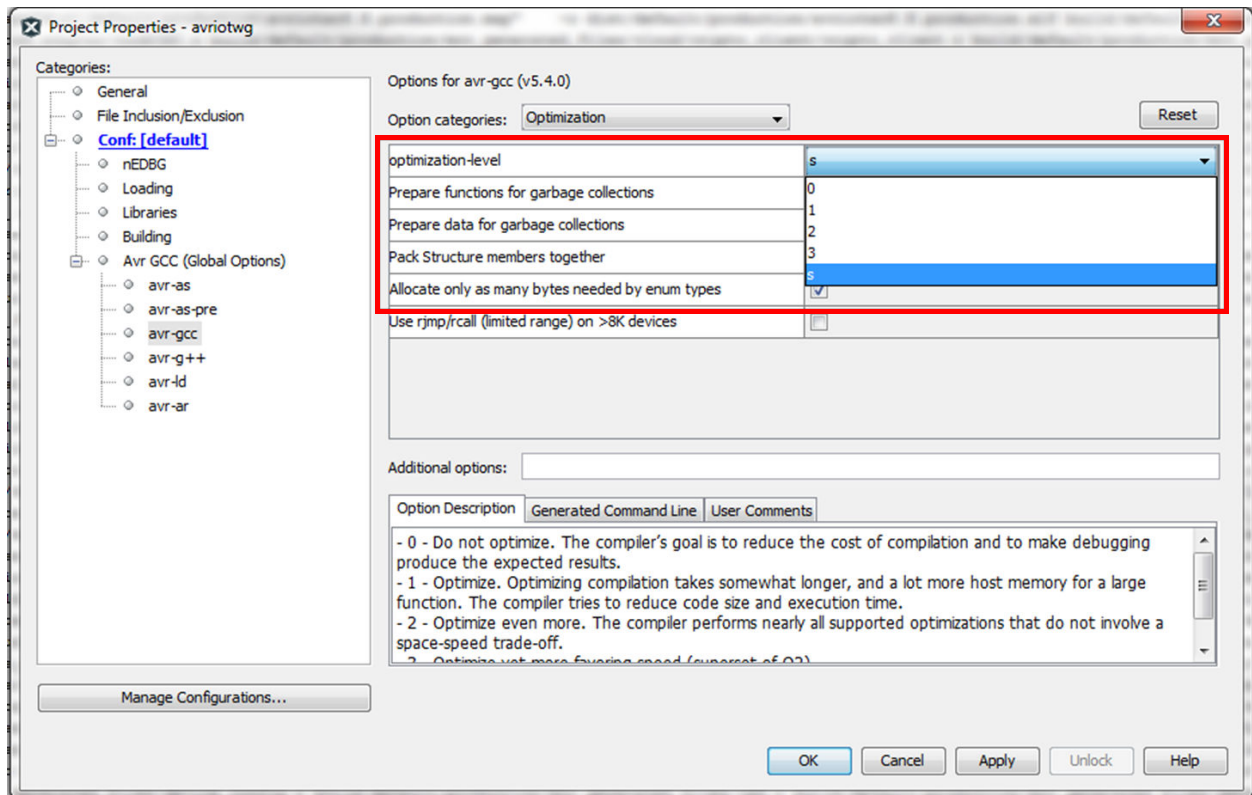
Figure 2-35. XC8 Optimization level
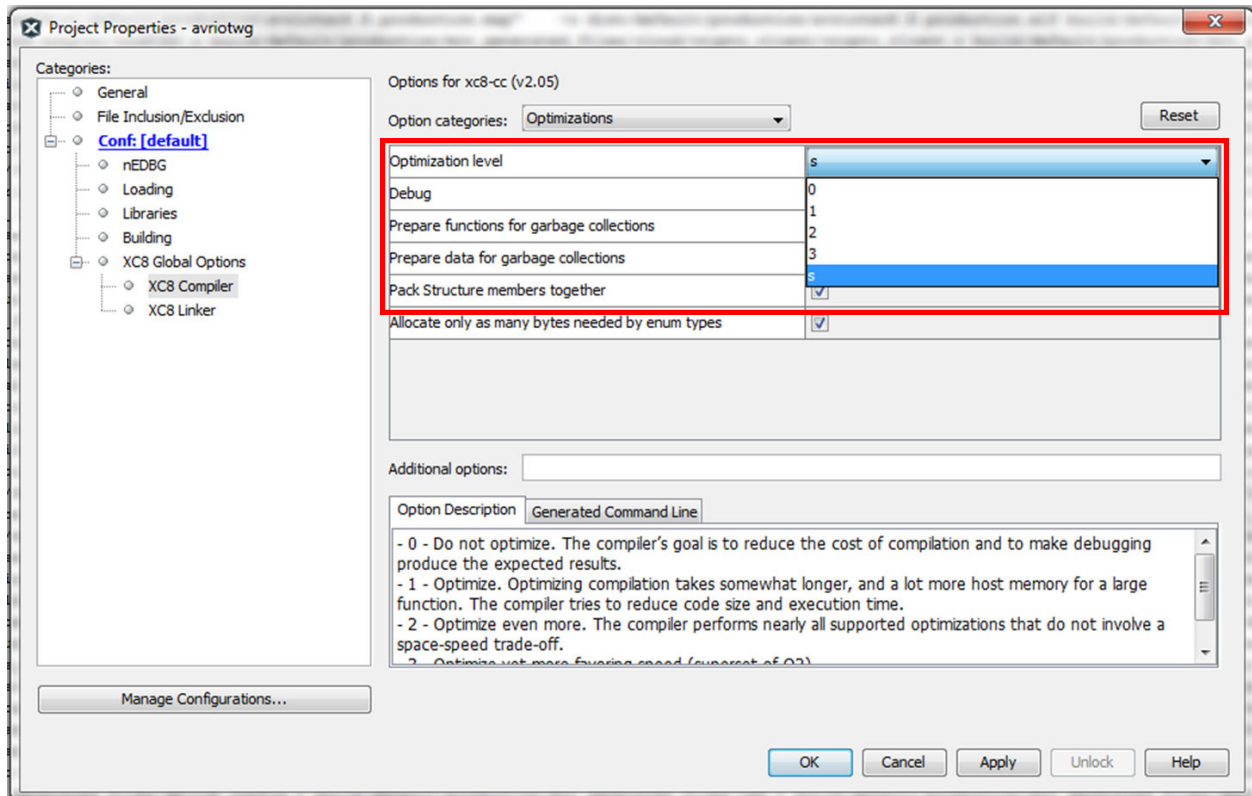
Figure 2-36. Make and Program the Device Button
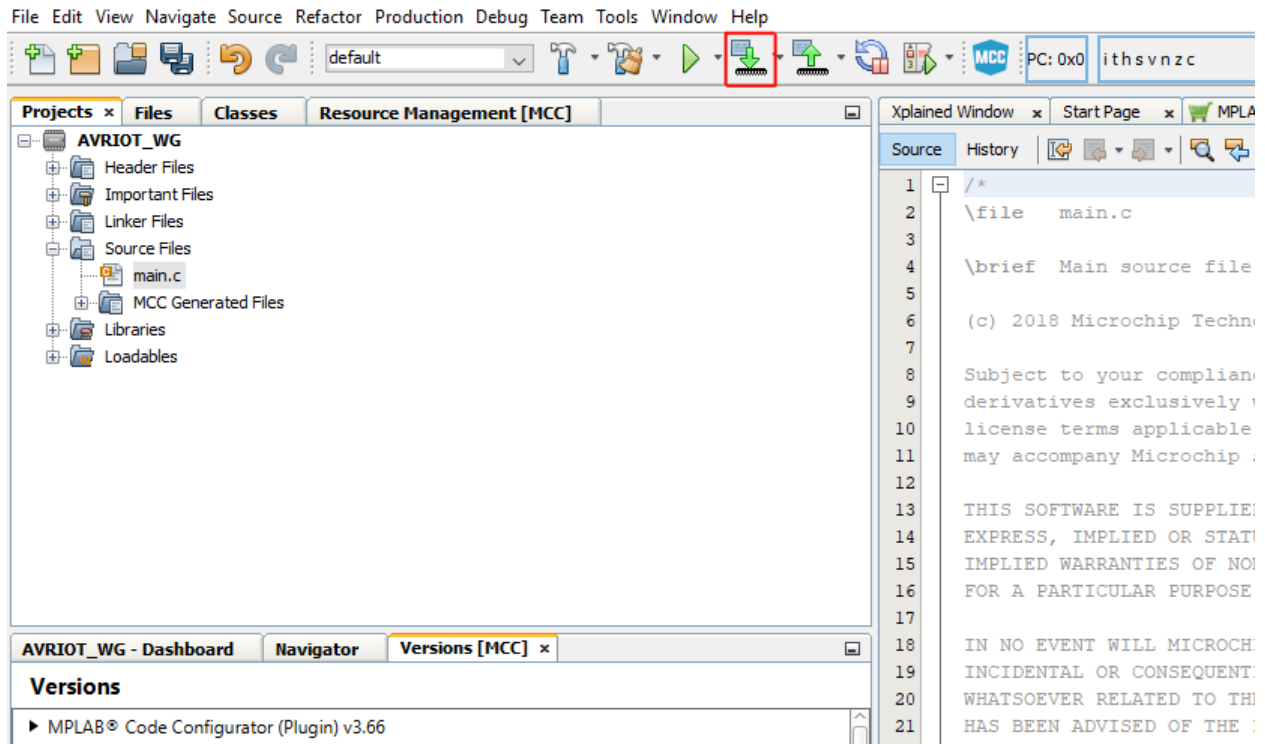
2.4 Advanced Modes

The AVR-IoT development board can be forced to enter one of a few advanced modes of operation at start-up. These modes can be entered by pressing one or a combination of the push buttons that are present on the board, labeled Switch 0 (SW0) and Switch 1 (SW1). Table 2-2 enumerates these advanced modes, descriptions, physical indicators of entering a specific mode, and how to enter them.

Table 2-3. AVR-IoT WG Advanced Modes

Advanced Mode	Description	Instructions	Physical Indicators
Soft AP mode	Software-Enabled Access mode enables the WINC to be made a wireless access point.	Press and hold SW0 at power-up.	All lights are off
WINC OTA mode*	Enables over-the-air WINC firmware updates.	Press and hold SW1 at power-up.	Blinking Green LED
Bootloader mode*	Enables ATmega bootloader.	Press and hold SW0 and SW1 at the same time.	Blinking Red LED

* - Not implemented in firmware code version 1.00.

2.4.1 Soft AP Mode

The AVR-IoT WG development board can be accessed through a Wi-Fi access point enabled by the Software-Enabled Access mode of the WINC1510. This can be another way to connect the board to a

Wi-Fi network. To enter Soft AP mode, press and hold the SW0 push button before plugging the board. When connecting to this access point for the first time, the user will need to set the SSID and password of the network to which they are connected, as shown in Figure 2-14. The user should enter these details and then press the **Connect** button. The board is now connected to the network.

Figure 2-37. Connecting to the network using Soft AP mode
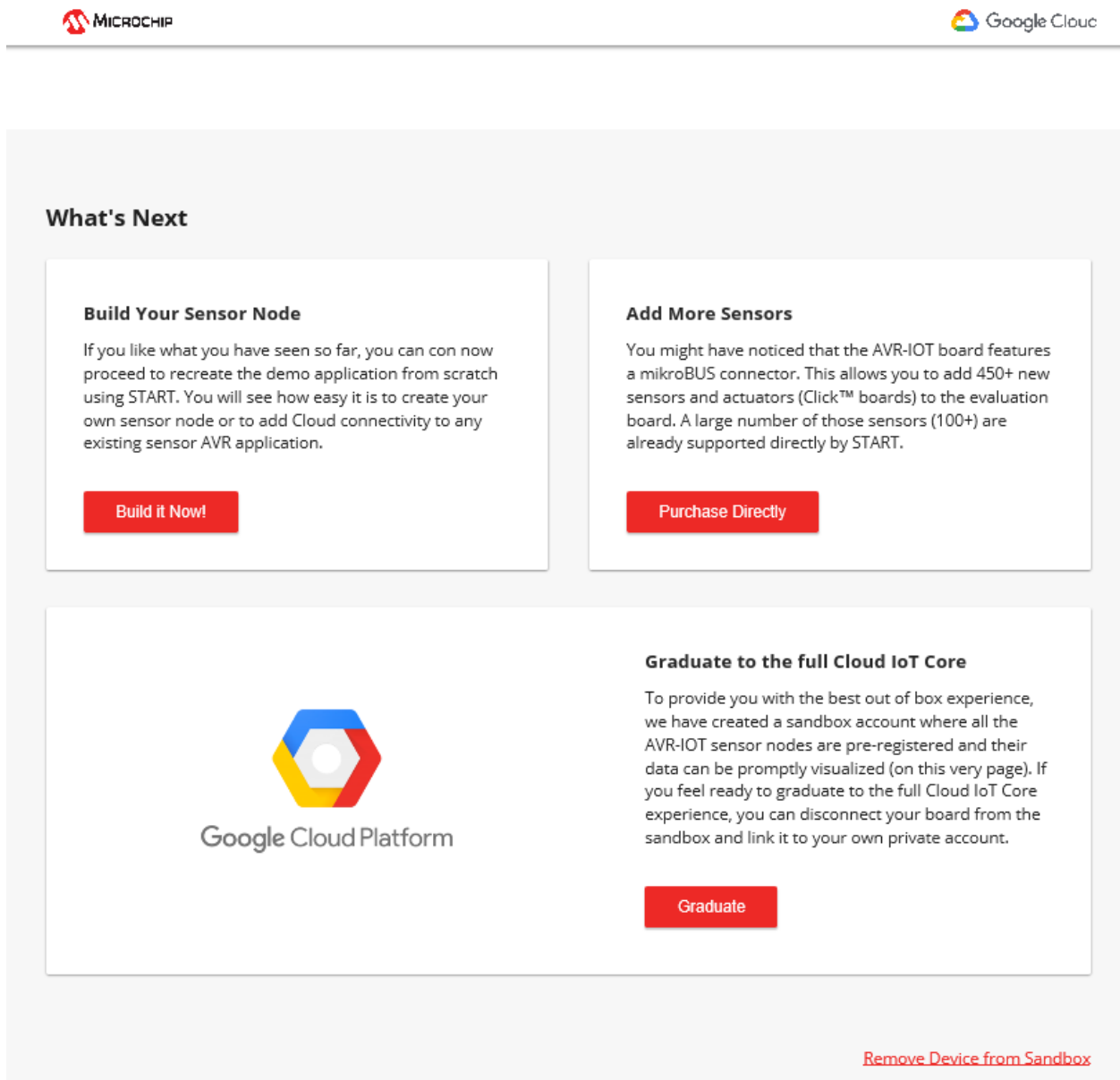
2.5 Migrating to a private Google Cloud account

Once the user is satisfied with the features and capabilities demonstrated by the AVR-IoT WG board, more information can be obtained by accessing the AVR-IoT WG sandbox. At the bottom of the avr-iot.com webpage, under the "What's Next" section, the user can find the "Graduate to the full Cloud IoT Core" experience option. Clicking the **Graduate** button unregisters the board from the Microchip sandbox account and transfers the users to a GitHub repository, containing the tutorials and files needed to connect the AVR-IoT WG board to the user's own Google Cloud account.

Figure 2-38. Migrating to a Private Google Cloud Account

3. Chapter 3: Troubleshooting

Table 3-1. Troubleshooting and Diagnostics

LED Sequence	Description	Diagnosis	Action
○○○●	Only Red LED is On	Board is not connected to Wi-Fi®	Verify Wi-Fi® credentials
●○○●	Blue and Red LEDs are On	Board is not connected to Google IoT Cloud servers	• Verify MQTT required ports. • Verify project credentials. • Check local network firewall settings. • Use tethered cellphone or laptop connection for internet.
●●○●	Blue, Green and Red LEDs are On	Sensor Data are not being published to the Cloud.	• Verify device registration to the project. • Check Google account for outages.
●●◐○	Blue and Green LEDs are On and Yellow LED is blinking	Everything is working	Nothing to be done.
○○○○	No LED is On	Board is not programmed	Download image `.hex` file from the Downloads section at the bottom of the Microsite page.
○nEDBG	nEDBG LED is Off	Board is not powered	• Check USB connection. • Replace the board.
●nEDBG	nEDBG LED is On but the Curiosity Drive is not found	Faulty USB connection	• Replace the USB connector • Check PC Device Manager.

4. Appendix A: Hardware Components

The AVR-IoT WG board features the following hardware components:

- ATmega4808 Microcontroller
- WINC1510 Wi-Fi Module
- Light and Temperature Sensors
- Four Light Emitting Diodes (1 each of Blue, Green, Yellow and Red)
- Two Mechanical Buttons
- mikroBUS Header Footprint
- nEDBG Programmer/Debugger

4.1 ATmega4808

The ATmega4808 is a microcontroller featuring the 8-bit AVR$^®$ processor with hardware multiplier - running at up to 20 MHz and with up to 48 KB Flash, 6 KB SRAM and 256 bytes of EEPROM in 28- and 32-pin packages. The series uses the latest Core Independent Peripherals (CIPs) with low-power features, including event system, intelligent analog and advanced peripherals.

Figure 4-1. ATmega4808
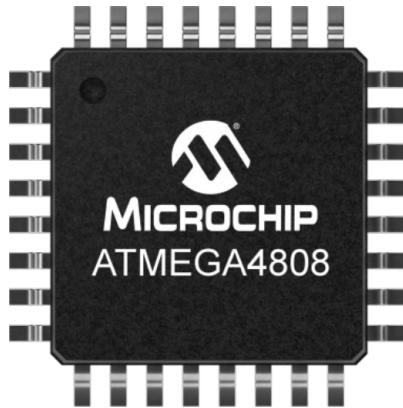
4.2 ATWINC1510

Microchip's WINC1510 is a low-power consumption 802.11 b/g/n IoT (Internet of Things) module, specifically optimized for low-power IoT applications. The module integrates the following: Power Amplifier (PA), Low-Noise Amplifier (LNA), switch, power management, and a printed antenna or a micro co-ax (u.FL) connector for an external antenna, resulting in a small form factor (21.7 x 14.7 x 2.1 mm) design. It is interoperable with various vendors' 802.11 b/g/n access points. This module provides SPI ports to interface with a host controller. The WINC1510 provides internal Flash memory as well as multiple peripheral interfaces, including UART and SPI. The only external clock source needed for the WINC1510 is the built-in, high-speed crystal or oscillator (26 MHz). The WINC1510 is available in a QFN package or as a certified module.

Figure 4-2. WINC1510

4.3 ATECC608A

The ATECC608A is a secure element from the Microchip CryptoAuthentication™ portfolio with advanced Elliptic Curve Cryptography (ECC) capabilities. With ECDH and ECDSA being built right in, this device is ideal for the rapidly growing IoT market, by easily supplying the full range of security such as confidentiality, data integrity, and authentication to systems with MCUs or MPUs running encryption/decryption algorithms. Similar to all Microchip CryptoAuthentication products, the new ATECC608A employs ultra-secure, hardware-based cryptographic key storage and cryptographic countermeasures, which eliminates any potential backdoors linked to software weaknesses.

Figure 4-3. ATECC608A

4.4 MCP9808 Temperature Sensor

The MCP9808 digital temperature sensor converts temperatures between -20°C and +100°C to a digital world with ±0.25°C/±0.5°C (typical/maximum) accuracy.

Additional Features

- Accuracy:
  ±0.25°C (typical) from -40°C to +125°C

  ±0.5°C (maximum) from -20°C to +100°C
- User Selectable Measurement Resolution:

0.5°C, 0.25°C, 0.125°C, 0.0625°C

- User Programmable Temperature Limits:
  1. Temperature Window Limit

  2. Critical Temperature Limit

- User Programmable Temperature Alert Output

- Operating Voltage Range: 2.7V to 5.5V

- Operating Current: 200 µA (typical)

- Shutdown Current: 0.1 µA (typical)

- 2-wire Interface: I^{2}C/SMBus Compatible

- Available Packages: 2x3 DFN-8, MSOP-8

- AEC-Q100 Qualified Grade 1

Figure 4-4. MCP9808

4.5 nEDBG

The AVR-IoT WG board contains an Embedded Debugger (nEDBG) for on-board programming and debugging. The nEDBG is a composite USB device of several interfaces: a debugger, a mass storage device, a data gateway and a Virtual COM port. Together with Atmel Studio, the nEDBG debugger interface can program and debug the ATmega4808. The Virtual COM port is connected to a UART on the ATmega4808 and provides an easy way to communicate with the target application through terminal software. It offers variable baud rate, parity, and Stop bit settings. The nEDBG controls one power and status LED on the AVR-IoT WG board. The table below shows how the LED is controlled in different operation modes.

The virtual COM port in the nEDBG requires the terminal software to set the Data Terminal Ready (DTR) signal to enable the UART pins connected to the ATmega4808. If the DTR signal is not enabled, the UART pins on the nEDBG are kept in high-Z (Tri-state) rendering the COM port unusable. The DTR signal is automatically set by some terminal software, but it may have to be manually enabled in your terminal.

Table 4-1. nEDBG LED CONTROL

Operation Mode	Status LED
Power-up	LED is lit - constant
Normal operation	LED is lit - constant

..........continued	
Operation Mode	**Status LED**
Programming	Activity indicator; the LED flashes slowly during programming/debugging with the nEDBG
Fault	The LED flashes fast if a power fault is detected.
Sleep/Off	LED is off. The nEDBG is either in Sleep mode or powered down. This can occur if the kit is externally powered.

5. Appendix B: Board Layout

Figure 5-1. AVR-IoT WG Development Board Layout
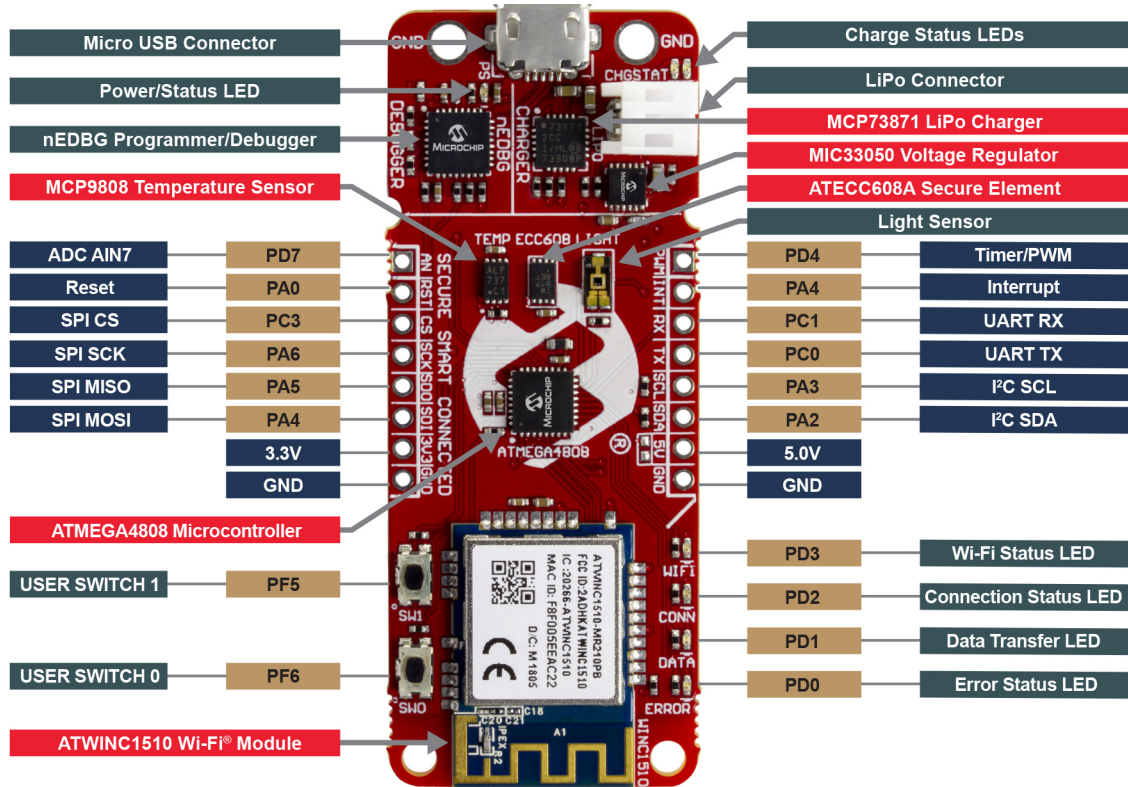
6. Appendix C: Firmware Flowchart

Figure 6-1. AVR-IoT WG Firmware Flowchart
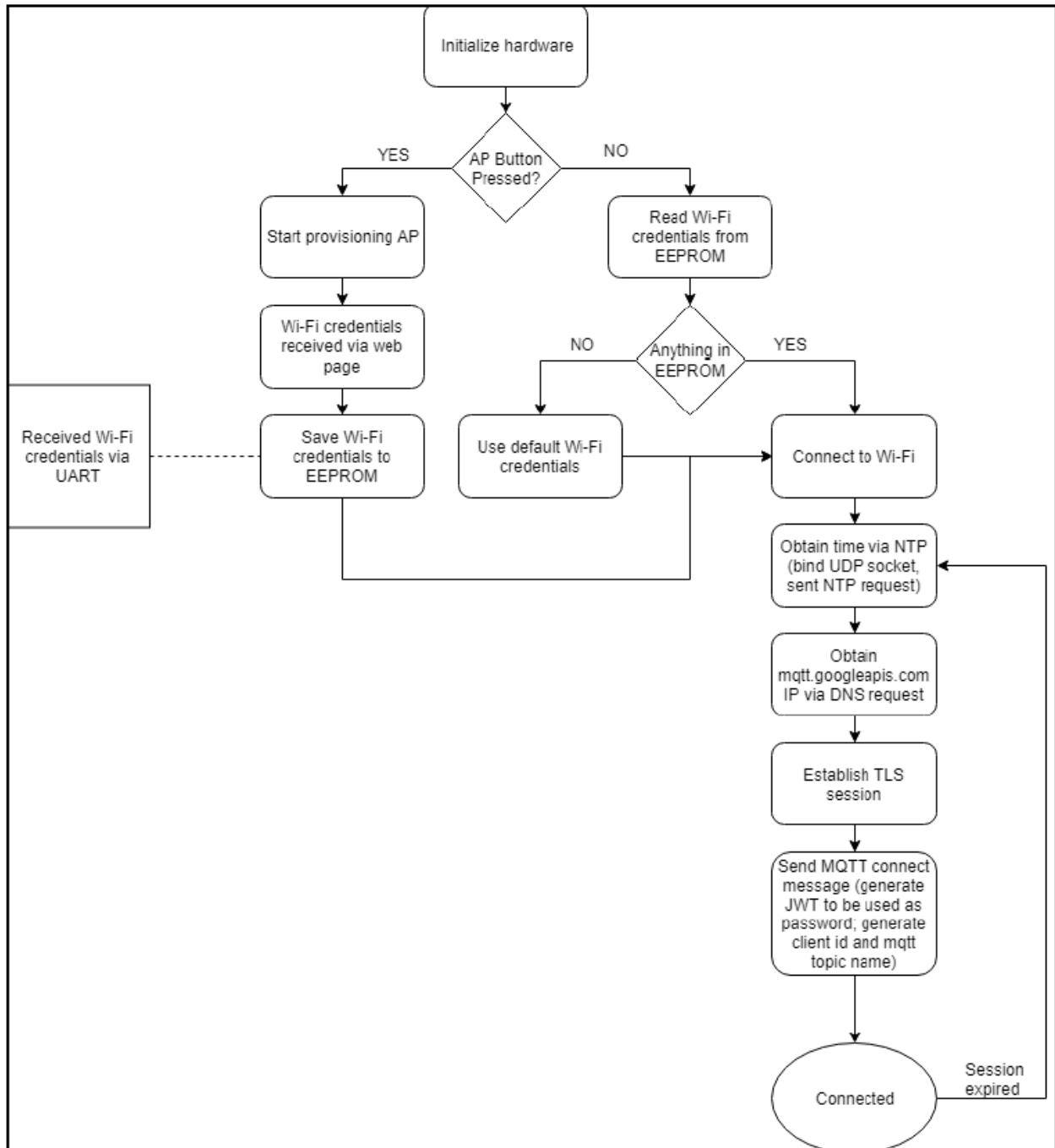
7. Appendix D: Relevant Links

The following list contains links to the most relevant documents and software for the AVR-IoT WG board. For those accessing the electronic version of this document, the underlined labels are clickable and will redirect to the appropriate website.

- Atmel Studio - Free IDE for the development of C/C++ and assembler code for microcontrollers.
- MPLAB® X IDE - Free IDE to develop applications for Microchip microcontrollers and digital signal controllers.
- IAR Embedded Workbench® for AVR® - This is a commercial C/C++ compiler that is available for 8-bit AVR microcontrollers. There is a 30-day evaluation version as well as a 4 KB code-size-limited kick-start version available on their website.
- Atmel START - Atmel START is an online tool that helps the user select and configure software components and tailor their embedded application in a usable and optimized manner.
- MPLAB® Code Configurator (MCC) - a free, graphical programming environment that generates seamless, easy-to-understand C code to be inserted into the project. Using an intuitive interface, it enables and configures a rich set of peripherals and functions specific to the application.
- Microchip Sample Store - Microchip sample store where you can order samples of devices.
- Data Visualizer - Data Visualizer is a program used for processing and visualizing data. The Data Visualizer can receive data from various sources such as the Embedded Debugger Data Gateway Interface found on Xplained Pro boards and COM ports.

8. Document Revision History

Doc. rev.	Date	Comment
B	11/2018	Added the AVR-IoT Development on MCC section.
A	10/2018	Initial document release.

The Microchip Web Site

Microchip provides online support via our web site at http://www.microchip.com/. This web site is used as a means to make files and information easily available to customers. Accessible by using your favorite Internet browser, the web site contains the following information:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQ), technical support requests, online discussion groups, Microchip consultant program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

Customer Change Notification Service

Microchip's customer notification service helps keep customers current on Microchip products. Subscribers will receive e-mail notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, access the Microchip web site at http://www.microchip.com/. Under "Support", click on "Customer Change Notification" and follow the registration instructions.

Customer Support

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Field Application Engineer (FAE)
- Technical Support

Customers should contact their distributor, representative or Field Application Engineer (FAE) for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in the back of this document.

Technical support is available through the web site at: http://www.microchip.com/support

Product Identification System

To order or obtain information, e.g., on pricing or delivery, refer to the factory or the listed sales office.

PART NO.   [X]⁽¹⁾   -   X   /XX   XXX

Device   Tape and Reel   Temperature   Package   Pattern
          Option   Range

Device:	PIC16F18313, PIC16LF18313, PIC16F18323, PIC16LF18323	
Tape and Reel Option:	Blank	= Standard packaging (tube or tray)
	T	= Tape and Reel[1]
Temperature Range:	I	= -40°C to +85°C (Industrial)
	E	= -40°C to +125°C (Extended)
Package:[2]	JQ	= UQFN
	P	= PDIP
	ST	= TSSOP
	SL	= SOIC-14
	SN	= SOIC-8
	RF	= UDFN
Pattern:	QTP, SQTP, Code or Special Requirements (blank otherwise)	

Examples:

- PIC16LF18313- I/P Industrial temperature, PDIP package
- PIC16F18313- E/SS Extended temperature, SSOP package

Note:

1. Tape and Reel identifier only appears in the catalog part number description. This identifier is used for ordering purposes and is not printed on the device package. Check with your Microchip Sales Office for package availability with the Tape and Reel option.
2. Small form-factor packaging options may be available. Please check http://www.microchip.com/packaging for small-form factor package availability, or contact your local Sales Office.

Microchip Devices Code Protection Feature

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.

- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable."

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

Legal Notice

Information contained in this publication regarding device applications and the like is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION, INCLUDING BUT NOT LIMITED TO ITS CONDITION, QUALITY, PERFORMANCE, MERCHANTABILITY OR FITNESS FOR PURPOSE. Microchip disclaims all liability arising from this information and its use. Use of Microchip devices in life support and/or safety applications is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights unless otherwise stated.

Trademarks

The Microchip name and logo, the Microchip logo, AnyRate, AVR, AVR logo, AVR Freaks, BitCloud, chipKIT, chipKIT logo, CryptoMemory, CryptoRF, dsPIC, FlashFlex, flexPWR, Heldo, JukeBlox, KeeLoq, Kleer, LANCheck, LINK MD, maXStylus, maXTouch, MediaLB, megaAVR, MOST, MOST logo, MPLAB, OptoLyzer, PIC, picoPower, PICSTART, PIC32 logo, Prochip Designer, QTouch, SAM-BA, SpyNIC, SST, SST Logo, SuperFlash, tinyAVR, UNI/O, and XMEGA are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

ClockWorks, The Embedded Control Solutions Company, EtherSynch, Hyper Speed Control, HyperLight Load, IntelliMOS, mTouch, Precision Edge, and Quiet-Wire are registered trademarks of Microchip Technology Incorporated in the U.S.A.

Adjacent Key Suppression, AKS, Analog-for-the-Digital Age, Any Capacitor, AnyIn, AnyOut, BodyCom, CodeGuard, CryptoAuthentication, CryptoAutomotive, CryptoCompanion, CryptoController, dsPICDEM, dsPICDEM.net, Dynamic Average Matching, DAM, ECAN, EtherGREEN, In-Circuit Serial Programming, ICSP, INICnet, Inter-Chip Connectivity, JitterBlocker, KleerNet, KleerNet logo, memBrain, Mindi, MiWi, motorBench, MPASM, MPF, MPLAB Certified logo, MPLIB, MPLINK, MultiTRAK, NetDetach, Omniscient Code Generation, PICDEM, PICDEM.net, PICkit, PICtail, PowerSmart, PureSilicon, QMatrix, REAL ICE, Ripple Blocker, SAM-ICE, Serial Quad I/O, SMART-I.S., SQI, SuperSwitcher, SuperSwitcher II, Total Endurance, TSHARC, USBCheck, VariSense, ViewSpan, WiperLock, Wireless DNA, and ZENA are trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

SQTP is a service mark of Microchip Technology Incorporated in the U.S.A.

Silicon Storage Technology is a registered trademark of Microchip Technology Inc. in other countries.

GestIC is a registered trademark of Microchip Technology Germany II GmbH & Co. KG, a subsidiary of Microchip Technology Inc., in other countries.

AMBA, Arm, Arm7, Arm7TDMI, Arm9, Arm11, Artisan, big.LITTLE, Cordio, CoreLink, CoreSight, Cortex, DesignStart, DynamIQ, Jazelle, Keil, Mali, Mbed, Mbed Enabled, NEON, POP, RealView, SecurCore, Socrates, Thumb, TrustZone, ULINK, ULINK2, ULINK-ME, ULINK-PLUS, ULINKpro, µVision, Versatile are trademarks or registered trademarks of Arm Limited (or its subsidiaries) in the US and/or elsewhere.

Quality Management System Certified by DNV

ISO/TS 16949

Microchip received ISO/TS-16949:2009 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona; Gresham, Oregon and design centers in California and India. The Company's quality system processes and procedures are for its PIC® MCUs and dsPIC® DSCs, KEELOQ® code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.

MICROCHIP

Worldwide Sales and Service

AMERICAS	ASIA/PACIFIC	ASIA/PACIFIC	EUROPE
Corporate Office 2355 West Chandler Blvd. Chandler, AZ 85224-6199 Tel: 480-792-7200 Fax: 480-792-7277 Technical Support: http://www.microchip.com/ support Web Address: www.microchip.com **Atlanta** Duluth, GA Tel: 678-957-9614 Fax: 678-957-1455 **Austin, TX** Tel: 512-257-3370 **Boston** Westborough, MA Tel: 774-760-0087 Fax: 774-760-0088 **Chicago** Itasca, IL Tel: 630-285-0071 Fax: 630-285-0075 **Dallas** Addison, TX Tel: 972-818-7423 Fax: 972-818-2924 **Detroit** Novi, MI Tel: 248-848-4000 **Houston, TX** Tel: 281-894-5983 **Indianapolis** Noblesville, IN Tel: 317-773-8323 Fax: 317-773-5453 Tel: 317-536-2380 **Los Angeles** Mission Viejo, CA Tel: 949-462-9523 Fax: 949-462-9608 Tel: 951-273-7800 **Raleigh, NC** Tel: 919-844-7510 **New York, NY** Tel: 631-435-6000 **San Jose, CA** Tel: 408-735-9110 Tel: 408-436-4270 **Canada - Toronto** Tel: 905-695-1980 Fax: 905-695-2078	**Australia - Sydney** Tel: 61-2-9868-6733 **China - Beijing** Tel: 86-10-8569-7000 **China - Chengdu** Tel: 86-28-8665-5511 **China - Chongqing** Tel: 86-23-8980-9588 **China - Dongguan** Tel: 86-769-8702-9880 **China - Guangzhou** Tel: 86-20-8755-8029 **China - Hangzhou** Tel: 86-571-8792-8115 **China - Hong Kong SAR** Tel: 852-2943-5100 **China - Nanjing** Tel: 86-25-8473-2460 **China - Qingdao** Tel: 86-532-8502-7355 **China - Shanghai** Tel: 86-21-3326-8000 **China - Shenyang** Tel: 86-24-2334-2829 **China - Shenzhen** Tel: 86-755-8864-2200 **China - Suzhou** Tel: 86-186-6233-1526 **China - Wuhan** Tel: 86-27-5980-5300 **China - Xian** Tel: 86-29-8833-7252 **China - Xiamen** Tel: 86-592-2388138 **China - Zhuhai** Tel: 86-756-3210040	**India - Bangalore** Tel: 91-80-3090-4444 **India - New Delhi** Tel: 91-11-4160-8631 **India - Pune** Tel: 91-20-4121-0141 **Japan - Osaka** Tel: 81-6-6152-7160 **Japan - Tokyo** Tel: 81-3-6880- 3770 **Korea - Daegu** Tel: 82-53-744-4301 **Korea - Seoul** Tel: 82-2-554-7200 **Malaysia - Kuala Lumpur** Tel: 60-3-7651-7906 **Malaysia - Penang** Tel: 60-4-227-8870 **Philippines - Manila** Tel: 63-2-634-9065 **Singapore** Tel: 65-6334-8870 **Taiwan - Hsin Chu** Tel: 886-3-577-8366 **Taiwan - Kaohsiung** Tel: 886-7-213-7830 **Taiwan - Taipei** Tel: 886-2-2508-8600 **Thailand - Bangkok** Tel: 66-2-694-1351 **Vietnam - Ho Chi Minh** Tel: 84-28-5448-2100	**Austria - Wels** Tel: 43-7242-2244-39 Fax: 43-7242-2244-393 **Denmark - Copenhagen** Tel: 45-4450-2828 Fax: 45-4485-2829 **Finland - Espoo** Tel: 358-9-4520-820 **France - Paris** Tel: 33-1-69-53-63-20 Fax: 33-1-69-30-90-79 **Germany - Garching** Tel: 49-8931-9700 **Germany - Haan** Tel: 49-2129-3766400 **Germany - Heilbronn** Tel: 49-7131-67-3636 **Germany - Karlsruhe** Tel: 49-721-625370 **Germany - Munich** Tel: 49-89-627-144-0 Fax: 49-89-627-144-44 **Germany - Rosenheim** Tel: 49-8031-354-560 **Israel - Ra'anana** Tel: 972-9-744-7705 **Italy - Milan** Tel: 39-0331-742611 Fax: 39-0331-466781 **Italy - Padova** Tel: 39-049-7625286 **Netherlands - Drunen** Tel: 31-416-690399 Fax: 31-416-690340 **Norway - Trondheim** Tel: 47-72884388 **Poland - Warsaw** Tel: 48-22-3325737 **Romania - Bucharest** Tel: 40-21-407-87-50 **Spain - Madrid** Tel: 34-91-708-08-90 Fax: 34-91-708-08-91 **Sweden - Gothenberg** Tel: 46-31-704-60-40 **Sweden - Stockholm** Tel: 46-8-5090-4654 **UK - Wokingham** Tel: 44-118-921-5800 Fax: 44-118-921-5820